CIT 322: INTRODUCTION TO INTERNET PROGRAMMING

NATIONAL OPEN UNIVERSITY OF NIGERIA

FACULTY OF SCIENCE

COURSE CODE: CIT 322

COURSE TITLE: INTRODUCTION TO INTERNET PROGRAMMING

COURSE GUIDE

CIT 322
INTRODUCTION TO INTERNET PROGRAMMING

Course Team Dr. Ikhu Omoregbe Nicholas
(Course Developer/Writer)- Covenant University,
Ogun State
Prof. Olayide Abass (Course Editor) - UNILAG
Ms. A.A. Afolorunso (Programme Leader)-NOUN

Reviewer Maitanmi, Stephen O. Ph.D
Babcock University, Ilisan Remo, Ogun State,
Nigeria

NATIONAL OPEN UNIVERSITY OF NIGERIA

National Open University of Nigeria
Headquarters
14/16 Ahmadu Bello Way
Victoria Island, Lagos

Abuja Office
5 Dar es Salaam Street
Off Aminu Kano Crescent
Wuse II, Abuja

e-mail: centralinfo@nou.edu.ng
URL: www.nou.edu.ng
Published by
National Open University of Nigeria

Printed 2013

ISBN: 978-058-196-0

Printed by
For National Open University of Nigeria

CONTENTS **PAGE**

INTRODUCTION

CIT 322 – Introduction to Internet Programming is a three-credit unit course. It deals with the structure and functionality of the world wide web, create static and dynamic web pages using a combination of HTML, CSS, and JavaScript, by applying essential programming skills in creating HTML forms, select an appropriate web hosting service, and publish your web pages globally.

WHAT YOU WILL LEARN IN THIS COURSE

This course will provide you with the necessary skills required to design and deploy solutions on the Internet. A number of tools for constructing Web application include Hypertext Markup Language (HTML), Extensible Markup language (XML), Cascading Style Sheets, JavaScript, and overview of XML, search engines and tools, and the future web, technologies and development tools have a wide coverage in this course.

This course is divided into four modules. Module one lays the background for the entire course. It provides fundamental information on the meaning, origin, architecture and services offered by the Internet. It also explores network models and the protocols that work behind the scene to display Web pages. Module two focuses on Hypertext Markup Language (HTML) Tags and attributes, and extensible markup language (XHTML) and HTML5 as some of the most recent and important languages used to construct Web pages. This module concentrates on the syntax and the various HTML tags, elements and attributes used in developing Web page. Some of these include anchor, table, form, image, video, and audio. Module three explores Cascading Style Sheets and JavaScript for formatting text and adding interactivity respectively to Web pages. The module contains some sample codes in JavaScript and Cascading Style Sheets that demonstrate their features and capabilities in making Web pages come alive. Module four contains information on overview of XML, search engines and tools, and the future web, technologies and development tools.

COURSE AIMS

The aim of this course is to equip you with the basic skills of studying and understanding Internet programming as well as lay the foundation of the basic knowledge and tools you need to become a proficient Web content developer /administrator. Specifically, this aims to:

- introduce you to the concepts, features and services of the Internet
- explore the architecture for Web applications and the tools for building standard websites
- explore and use various HTML tags for web development
- explore the relevant of XML in web development
- expose you to the techniques for writing Cascading Style Sheet as a standardised way of imposing style on the content of HTML tags
- teach you how to write JavaScript which is used for adding interactivity to static pages
- programming interface for both HTML and XML documents.
- Form validation

COURSE OBJECTIVES

Certain objectives have been set out to ensure that the course achieves its aims. Apart from the general objectives of this course, each unit of this course has set objectives. At the end of this course, you should be able to:

- define and discuss the evolution of the Internet and explain the meaning of Intranet and extranet
- list the devices used to access the Internet and explain the various means of accessing the Internet
- differentiate between statis and dynamic pages
- describe the term “computer network,” discuss the client-server model and describe the Web application architecture
- explain the term “HTML,” write simple HTML codes using popular tags and use Web browsers to display HTML codes
- explain the term “XHTML,” write simple HTML5 codes using popular tags and use Web browsers to display XTML codes
- outline how to create, modify, process, view and validate XML document
- write HTML codes to process form information, explain how to use Form action and Methods and discuss and use various form elements
- discuss and state the importance of CSS, use CSS format web pages and add CSS to HTML files
- explain the meaning of JavaScript, write and run simple JavaScript programs
- and hierarchy of the model
- Form validation

WORKING THROUGH THIS COURSE

In order to have a thorough understanding of the course units, you will need to read and understand the contents, practice what you have learnt by studying and developing simple websites and Web applications for your organisation and be committed to learning and using skills acquired from the course to enhance your career.

COURSE MATERIALS

The materials you will need for this course include:

1. Course Guide
2. Study Units
3. Recommended Texts
4. A file for your assignments and records to monitor your progress.

STUDY UNITS

There are four modules broken down into 14 study units in this course. They are listed as follows:

Module1

Unit 1	Definitions and how to connect to the Internet
Unit 2	Internet services, communication and protocol
Unit 3	Network model and web application development

Module 2

Unit 1	HTML tags and attributes
Unit 2	XHTML and HTML5 Elements
Unit 3	Tables
Unit 4	Input Tags and Form processing

Module 3

Unit 1	Cascading Style Sheet (CSS) Fundamentals
Unit 2	Fundamental of JavaScript for dynamic statements
Unit 3	Using statements in control JavaScript
Unit 4	Event Handlers in JavaScript

Module 4

Unit	1	Overview of XML
Unit	2	Search Engines and Tools
Unit	3	The future Web, Technologies and Development Tools

TEXTBOOKS AND REFERENCES

Alex, L. & Mathew, L. (1999). *Fundamentals of Information Technology*. New Delhi: Vikas Publishing House PVT Ltd.

Andy, S. (1999). *Computer Communications, Principles and Business Applications*. (2nd ed.). England: McGraw-Hill.

Boulton, M. (2009). A Practical Guide to Designing for the Web. United Kingdom: Mark Boulton Design Ltd.

Barbara, K. K. & Norman, J. M. (2001). *The World Wide Web: A Mass Communication Perspective*. USA: Mayfield Publishing Company.

Barrie, S. &Valda, H.(2004). *Programming the Web: An Introduction*. USA: McGraw-Hill.

Behrouz, A. F. (2003).*Data Communications and Networking*. (3rd ed.). International Edition. N.Y, USA: McGraw-Hill/Osborne.

Brian, K. W, Stacey, C. S. & Sarah E. H. (1999). *Using Information Technology: A Practical Introduction to Computers and Communication*. Irwin/McGraw-Hill

Deitel, P. J. & Deitel, H.M. (2008). *Internet and World Wide Web: How to Program*. (4th ed.). New Jersey, USA: Pearson Prentice Hall.

Gasston, P. (2013). The Modern Web-Multi-Device Web Development with HTML5, CSS3 and JavaScript. San Francisco: No Starch Press.

Idowu, S.A., Maitanmi, S.O. & Adetunji, O. O. (2020). Introductory to Web Technology and Development. Nigeria: Jamiro Press.

June, C. (2003). *The Unusually Useful Web Book*. USA: New Ride.

MacBride, K. (2006). *Brilliant Internet for the Over 50s: What you need to know about it*. England: Pearson Education Limited.

Marc, D. M. & Thomas, C. P. (2003). *Web Design Using Macromedia Dreamweaver*. USA: McGraw-Hill/ Irwin.

Miller, D. (2006). *Data Communications and Network.* New York: McGraw Hill.

Nagpal, D. P. (2006). *Web Design Technology, Theory and Technique on the Cutting Edge.* New Delhi, India: S. Chand and Company Ltd.

Shklar, L. & Rosen, R. (2009). Web Application Architecture, Principles, Protocols and Practices. England: John Wiley & Sons Ltd.

Wang, P., & Katila, S. (2003). An Introduction to Web Design and Programming. Brooks/Cole book/

Nolan, H. (2005). *Creating a Web Page in Dreamweaver*. USA: Peachpit Press, Berkeley.

Weverka, P. (2001). *Instant Web Pages*. USA: Sybex Inc.

Raymond, G. & Ellen, H. (2001). *In-Line/On-Line Fundamentals of the Internet and World Wide Web.* USA: McGraw-Hill Learning Centre.

Richard, A .M. Sr. (2003). *Introduction to Networking*. N.Y, USA: McGraw-Hill/Osborne.

Robert, W. S. (2009). *Programming the World Wide Web*. New Jersey, USA: Pearson Addition-Wesley.

Terry, F-M. (2009). *Web Development and Design Foundations with XHTML* USA*:* Pearson International Edition.

Katarzyna J. M. “Internet and You: Connecting to the Internet”. Retrieved from http://www.rsna.org/Technology/internet2-1.cfm

ASSIGNMENT FILE

An Assignment File and a marking scheme will be made available to you. In the File, you will find details of the work you must submit to your tutor for marking. There are two aspects of assessment of this course: the tutor-marked and the written examination. The marks you obtain in these two areas will make up your final marks. The assignment must be submitted to your tutor for formal assessment according to the deadline stipulated in the presentation schedule and Assignment File. The work you submit to your tutor for assessment will account for 30 per cent of your total score.

ASSESSMENT

There are two aspects to the assessment of the course. First are the tutor marked assignments; second, is a written examination. In tackling the assignments, you are expected to apply information and knowledge acquired during this course. The assignments must be submitted to your tutor for formal assessment in accordance with the deadlines stated in the Assignment File. The work you submit to your tutor for assessment will count for 30 per cent of your total course mark.

At the end of the course, you will need to sit for a final three-hour examination. This will also count for 70 per cent of your total course mark.

PRESENTATION SCHEDULE

The presentation schedule included in your course materials gives you the important dates for the completion of tutor marked assignments and attending tutorials. Remember, you are required to submit all your assignments by the due date. You should guard against lagging behind in your work.

TUTOR-MARKED ASSIGNMENT

There are 14 tutor-marked assignments in this course. You will be assessed on four of them but the best three performances from the TMAs will be used for your 30 per cent grading. Assignment questions for the units in this course are contained in the Assignment File. You should be able to complete your assignments from the information and materials contained in your set textbooks, reading and study units. However, you may wish to use other references to broaden your viewpoint and provide a deeper understanding of the subject. When you have completed each assignment, send it together with form to your tutor. Make sure that each assignment reaches your tutor on or before the deadline given.

EXAMINATION AND GRADING

The final examination for the course will carry 70 per cent of the total marks available for this course. The examination will cover every aspect of the course, so you are advised to revise all your corrected assignments before the examination.

COURSE MARKING SCHEME

This table shows how the actual course marking is broken down.

***Table 1*: Course Marking Scheme**

Assessment	**Marks**
Assignment 1- 4	Four assignments, best three marks of the four count at 30% of course marks
Final Examination	70% of overall course marks
Total	100% of course marks

COURSE OVERVIEW

Unit	Title of Work	Weeks Activity	Assessment (End of Unit)
	Course Guide	Week 1	
	Module 1		
1	Definitions of and how to connect to the Internet	Week 1	Assignment 1
2	Internet services, communication and protocol	Week 2	Assignment 2
3	Network model and web application development	Week 3	Assignment 3
	Module 2		
1	HTML tags and attributes	Week 4	Assignment 4
2	XHTML and HTML5 elements	Week 5	Assignment 5
3	Tables	Week 6	Assignment 6
4	Input Tags and Form processing	Week 7	Assignment 7
	Modules 3		
1	Cascading Style Sheet Fundamentals	Week 8	Assignment 8
2	JavaScript for dynamic statements	Week 9	Assignment 9
3	Using statements in control JavaScript	Week 10	Assignment 10
4	Evens Handlers in JavaScript	Week 11	Assignment 11
	Module 4		
1	Overview of XML	Week 12	Assignment 12
2	Search Engines and Tools	Week 13	Assignment 12
3	The future Web Technologies and Development Tools	Week 14	Assignment 13
	Revision	Week 15	
	Examination	Week 16	
Total		16 weeks	

HOW TO GET THE MOST FROM THIS COURSE

In distance learning, the study units replace the university lecturer. This is one of the great advantages of distance learning; you can read and work through specially designed study materials at your own pace, and at a time and place that suit you best. Think of it as reading the lecture instead of listening to a lecturer. In the same way that a lecturer might set you some reading to do, the study units tell you when to read your textbooks or other material. Just as a lecturer might give you an in-class exercise, your study units provide exercises for you to do at appropriate points.

Each of the study units follows a common format. The first item is an introduction to the subject matter of the unit and how a particular unit is integrated with the other units and the course as a whole. Next is a set of learning objectives. These objectives enable you to know what you should be able to do by the time you have completed the unit. You should use these objectives to guide your study. When you have finished the units, you must go back and check whether you have achieved the objectives. If you make a habit of doing this, you will significantly improve your chances of passing the course.

Remember that your tutor's job is to assist you. When you need help, do not hesitate to call and ask your tutor to provide it.

1. Read this Course Guide thoroughly.

2. Organise a study schedule. Refer to the 'Course Overview' for more details. Note the time you are expected to spend on each unit and how the assignments relate to the units. Whatever method you chose to use, you should decide on it and write in your own dates for working on each unit.

3. Once you have created your own study schedule, do everything you can to stick to it. The major reason that students fail is that they lag behind in their course work.

4. Turn to Unit 1 and read the introduction and the objectives for the unit.

5. Assemble the study materials. Information about what you need for a unit is given in the 'Overview' at the beginning of each unit. You will almost always need both the study unit you are working on and one of your set of books on your desk at the same time.

6. Work through the unit. The content of the unit itself has been arranged to provide a sequence for you to follow. As you work through the unit, you will be instructed to read sections from your set books or other articles. Use the unit to guide your reading.

7. Review the objectives for each study unit to confirm that you have achieved them. If you feel unsure about any of the objectives, review the study material or consult your tutor.

8. When you are confident that you have achieved a unit's objectives, you can then start on the next unit. Proceed unit by unit through the course and try to pace your study so that you keep yourself on schedule.

9. When you have submitted an assignment to your tutor for marking, do not wait

   for its return before starting on the next unit. Keep to your schedule. When the assignment is returned, pay particular attention to your tutor's comments, both on the tutor-marked assignment form and written on the assignment. Consult your tutor as soon as possible if you have any questions or problems.

10. After completing the last unit, review the course and prepare yourself for the final examination. Check that you have achieved the unit objectives (listed at the beginning of each unit) and the course objectives (listed in this Course Guide).

FACILITATION/TUTORS AND TUTORIALS

There are 12 hours of tutorials provided in support of this course. You will be notified of the dates, times and location of these tutorials, together with the name and phone number of your tutor, as soon as you are allocated a tutorial group. Do not hesitate to contact your tutor by telephone, e-mail, or discussion board if you need help. You will definitely benefit a lot by doing that. Contact your tutor if:

- you do not understand any part of the study units or the assigned readings
- you have difficulty with the self-tests or exercises
- you have a question or problem with an assignment, with your tutor's comments on an assignment or with the grading of an assignment.

You should make an effort to attend the tutorials. Thus, it is the only opportunity you have to enjoy face-to-face contact with your tutor and to

ask questions which are answered instantly. You can raise any problem encountered in the course of your study. To gain the maximum benefit from course tutorials, prepare a question list before attending them. You will learn a lot from participating in discussion actively.

SUMMARY

CIT 322: Introduction to Internet Programming introduces you to basic principles, concepts and features of Internet technologies in addition to the skills for developing Web applications. The skills you need to understand the basics of Internet programming are intended to be acquired in this course. The content of the course material was planned and written to ensure that you acquire the proper knowledge and skills for the appropriate situations. Some real-life problems were mentioned or solved for you to apply. The essence is to help you acquire the necessary knowledge and competence.

I wish you success with the course and hope that you will find it
both interesting and useful.

MAIN COURSE

CONTENTS **PAGE**

MODULE 1

Unit 1	Definitions and How to Connect to the Internet
Unit 2	Internet Services, Communication and Protocol
Unit 3	Network Model and Web Application Development

UNIT 1 DEFINITIONS AND HOW TO CONNECT TO THE INTERNET

CONTENTS

1.0 INTRODUCTION

The Internet is a global network of networks with a massive store of multimedia and shared information. It seems to be everywhere and allows many people and devices to connect to it via phone line, cable, digital subscriber lines or wireless. In this unit, we shall explain the meaning of the Internet and discuss the various means of connection to the Internet.

2.0 OBJECTIVES

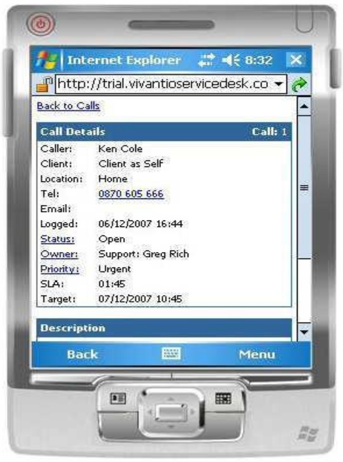
At the end of this unit, you should be able to:

- define and discuss the evolution of the Internet
- explain the meaning of intranet and extranet
- list the devices use to access the Internet
- discuss the various means of accessing the Internet.

3.0 MAIN CONTENT

3.1 The Definition and Evolution of the Internet

The Internet also referred to as the net, in simplest terms, consists of large a group of millions of computers around the world that are connected to one another for the purpose of communication. It is a network of networks that consists of millions of private, public, academic, business, and government networks, local to global scope, that are linked by a broad array of electronic, wireless and optical networking technologies such as phone lines, fibre optic lines, coaxial cable, satellites, and wireless connections. The Internet seems to be everywhere today with many people and devices connected to it. When connected to the Internet people can access services such as online shopping, listen to radio and TV broadcast, chat, and send mail, access information, read newspaper and so on. Today Internet is not only accessed from regular stationary computer but also from mobile / portable devices such as Personal Digital Assistants (PDAs) as seen in figure 1.1, cell phones, netbook, iPod, iPad, Palm Pilots and others.



http://trial.vivantioservicedesk.co

Call Details Call: 1

Caller: Ken Cole
Client: Client as Self
Location: Home
Tel: 0870 605 666
Email:
Logged: 06/12/2007 16:44
Status: Open
Owner: Support: Greg Rich
Priority: Urgent
SLA: 01:45
Target: 07/12/2007 10:45

Fig. 1.1: PDA access to the Internet

The Internet originated as a proposal from the Advanced Research Project Agency (ARPA). The idea was to see how computers connected in a network i.e. (ARPANET) could be used to access information from research facilities and universities. In 1969, four computers (located at UCLA, Stanford Research Institute, University of California Santa Barbara and the University of Utah) were successfully connected. As time went on, other networks were connected. With four nodes by the end of 1969, the ARPANET spanned the continental United States (US) by 1971 and had connections to Europe by 1973. Though the Interconnected Network, or Internet, was originally limited to the military, government, research, and educational purposes it was eventually opened to the public. Today there are hundreds of millions of computers and other devices connected to the Internet worldwide.

Other definitions that are closely related to the term Internet are intranet and extranet.

Intranet

The term “Intranet” is used to describe a network of personal computers (PC) without any personal computers on the network connected to the world outside of the Intranet. The Intranet resides behind a firewall; if it allows access from the Internet, it becomes an Extranet. The firewall helps to control access between the intranet and Internet so that only authorised users will have access to the Intranet. Usually these people are members of the same company or organisation. Like the Internet itself, intranets are used to share information. Secure intranets are now the fastest-growing segment of the Internet because they are much less expensive to build and manage than private network based on proprietary protocols.

Extranet

Extranets are becoming a very popular means for business partners to exchange information. An Extranet is a term used to refer to an intranet that is partially accessible to authorised outsiders. Privacy and security are important issues in extranet use. A firewall is usually provided to help control access between the Intranet and Internet. In this case, the actual server will reside behind a firewall. The level of access can be set to different levels for individuals or groups of outside users.

3.2 Internet Access

In order to have access to the vast resources on the Internet, you need to connect your computer to a computer system that is already on the Internet, usually one run by an Internet Service Provider (ISP). There are four major ways of connecting a client (user) computer to the vast resources on the Internet; these are by a dial-up connection using a telephone line or an Integrated Services Digital Network (ISDN), a Digital Subscriber Line (DSL), a cable TV connection or a satellite connection. While rural users may consider installing a satellite dish for Internet connections, urban users may have access to wireless connections. In most offices, users connect their computers via a local area network (LAN) connected to the Internet. Similarly, in many home, users are beginning to connect their computers into Internet-connected LANs, too. The Dial-up access gives a low speed connection to the Internet. High-speed Internet connections, which include DSL, ISDN, leased lines, cable Internet, and satellite, are called broadband connections.

3.2.1 Word Wide Web (WWW)

The WWW is a system of interlinked hypertext documents accessed via the Internet. The web browser is used to access the web pages that may contain text, images, videos, and other multimedia pages. The WWW use a protocol called HTTP-hypertext transfer protocol.

The HTTP defines how messages are formatted and transmitted, and the actions the web servers and other browsers should take in response to the various commands are also determined by the HTTP.

There are two types of website called the static and dynamic pages. The static website/page is a one that has web pages on the server in the format that is sent to a client web browser which is popularly called the front end design. The dynamic website is one that changes or customizes itself automatically basic on certain criteria which is equally called backend design.

3.2.2 Electronic Mail (Email)

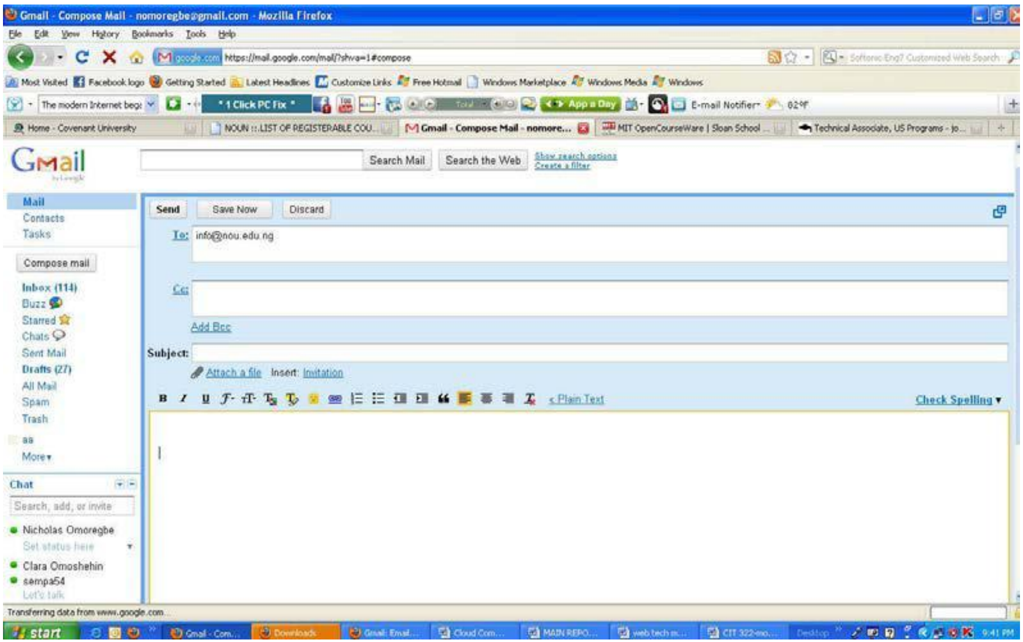
The electronic mail is one of the fastest ways of sending information from one region to another via the computer and other electronic devices. You can communicate with one person at a time or thousands; you can receive and send files and other information. The basic email functions include send and receive email messages, save your message in a file, print mail messages, reply to mail messages and attach a file to a mail message.

There are a number of webmail providers but the most popular ones are gmail, yahoo mail and Hotmail.

An e-mail message consists of three components namely:

(i) the message header (ii) the message envelop, and (iii) the message body.

The message header contains control information, including, minimally, an originator's email address and one or more recipient addresses. Usually descriptive information is also added, such as a subject header field and a message submission date/time stamp. The message body carries the data to be sent. The message's body property usually contains details associated with the message. In addition to the data part, messages carry details that assist in distinguishing messages and selectively receiving them. This detail is made up of a fixed number of fields, which is referred to as the message envelope. These fields are source destination tag communicator. To use email, you should have an email address, which is created by an Internet Service Provider or on a Website such as yahoo, Google, and hotmail. Most e-mail addresses are set up in this manner: your username, followed by "@" (at) symbol, and then a domain name (for instance, .com, .edu.,.net, or .org). When you send e-mail to others, Simple Mail Transfer Protocol (SMTP) is used. When you receive e- mail, Post Office Protocol (POP, currently POP3) and Internet Message Access Protocol (IMAP) can be used.



To: info@nou.edu.ng

Fig. 1.2: Google Mail Interface

3.2.3 File Transfer Protocol (FTP)

FTP is protocol for exchanging files over the Internet used. It is used for moving files between hosts on a TCP/IP network. FTP is most commonly used to download a file from a server using the Internet or to upload a file to a server as seen in figure 1.2

Figure 1.3 FTP

To achieve FTP, the user invokes one of two commands:

- **get**: the command for transferring a file from another server to you own computer
- **put**: the command for moving a file from your computer to another one.

3.2.4 Search Engine

A search engine is designed to search for information on the Internet. This presents the results in the form of a search results list. The search results can be qweb pages, images, videos, and other type of files. Examples include Google, Bing, and Mamma among others. One of the popular images is seen in figure 1.3.

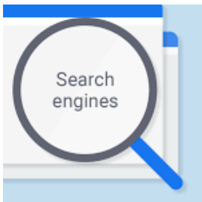


Figure 1.4 Search Engine

3.2.5 Chatting

This is the other method of Internet conversation which enables the connection of people anywhere on the Internet.

Chart sessions allow many users to join in the same free-form conversation which is usually within a discussion topic. Examples of the charting software are MSN messengers, Yahoo messengers, IRC, pidgin, WhatsApp among others.

3.2.6 Video Conferencing

A video conferencing is a set of interactive telecommunication technologies which allow two or more locations to interact via two-way video and audio transmission simultaneously using the Internet. Examples include zoom, video WhatsApp, Google meet among others.

3.2.7 E-Commerce

E-commerce also called electronic commerce is the buying and selling of products or services over electronic systems such as the Internet. Some of the common applications which are used in e-commerce are domestic and international payment systems, group buying, automatic online assistants, online shopping and other tracking, online banking, shopping cart software among other terms. Example of the online marketing sites are amoze.com, ebay.com, aliexpress.com, konga.com, jumia.com among others as seen in figure 1.4

Figure 1.5 logo of some e-commerce sites

3.2.8 Dial-up Connection

Dial-up Internet access is a form of Internet access that uses the facilities of the public switched telephone network (PSTN) to establish a dialed connection to an Internet service provider (ISP) via telephone lines. The user's computer or router uses an attached modem to encode and decode Internet Protocol packets and control information into and from analog audio frequency signals, respectively. The term "Dial-up Internet access" was coined during the early days of computer telecommunications when modems were needed to connect terminals or computers running terminal emulator software to mainframes, minicomputers, online services and bulletin board systems via a telephone line. To use a dial-up account, you need a modem. A modem (modulator-demodulator) is a device that modulates an analog carrier signal to encode digital information, and demodulates such a carrier signal to decode the transmitted information. To distinguish dial-up modems from newer, high-speed modems, they are could also be called analog modems or dial-up modems. Most computers come with an internal modem and most ISPs support modems at speeds of 28.8 kilobits per second (Kbps) and 56 Kbps. With dial-up, you connect only when you want to use Internet services and disconnect (hang up) when you are done. This type of data transmission is similar to using the telephone to make a call. The client computer modem dials the preprogrammed phone number for a user’s Internet Service Provider (ISP) and connects to one of the ISP’s modems. Once the ISP has verified the user's account, a connection is established and data can be transmitted. The communication ends when either modem hangs up. Dial-up connections is not expensive (it costs no more than a local telephone call) but the speed is usually low at about 28kps – 46kps because of the limitations of analog phone lines and telephone company switches.

3.2.9 ISDN

Integrated Services Digital Network (ISDN) is a set of communications standards for simultaneous digital transmission of voice, video, data, and other network services over the traditional circuits of the public switched telephone network. It allows dial up into the Internet at speeds ranging from 64 to 128 kbps. For this connection to be available, telephone companies would have to install special ISDN digital switching equipment. The ISDN service intended for residential use is Basic Rate Interface (BRI). On one ISDN line, BRI provides two 64-Kbps channels, or B (bearer) channels, and one 16-Kbps channel, or D(data) channel. The D channel is mostly used for signalling such as to indicate that the line is busy. The B channels are where the action is. Two B channels can be combined to have a 128-Kbps line to the

Internet. This is roughly twice the speed of the fastest analogue modem, 56 Kbps. To connect to your ISP via ISDN you need to confirm the availability of the access and this will require you to have an ISDN adapter. ISDN lines are more expensive than normal phone lines, so the telephone rates are usually higher.

3.2.10 Cable TV Connection

This is a connection made to the Internet via a Cable TV modem. The modem is designed to operate over cable TV lines. Since the coaxial cable used by cable TV provides much greater bandwidth than telephone lines, a cable modem can be used to achieve extremely fast speed as high as 128 kbps to 10 mbps to the World Wide Web . This combined with the fact that millions of homes are already wired for cable TV in developed countries has made the cable modem something of a holy grail for Internet and cable TV companies. The services offered are usually at low cost for unlimited, “always connected” access. However, there are a number of technical difficulties in this type of connection. The problem is that the cable network was designed to move information in one direction, from the broadcaster to the user. Downstream speeds have been very impressive such that the line can theoretically bring you data as fast as 30 Mbps but upstream speed depends on line quality. The Internet, however, is a two-way system where data also need to flow from the client to the server. In addition, it is still unknown whether the cable TV networks can handle the traffic that would ensue if millions of users began using the system for Internet access. Large cable companies are spending money to upgrade their networks to Hybrid Fiber-Coaxial (HFC) to handle two-way traffic better. Smaller providers cannot afford the upgrade, so they have to use a phone line at 28.8 Kbps for upstream data. Another issue bothers on security and the need to either share or not share files amongst users.

3.2.11 Digital Satellite Connection

Digital Satellite Systems (DSS), or direct broadcast satellite, allows one to get Internet information via satellite. Satellite Internet systems are an excellent, although rather costly, option for people in rural areas where Digital Subscriber Line (DSL) and cable modem connections are not available. A satellite installation can be used even where the most basic utilities may be lacking, if there is a generator or battery power supply that can produce enough electricity to run a desktop computer system. The two-way satellite Internet option offers an always-on connection that bypasses the dial-up process. In a two-way satellite Internet connection, the upstream data is usually sent at a slower speed than the downstream data arrives. Thus, the connection is asymmetric. A dish antenna, measuring about two feet high by three feet wide by three feet deep, transmits and receives signals. Uplink speeds are nominally 50 to 150 Kbps for a subscriber using a single computer. The downlink occurs at speeds ranging from about 150 Kbps to more than 1200 Kbps, depending on factors such as Internet traffic, the capacity of the server. The main advantage of the Satellite technology over cable modems and DSL is accessibility. Satellite connections are faster than dial up and ISDN. Although it is not as fast as cable modems or DSL services, which both can provide more than megabits of bandwidth. In addition, cable and DSL access methods are cheaper. Figure 1.5 shows a Satellite connection to the Internet.

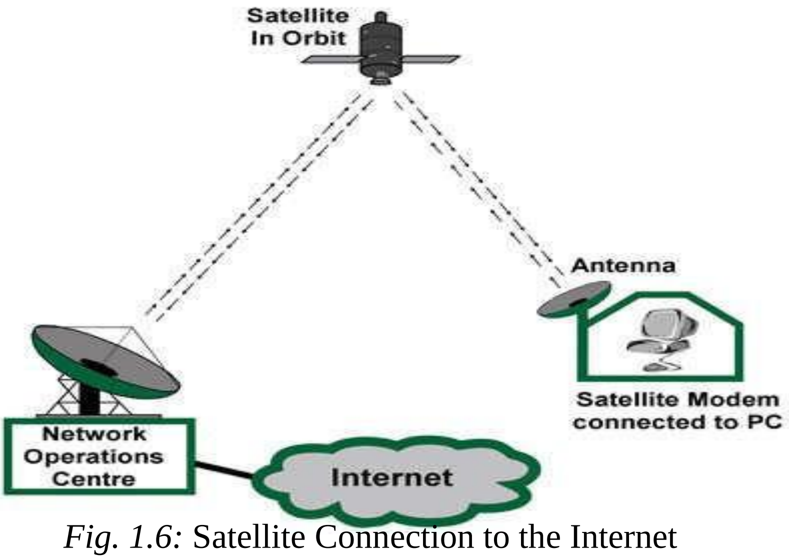


Fig. 1.6: Satellite Connection to the Internet

Equipment required for satellite connection includes installation of a mini-dish satellite receiver and a satellite modem. Satellite systems are also prone to rain fade (degradation during heavy precipitation) and occasional brief periods of solar interference.

4.0 CONCLUSION

The Internet has remained a dominant means of communication over the past decade. It represents one of the most remarkable developments in the technological history of the world. It began as a medium for exchanging files by academia and has become a nearly ubiquitous phenomenon that has transformed almost every aspect of daily life. The Internet has made information available in a quick and easy manner, publicly accessible and within easy reach via the connection's infrastructure discussed in this unit. In the next unit, we shall look at some of the services available on the Internet and the enabling protocols.

5.0 SUMMARY

The general rule about the Internet connection is "the faster, the better." The bandwidth and transfer rate determine how quickly pictures, sounds, animation and video clips will be downloaded. Since multimedia and interactivity make the Internet such an exciting tool for information sharing, the speed is the key. Dial-up access provides an easy and inexpensive way for users to connect to the Internet, however, it is a slow-speed technology and most users are no longer satisfied with dial-up or ISDN connections. The Internet has also made buying and selling very easy. Fortunately, the broadband access, we once dreamed of, is now possible with TV cable, DSL and satellite links.

6.0 TUTOR-MARKED ASSIGNMENT

i. Briefly explain the origin of the Internet.
ii. List three examples of mobile devices that can be used to access the Internet.
iii. Discuss the four major ways of connecting a client computer to the vast resources on the Internet.

7.0 REFERENCES/FURTHER READING

Alex, L. & Mathew, L. (1999). *Fundamentals of Information Technology*. Ne Delhi: Vikas Publishing House PVT LTD.

Andy, S. (1999).*Computer Communications, Principles and Business Applications*. (2nd ed.). England: McGraw-Hill.

Behrouz, A. F. (2003). *Data Communications and Networking*. (3rd ed.). N.Y, USA: McGraw-Hill/Osborne.

Brian, *et al*.(1999). *Using Information Technology: A Practical Introduction to Computers and Communication*. Irwin/McGraw-Hill.

Terry, F-M. (2009). *Web Development and Design Foundations with XHTML*. USA: Pearson.

Katarzyna J. M. "Internet and You: Connecting to the Internet." Retrieved from http://www.rsna.org/Technology/internet2-1.cfm

UNIT 2 INTERNET SERVICES, COMMUNICATION AND PROTOCOL

CONTENTS

1.0 INTRODUCTION

The web and the Internet deals with all the preliminaries that you are expected to know before in the design and development of the static and dynamic pages. This include the network and the various types, web and browsers, servers and various types and hosting.

2.0 OBJECTIVES

At the end of this unit, you should be able to:

- list categories and characteristics of Hosting services
- define URL and the purposes
- explain ICANN and functions?
- List common domain name registrar
- explain what command is used to access domain name registration record of a registrant?

3.0 MAIN CONTENT

3.1 Basic concepts in computer networking and technology

A computer network is a connection of computers that may be in various forms. Over the years, the networking industry has come up with terms such LAN an attempt to define working categories for major types of network designs. The following concepts have been in use in implementing networking technology:

3.1.1 Local Area Network (LAN): A LAN connects network devices over a relatively short distance. A networked office building, school, or home usually contains a single LAN, though sometimes one building will contain a few small LANs, and occasionally a LAN will span a group of nearby buildings. LANs are typically owned, controlled, and managed by a single person or organization.

3.1.2 Wide Area Network (WAN): A WAN is a geographically-dispersed collection of LANs. A network device called a router connects LANs to a WAN. WANs differ from LANs in several important ways. Like the Internet, most WANs are not owned by any one organization but rather exist under collective or distributed ownership and management. Others network arrangements may be MAN, SAN, PAN, DAN and CAN

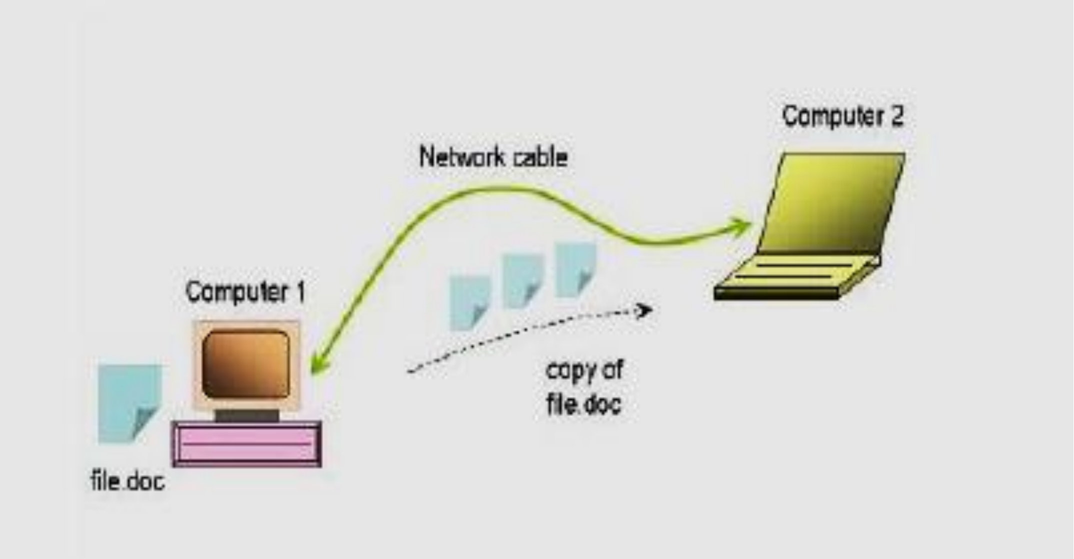


Figure 2.1 A Computer network for file sharing

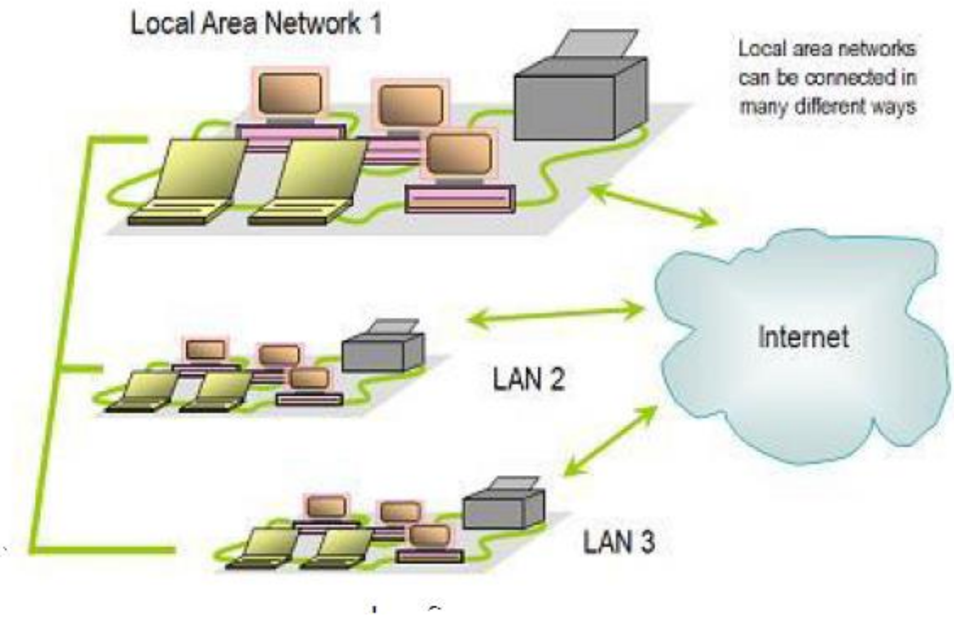


Figure 2.2 A Local Area Network environment

3.2 The Web and the Internet

The **Internet** is a global system of interconnected computer networks that use the standard Internet Protocol Suite **(TCP/IP)** to serve billions of users worldwide. It is a *network of networks* that consists of millions of private, public academic, business, and government networks.

The **World Wide Web (WWW)** otherwise known as the Web is a system of interlinked hypertext documents accessed via the Internet

3.2. 1 Web Browsers and Servers

A **web browser** is the program used to view pages and navigate the World Wide Web. A wide array of web browsers is available for almost every platform you can imagine. Microsoft Internet Explorer, for example, is included with Windows and Safari is included with Mac OS X. Mozilla Firefox, Netscape Navigator, and Opera are all available for free.

A **web server** is the program that runs on a computer and is responsible for replying to web browser requests for files. You need a web server to publish documents on the Web. When you use a browser to request a page on a website, that browser makes a web connection to a server using the HTTP protocol. The browser then formats the information it got from the server. Server accepts the connection, sends the contents of the requested files and then closes.

3.3 **Client/Server Model in Internet Technologies**

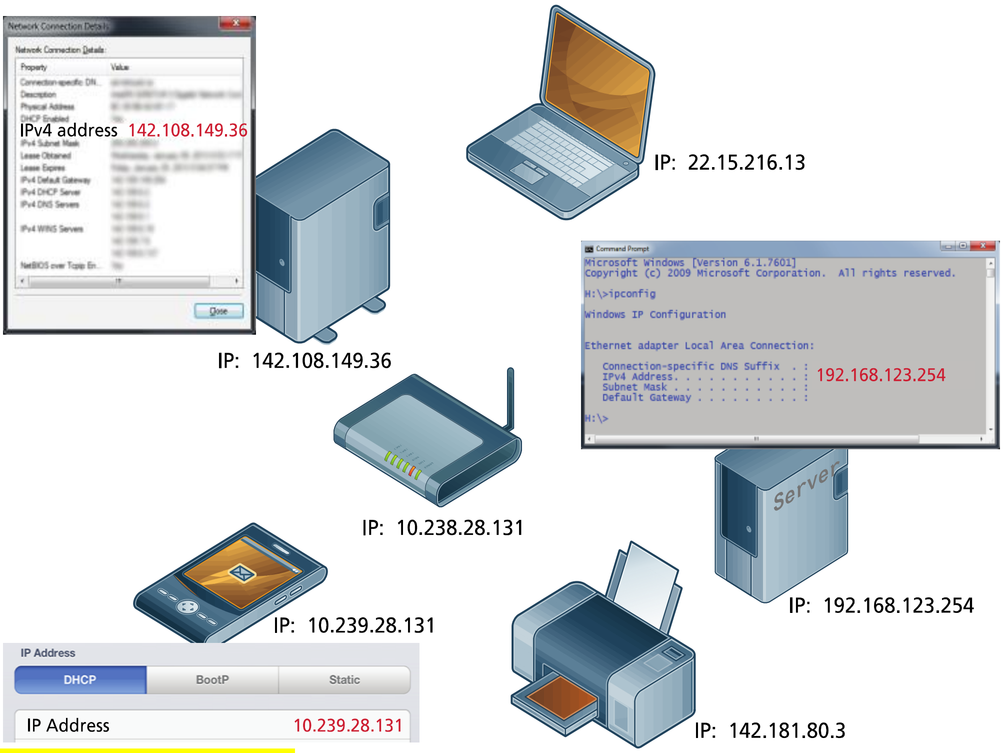
The **client-server model** describes how a server provides resources and services to one or more clients. Examples of servers include

web servers
mail servers
database servers
and file servers.

Each of these servers provide resources to client devices, such as desktop computers, laptops, tablets, and smartphones.
Most servers have a one-to-many relationship with clients, meaning a single server can provide resources to multiple clients at one time.

3.4 Internet Protocol (IP) and Domain Name System

A computer is hardwired to obey a list of built-in commands, rules, and standards, known as **protocols**, to communicate and identify with other computers, or connect to the internet. One such protocol is known as the **Internet Protocol.** This can be illustrated in the figure 2.3



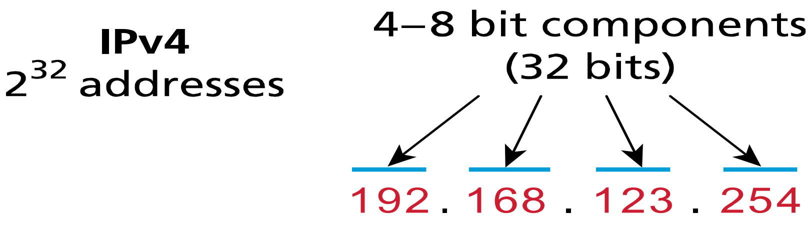
IPv4 address 142.108.149.36

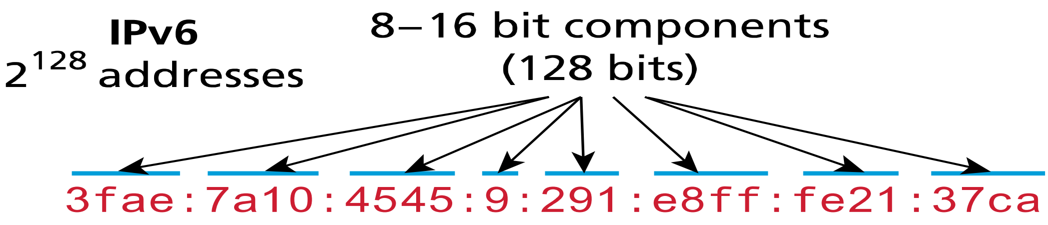
```
Microsoft Windows [Version 6.1.7601]
Copyright (c) 2009 Microsoft Corporation.  All rights reserved.

H:\>ipconfig

Windows IP Configuration


Ethernet adapter Local Area Connection:

   Connection-specific DNS Suffix  . :
   IPv4 Address. . . . . . . . . . . : 192.168.123.254
   Subnet Mask . . . . . . . . . . . :
   Default Gateway . . . . . . . . . :

H:\>
```

IP Address 10.239.28.131

Figure 2.3 Internet Protocol

Types of Internet Protocol Address



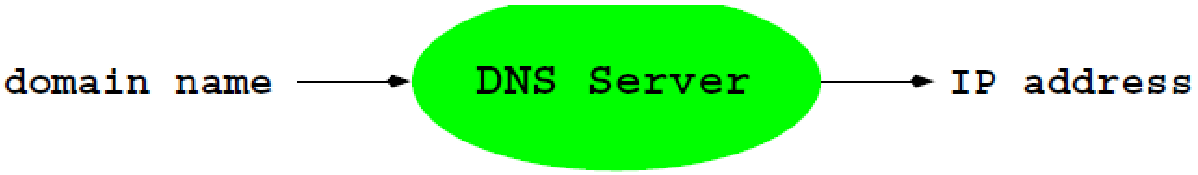


Domain Name System

Every host on the Internet has a unique **IP address** and a **domain name**.

The **network name space** is the set of all host names and that changes dynamically with time due to addition/deletion of hosts, regrouping of local work groups, reconfiguration of subparts of the network, maintenance of systems and networks, and so on.

The **domain name system (DNS**) provides a distributed database service that supports dynamic update and retrieval of information contained in the name space. The fig.2.4 below illustrates the relationship of domain name and IP



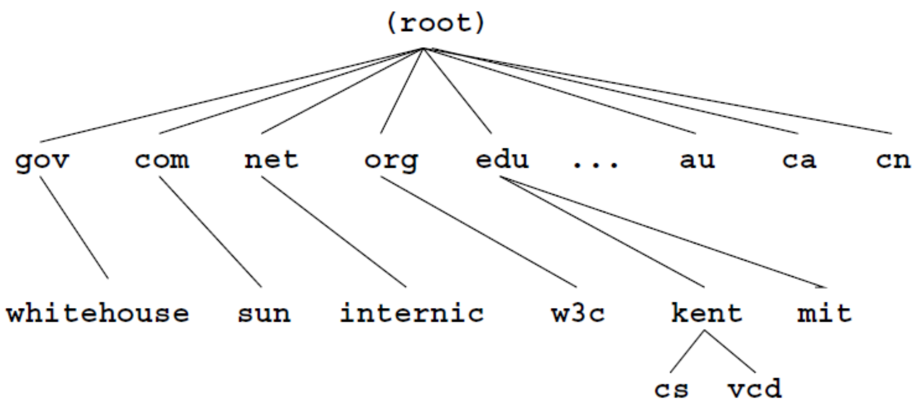


Figure 2.4 The Domain Name Hierarchy

3.5 Web Hosting and Domain Registration

Web Hosting is a service to store and serve ready-made files and programs so they are accessible on the Web. Hence publishing on the web involves

- designing and constructing the pages and writing the programs for a website
- placing the completed website with a hosting service

To host a site under a given domain name, a hosting service associates that domain name to an IP number assigned to the hosted site. The **domain-to-IP** association is made through domain name servers (DNS) managed by the hosting service.

Domain Name Registration

To obtain a domain name you need the service of a **domain name registrar**. Avery modest yearly fee is usually charged for new domain name registration. Once registered, the domain name is property belonging to the **registrant.**
ICANN accredits commercial registrars for common Top Level Domains(TLDs) including .com, .net, .org. New TLDs being added include .biz, .info, .pro, .aero, .name, and .museum.
The registration record of a domain name is publicly available. The standard Internet **WHOIS** service allows easy access to this information. WHOIS is provided by the **whois** command:

whois *domain name*

3.6 Content Types and the Web

Many different types of files can be placed and retrieved on the web. The Web server and Web browser use a set of standard designations to indicate file content types in order to process different files correctly. Many popular types are associated with standard file extensions. as indicated in the table below:

3.6.1 Content Types and Suffixes

Table 2.1 content types and suffixes

Content Types	Suffixes
text/html	html htm
image/jpeg	jpeg jpg jpe
audio/basic	au snd
audio/mpeg	mpeg mp2 mp3
audio/x-realaudio	ra
audio/x-wav	wav
video/mpeg	mpeg mpg mpe
video/quicktime	qt mov

3.6.2 The Web

There is no central control or administration for the Web. Anyone can potentially put material or content on the Web and retrieve information from it. The Web consists of a vast collection of documents that are located on computers throughout the world. These documents are created by academic, professional, governmental, and commercial organizations as well as by individuals.

As the Web grows explosively, MIT of the USA and INRIA (the French National Institute for Research in Computer Science and Control) have agreed to become joint hosts of the W3 Consortium which is supported by industry and will further develop Web related standards, protocols, and services.

The Web uses **Uniform Resource Locators (URLs)** to identify (locate) resources (files and services) available on the Internet. A URL may identify a host, a server port, and the target file stored on that host. URLs are used, for example, by browsers to retrieve information and by HTML to link to other resources.

3.6.3 Websites and Web Development Process

A **website** or **web** site is a collection of related network **web** resources, such as **web pages**, multimedia content, which are typically identified with a common domain name, and published on at least one **web** server. They may incorporate elements from other **websites** with suitable markup anchors.

Web Development Process

The core part of website development and design is not necessarily about the coding process using different available tools. The crucial part of web development is the life cycle which involve different stages.

Website Development Life Cycle

STEP-1. **Gathering Information: Purpose, Main Goals, and Target Audience**
STEP-2. **Planning: Sitemap and Wireframe Creation**
STEP-3. **Design: Page Layouts, Review, and Approval Cycle**
STEP-4. **Content Writing and Assembly**
STEP-5. **Coding**
STEP-6. **Testing, Review, and Launch**
STEP-7. **Maintenance: Opinion Monitoring and Regular Updating**

4.0 CONCLUSION

A LAN connects network devices over a relatively short distance. A networked office building, school, or home usually contains a single LAN, though sometimes one building will contain a few small LANs, and occasionally a LAN will span a group of nearby buildings. LANs are typically owned, controlled, and managed by a single person or organization. Similarly, a WAN is a geographically-dispersed collection of

LANs. A network device called a router connects LANs to a WAN. WANs differ from LANs in several important ways. Like the Internet, most WANs are not owned by any one organization but rather exist under collective or distributed ownership and management. Others network arrangements may be MAN, SAN, PAN, DAN and CAN.

5.0 SUMMARY

The unit successfully explained the concepts of the network such as the LAN, WAN and other types as applied. The **Internet** is a global system of interconnected computer networks that use the standard Internet Protocol Suite **(TCP/IP)** to serve billions of users worldwide. It is a *network of networks* that consists of millions of private, public academic, business, and government networks.

A web browser is the program used to view pages and navigate the World Wide Web. A wide array of web browsers is available for almost every platform you can imagine. Microsoft Internet Explorer, for example, is included with Windows and Safari is included with Mac OS X. Mozilla Firefox, Netscape Navigator, and Opera are all available for free. A web server is the program that runs on a computer and is responsible for replying to web browser requests for files. You need a web server to publish documents on the Web. When you use a browser to request a page on a website, that browser makes a web connection to a server using the HTTP protocol. The browser then formats the information it got from the server. Server accepts the connection, sends the contents of the requested files and then closes.

6.0 TUTOR MARKED ASSIGNMENT

1. List categories and characteristics of Hosting services
2. Define URL and the purposes
3. What is ICANN and functions?
4. List common domain name registrar
5. What command is used to access domain name registration record of a registrant?

7.0 REFERENCES/FURTHER READING

Alex, L. & Mathew, L. (1999). *Fundamentals of Information Technology*. New Delhi: Vikas Publishing House PVT LTD.

Andy, S. (1999).*Computer Communications, Principles and Business Applications*. (2nd ed.). England: McGraw-Hill.

Behrouz, A. F. (2003). *Data Communications and Networking*. (3rd ed.). N.Y, USA: McGraw-Hill/Osborne.

Brian, *et al*.(1999). *Using Information Technology: A Practical Introduction to Computers and Communication*. Irwin/McGraw-Hill.

Terry, F-M. (2009). *Web Development and Design Foundations with XHTML*. USA: Pearson .

Katarzyna J. M. “Internet and You: Connecting to the Internet.”
Retrieved from http://www.rsna.org/Technology/internet2-1.cfm

Idowu, S.A., Maitanmi, S.O. & Adetunji, O. O. (2020). Introductory to Web Technology and Development. Nigeria: Jamiro Press.
June, C. (2003). *The Unusually Useful Web Book*. USA: New Ride.

UNIT 3 NETWORK MODEL AND WEB APPLICATION DEVELOPMENT

CONTENTS

1.0 INTRODUCTION

A client may be a program running on the local machine requesting service from server. A client program is started by the user or another application program and terminates when the service is complete**.** A server – can sometimes be a program running on the remote machine providing service to the clients. When it starts, it opens the door for incoming request from clients, but it never initiates a service until it is requested to do so.

A network of networks or “Internet” refers to a group of two or more networks that are interconnected and physically capable of communication, share data and act together as a single network. Machine on one network can communicate with machines on other networks, and data, file and other information back and forth. For this to work, the systems must follow some set of rules or protocols. This is a “language” or software that enables different types of machines on separate network to communicate and exchange information. The Internet uses the TCP/IP protocol. The Internet offers access to data, graphics, sound, software, text, and people through a variety of services and tools for communications and data exchange. Some services available on the Internet are as follows:

- Remote login (telnet)
- File transfer (ftp)
- Electronic mail (e-mail)
- News (USENET or network news)
- Hypertext (www)

2.0 OBJECTIVES

At the end of this unit, you should be able to:

- discuss the client-server model
- describe the Web application architecture
- explain the meaning of Common Gateway Interface.

3.0 MAIN CONTENT

3.1 The Client Server Model

The term "client / server" dates back from (1980's) and refers to personal computers joined by a network. Client/server can also describe a relationship between two computer programs- the client and the server. The client/server technology evolved as a result of downsizing of mainframe applications and upsizing of microcomputer applications.

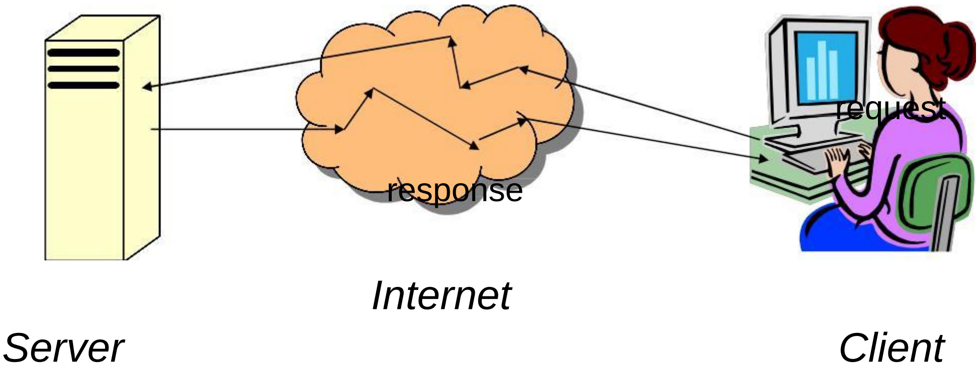


Fig. 3.1: Client-server Model

The client requests some type of service (such as file or database access) from the server. The server fulfils the request and transmits the results to the client over a network. While both the client and the server programs can reside on the same computer, typically, they run on different computers. Specific types of clients include web browsers, e-mail clients, and online chat clients. Specific types of servers include web servers, ftp servers, application servers, database servers, name servers, mail servers, file servers, print servers, and terminal servers. Most web services are also types of servers. Where a server is made to handle request from multiple clients and transaction processing is done on both the server and the client we have distributed processing. Distributed processing involves the storage of data on database servers called back-ends from where clients' applications called front-ends access the data needed for their operations. In addition, the client (front–end) does data presentation and or processing, while the server (back-end) does storage, security and major data processing. The client / server inter-relationship is given in terms of layers and tiers. The Internet is a great example of client / server architecture at work. Consider a scenario where a user accesses his bank from a location away from the bank's computer. The user will require a web browser client to send a request to a web server at a bank. That program may in turn forward the request to its own database client program that sends a request to a database server at another bank computer to retrieve the account information. The balance is returned to the bank database client, which in turn sends it back to the web browser client displaying the results to the user. The client–server model has become one of the central ideas of network computing. Many business applications being written today use the client–server model. So do the Internet's main application protocols, such as HTTP, SMTP, Telnet, and DNS.

3.2 Types of Server

Servers are usually high-performance computers connected to the Internet by high-speed communication lines. Depending on your application, you may deploy it on less-powered machine with less substantial connections. The following are variation of servers:

a) **Web server:** This is use to store and deliver the elements of web pages.

Application server: This is used to run specialised Internet application, such as e-commerce or e-health's engine. It is designed to process requests and deliver dynamic results.

b) **Streaming server:** This is used to deliver audio or video to the visitors to a site real-time.

c) **Mail server:** This is used to send and receive e-mail.

d) **Name server:** This is a specialised server that stores huge directories of web servers. It keeps track of all the registered domain names on the Internet.

e) **Secure server:** This is a Web server that encrypts data before transmitting it, to prevent unauthorised access. They are commonly used to secure for financial transactions in the Internet.

3.3 Web Application (Webapps)

Webapps are applications that are accessed with a web browser over a network such as the Internet or an intranet. They are popular because of the ubiquity of the browser as a client (thin client). Similarly, its popularity is equally due to the possibility of updating and maintaining the application without necessarily distributing and installing it on every available client. Webapps or web applications as they are sometimes called are used to implement webmail, online retail sales, online auctions, discussion boards, and weblogs and so on. Web developers often use client-side scripting to add functionality to the webapps by creating an interactive site that does not require page reloading. Webapps generate a series of web pages dynamically in a standard format such as

Hypertext Markup Language (HTML) supported by common browsers. Through Java, JavaScript, Flash and other technologies, application specific methods such as drawing on the screen, playing audio and accessing the keyboard and mouse are all possible. Webapps are the present and the future of business transactions. The Web is based on the client/server architecture. That is, both the server and the client application are responsible for some sort of processing. Web application is commonly structured as a 3-tier application. The web browser constitutes the first tier, a middleware engine using some dynamic web content technology such as: Common Gateway Interface (CGI), Hypertext Preprocessor (PHP), Java Servlets or Java Server Pages (JSP) or Active Server Pages (ASP) constitutes the middle-tier and the database is the third tier or back-end. The backend applications include MySQL, SQL Server, Oracle, etc. The bulk of online transactions take place between the middleware and the database server. While the middle-ware is responsible for the business logic transaction processing, the back-end is responsible for information storage and retrieval from the database.

The middle-tier may be multi-tiered. That is, it can be composed of several other servers with designated responsibilities, hence the over-all architecture is said to be N-tier. A fundamental rule in 3-tier architecture is that the client has no direct line of communication with the data tier. That is, all communications are routed through the middleware tier.

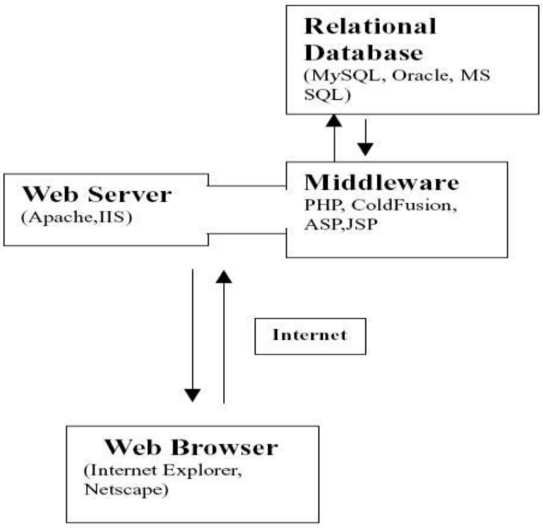


Fig. 3.2: Architecture of a Web Application

3.3.1 The Web Browser

Client-side refers to operations that are performed by the client in a client–server relationship in a computer network. Typically, a client is a computer application, such as a web browser, that runs on a user's local computer or workstation and connects to a server as necessary. Operations may be performed client-side because they require access to information or functionality that is available on the client but not on the server. Programs that run on a user's local computer without ever sending or receiving data over a network are not considered clients, and so the operations of such programs would not be considered client-side operations. The Web browser constitutes the client. It is a software application that enables a user to display and interact with text, images and other information that are located on the web page or a local area network. Browsers can be used to access information on web servers. Examples of web browsers are MS Internet Explorer, Mozilla Firefox, Apple Safari, Netscape and Opera and Google Chrome. Web browsers communicate with web servers using the Hypertext Transfer Protocol (HTTP) to fetch web pages and it allows web browsers to submit information to web servers as well as fetch web pages from them. The primary language of browsers is the HTML, which consists of tags that are used to describe a web page. Most browsers have some level of support for JavaScript and Extensible Markup Language (XML).

3.3.2 The Web Server

A web server can be referred to as either the hardware (the computer) or the software (the computer application) that helps to deliver content that can be accessed through the Internet. The most common use of Web servers is to host Web sites but there are other uses like data storage or for running enterprise applications. The primary function of a web server is to deliver web pages on the request to clients. This means delivery of HTML documents and any additional content that may be included by a document, such as images, style sheets and JavaScript. A client, commonly a web browser or web crawlers, initiates communication by making a request for a specific resource using HTTP and the server responds with the content of that resource or an error message if unable to do so. The resource is typically a real file on the server's secondary memory, but this is not necessarily the case and depends on how the web server is implemented. While the primary function is to serve content, a full implementation of HTTP also includes ways of receiving content from clients. This feature is used for submitting web forms, including uploading of files. Many generic web servers support server-side scripting. The scripting tools used for middleware development include PHP, JSP, ASP, Servlet, PERL, Python, and so on. These tools allow the behaviour of the web server to be scripted in separate files, while the actual server software remains unchanged. Usually, this function used to create HTML document "on-the-fly" as opposed to returning fixed documents. This is referred to as dynamic and static content respectively. The former is primarily used for retrieving and/or modifying information from databases. The latter is, however, typically much faster and more easily cached. Web servers are not only used for serving the world wide web, they can also be found embedded in devices such as printers, routers, webcams and serving only a local network. The web server may then be used as a part of a system for monitoring and/or administrating the device in question. This usually means that no additional software has to be installed on the client computer; since only a web browser is required (which now is included with most operating systems). There are many web server programs available. Table 3.1: Shows a statistics of the market share of the top web servers on the Internet by Netcraft survey in March 2011.

Table 3.1: Popular Servers

Vendor	Product	Web Site Hosted	Percent
Apache	Apache	179,720,332	60.31
Microsoft	IIS	57,644,692	19.34
Igor Sysoev	nginx	22,806,060	7.65
Google	GWS	15,161,530	5.09
Lighttpd	lighttpd	1,796,471	0.60
Sun Microsytemss	SunOne		

Servers are slave programs. They act only when requests are made to them by browsers running on other computers and the Internet. The most commonly used Web Servers are Apache, which has been implemented for variety of computer platforms, and Microsoft's Internet Information Server (IIS), which runs under windows operating systems.

1. Apache HTTP Server

This is most popular web server. It is a free software/open source like Linux, PHP and MySQL.

Apache runs on Unix, Linux, MS Windows, Novell Netware and some other platforms. Apache serves over 68 per cent of websites and serves both static and dynamic contents on the web in a very reliable and secure manner. The name Apache has nothing to do with the Native American tribe of the same name. Rather, it came from the nature of its first version, which was patchy version of the http server. As seen in the usage statistics it is the most widely used server. The primary reasons for this are as follows: It is an excellent server because it is both fast and reliable. Furthermore, it is open-sources software, which means it is free and managed by a large team of volunteers, a process that efficiently and effectively maintains the systems. Finally, it is one of the best available servers for Unix-based systems.

2. The Internet information services (IIS)

The Internet information services (IIS) is a server or system-based services for servers using Microsoft Windows operation system. It is a major component of the Microsoft Server operating system and particularly, a component of its Active Server Pages

(ASPs). IIS is recommended if both the middleware (ASP) and the database Server (SQL Server) are Microsoft products. Though the Apache server may be installed on Windows platforms, it is not the most popular server on those systems. IIS remains the most popular on Windows platform because it is supplied as part of Windows and because it is a reasonable good server. Apache and IIS provide similar varieties of services.

In summary, you can distinguish between Web Client and Web Servers as follows:

Web Client

- Connected to the Internet when needed
- Usually runs Web browser(client) software such as Internet Explorer or Netscape
- Uses HTTP
- Request Web pages from a server
- Receives Web Pages and files from as server

Web Server

- Continually connected to the Internet
- Runs Web server software (such as Apache or Internet Information Services (IIS)
- Uses HTTP
- Receives a request for the Web page
- Responds to the request and transmits the status code, Web page, and associated files

3.4 OSI Reference Model Concept

Another model, the Open Systems Interconnection or OSI, model was designed by the International Standards Organisation (ISO). It is a seven-layered model. OSI was never seriously implemented as a protocol stack, however, it is a theoretical model designed to show how protocols stack should be implemented. The OSI model simplifies complex networking activities by grouping the steps in the process into seven separate task layers (The physical, Data Link, Network, Transport, Session, Presentation, and Application layer). This is shown in Table 3.2 below. By dividing the process into smaller tasks, it becomes easy for vendors to manage smaller pieces of the problem.

Table 3.2: OSI model

Number	Name	Function
Layer 1	Physical	This layers consist of the networking media (wiring and interconnections) and the components necessary to transmit a signal from one end to the other
Layer 2	Data Link Layer	This layer packages the data so that it can be transmitted over the Physical layer

Layer 3	Network Layer	This layer is where data is separated into frames. It also determines the route the data will take to the destination
Layer 4	Transport	This layer ensures data packets are sequenced properly and do not contain any errors
Layer 5	Session	This layer maintains a connected link, called a session, between the two communication ends
Layer 6	Presentation layer	It determines the format used for communication and compresses, encrypts, or converts the data as necessary for the protocol in use
Layer 7	Application Layer	This layer completes or initiates the actions being communicated.

Although the seven layers of the OSI model describe unique tasks performed during network communications, the demarcation between the layers and the total number of layers is irrelevant as long as all the actions are accomplished. Several other models exist apart from the OSI model.

3.5 TCP/IP Protocols Suit

A new model with only five layers would be a bit easier to understand than the OSI model. Table 3.3 presents the layered protocol stack that dominates data communications and networking today. It is a five-layered Internet model sometimes called the TCP/IP protocol suite. The model is composed of five ordered layers: physical (layer1), data link (layer 2), network (layer 3), transport (layer 4), and application (layer 5). Each layer defines a family of functions distinct from those of other layers.

Within a single machine, each layer calls upon the services of the layer just below it. Layer 4, for example, uses the services provided by layer 3 and provides services for layer 5. Between machines, layer Y on one machine communicates with the corresponding layer Y on another machine. This communication is governed by an agreed-upon series of rules and conventions called protocols. The processes on each machine that communicate at a given layer are called peer-to-peer processes. Thus, the communication between machines is therefore a peer-to-peer process using appropriate protocols for a given layer.

Table 3.3: Internet Model

Application
Transport
Network
Data Link
Physical

3.6 Common Gateway Interface (CGI)

Most users of the Internet would always prefer to visit sites that are responsive and interactive sites. Common Gateway Interface is a standard that permits the addition of dynamic functionalities to static web pages. Dynamic functionality is a requirement for site that implements site search, order form, e- mail, database display, or other type of processing. These applications expect responses from a server based on client requests. This is when server-side processing is needed, commonly known CGI. CGI is a thus a standard that defines how Web server software can delegate the generation of web pages to a client application or an executable file. A technology creates and handles dynamic documents. CGI defines how a dynamic document should be written, how input data should be supplied to the program, and how the output result should be used. The use of 'common' in CGI connotes that the standard defines a set of rules or protocols that are common to any language or platform. The term 'gateway' means that a CGI program is a gateway that can be used to access other resources such as databases and graphic packages. While the term 'interface' implies that there is a set of predefined terms, variables, calls, etc that can be used in any CGI program. CGI scripts are preferably written in scripting languages such as PHP, ASP, JSP or PERL though a common programming language such as C can also be used.

CGI programs

A CGI program in its simplest form is code written in one of the languages supporting CGI. Once you can encode a sequence of thoughts in a program and you are familiar with the syntax of one of the above-mentioned languages, writing simple CGI programs becomes very easy. The program in example 1 below is used to outputs the systems date to the browser.

A CGI program written in HTML

Example 1

```
#!/bin/sh
#The head of the program
echo Content_type:text/html
echo
          # The body of the
          program echo <HTML>
echo <HEAD><TITLE> Date and Time </TITLE></HEAD> echo
<BODY>
now=’date’
```

```
echo <CENTER><B>$now </B></CENTER>
echo</BODY>
echo</HTML> exit 0
```

Though example 1 is used to demonstrate how a simple CGI program could be written, CGI is a standard method, for a Web server to pass a Web page user's request. This is usually initiated with a form element in HTML to an application program and to accept information to send to the user. As soon as the Web server receives the request (as detailed in the form|) it passes the form information to a small application program that processes the data, and sends back a confirmation to a Web page or a message to the browser. This specification for passing data back and forth between the server and the application is called CGI and is part of the Hypertext Transfer Protocol (HTTP) specification.

Active Document

The program that runs from the client side is known as the active document. This is the document or page that may make a request to the server. For example, suppose one wants to complete an online registration form at National Open University's site. This program would certainly need to run at the client (user's) site where the online form is made available. When a browser requests an active document, the server sends a copy of the document in the form of byte code. The document is then run at the client (browser) site. An active document is stored in the server in the form of binary code. However, it does not create any form of overhead for the server in the same way that a dynamic document does. When a client retrieves an active document from a server it may store it in its storage area so that should there be a need for re-use, it would not need to make another request before it can be used. An active document is transported from the server to the client in binary form. The compression of an active document at the server side and it decompression at the client side helps to save bandwidth and transmission time during transportation.

Dynamic Document

Dynamic documents do not exist in a predefined format instead they are created by a Web server whenever a browser requests the document. When a web browser requests Web pages and their related files from a Web server, the Web server locates the files and sends them to the user's Web browser. Then the Web browser renders the returned files and displays the requested Web pages. Because a fresh document is created for each request, the contents of dynamic document can vary from one request to another. For example, suppose one wants to get the system's time and date from a server at different times. Time and date values are kinds of information that are dynamic in that they change from moment to moment. At different instances of such calls, one would expect to receive different values for time delivered to the client by the server.

CGI – Server Side Processing

A Web page engages CGI by either an action attribute on a form or a hyperlink. Action attributes on a form and hyperlink will be covered in Module Two of this course material. At this instance, any form data that exists is passed to the CGI script. The CGI script usually written in PHP, ASP, JSP, ColdFusion or PERL would complete the processing and may create a confirmation or response message, which is rendered to the browser (client). Anytime you use Google or other search engines, you are really using CGI.

Steps in Utilising CGI

- Web page engages CGI by a form or hyperlinks
- Web server executes server-side script or program
- Server-side script accesses requested database (where available), file or process
- Web server returns Web page with requested information or confirmation of action to the browser.

The location of the script being used must be known for proper referencing especially if a third party is involved in Web application development.

4.0 CONCLUSION

The client/server model of computing is a distributed application structure that partitions tasks or workloads between the providers of a resource or service, called servers, and service requesters, called clients. Often this communication is over a computer network on separate hardware, but both client and server may reside in the same system. In this section, we have covered the network model and protocols that make a Web application development possible.

5.0 SUMMARY

A server machine is a host that is running one or more server programs, which share their resources with clients. A client does not share any of its resources, but requests a server's content or service function. Clients therefore initiate communication sessions with servers, which await incoming requests. This standard for passing data back and forth between the server and the client application is called CGI and is part of the Hypertext Transfer Protocol (HTTP) specification. CGI scripts are written in PHP, ASP, JSP, ColdFusion, PERL, C or any other related programming languages.

6.0 TUTOR-MARKED ASSIGNMENT

i. With the aid of a diagram, explain the term Web application.
ii. Discuss the OSI Reference Model Concept.
iii. What is a Web Server? Give two examples.
iv. What are the steps in utilising CGI?

7.0 REFERENCES/FURTHER READING

Alex, L. & Mathew, L. (1999). *Fundamentals of Information Technology*. New Delhi: Vikas Publishing House PVT LTD.

Andy, S. (1999). *Computer Communications, Principles and Business Applications*. (2nd ed.). England: McGraw-Hill.

Behrouz, A. F. (2003). *Data Communications and Networking*. (3rd ed.). N.Y, USA: McGraw-Hill/Osborne.

Brian, *et al*.(1999). *Using Information Technology: A Practical Introduction to Computers and Communication*. Irwin/McGraw-Hill.

Terry, F-M. (2009). *Web Development and Design Foundations with XHTML*. USA: Pearson .

Katarzyna J. M. "Internet and You: Connecting to the Internet." Retrieved from http://www.rsna.org/Technology/internet2-1.cfm

Idowu, S.A., Maitanmi, S.O. & Adetunji, O. O. (2020). Introductory to Web Technology and Development. Nigeria: Jamiro Press.
June, C. (2003). *The Unusually Useful Web Book*. USA: New Ride.

MODULE 2

UNIT 1 HTML TAGS AND ATTRIBUTES

CONTENTS

1.0 INTRODUCTION

HTML stands for Hypertext Mark-up Language. It is the language for building Web pages and consists of standardised codes or “tags” that are used to define the structure of information on the Web page. Web pages come in many different varieties. In their simplest form, they contain static information, which is made up of simple texts. On the other extreme are pages, which are highly colourful, containing animation, sound and interactive elements. HTML codes make it possible for web pages to have many features including bold text, italic text, heading, paragraph breaks, tables, forms etc. Web pages generally reside on the HTTP server. A user request a web page from an HTTP (Web) server through a web browser such as, Internet Explorer, Mozilla Firefox, Safari, Chrome and so on, either by clicking on the hypertext or designating a particular URL (Uniform Resource Locator). The server then sends the requested information to the user computer.

In this unit, we shall discuss the background of HTML, Web browsers, tags and editors.

2.0 OBJECTIVES

At the end of this unit, you should be able to:

- discuss the meaning of HTML
- use simple HTML codes
- use simple HTML tags is practical designs
- write HTML codes using popular tags
- run HTML codes.

3.0 MAIN CONTENT

3.1 Background of HTML

HTML is the set of mark-up symbols or codes placed in a file intended for display on the Web browser page. These mark-up symbol and codes identify structural elements such as paragraphs, heading, and lists. HTML can be used to place media (such as graphics, video, and audio) on the Web page and describe fill-in-forms. The browser interprets the mark-up codes and renders the page. HTML permits the platform-independent display of information across network. That is, no matter what type of computer a Web page is created on, any browser running on any operating system can display the page. The new version of HTML used today is eXtensible HyperText Mark-up Language (XHTML). XHTML uses the tags and attributes of HTML along with the syntax of eXtensible Mark-up Language (XML). We shall focus on HTML in this module.

3.2 Browser

A Web browser is a software program that interprets the coding language of the World Wide Web in graphic form, displaying the translation rather than the coding. A browser acts as an interface between the user and the inner working of the web. The browser software such as the Internet Explorer, Google Chrome, Firefox, Mozilla, Safari, and so on, interpret HTML codes and presents the information contained in the web pages in a readable format on the users' computer. A browser does not display HTML tags. Browsers function as client programs by contacting the web server and sending the request for information received to the users' computes.

3.3 HTML Tags

HTML consists of standardised "tags" that are used to define the structure of information on the Web pages. The decision about the structure of the text is made by the browser based on the tags, which are marks that are embedded into the text. A tag is enclosed in two signs (< and >) and usually comes in pairs. The beginning tag starts with the name of the tag, and the ending tag starts with a slash followed by the name of the tag. The use of tags enables web pages to have many features including bold text, italic text, heading, paragraph breaks and numbered or bulleted list. Table 2.1 shows a list of common HTML tags

Table 1.1: **HTML Tags**

Opening Tag	Closing Tag	Meaning
<A>	</A>	Defines an address (hyperlink)
<BODY>	</BODY>	Defines the body of the document
<BR>		Line break
<HEAD>	</HEAD>	Defines the head of the document
<HTML>	</HTML>	Defines an HTML document
<IMG>		Define an Image
<LI>	</LI>	An item in a list
<OL>	</OL>	Ordered list
<UL>	<//UL>	Unordered list
<TITLE>	</TITLE>	Defines the title of the document

Tags are generally used to specify “mark-up” regions of HTML documents for the web browser to interpret. Tags are composed of the name of the element, surrounded by angle brackets. An end tag also has a slash after the opening angle bracket, to distinguish it from the start tag. For example, p, which is representst a paragraph by p element, would be written as:

<p> This represent a paragraph</p>

Not all elements require the end tag. An example of an element that does not require an end tag is the <br> element which forces a line break on the display of interpreted HTML codes on a browser.

HTML attributes are modifiers of HTML elements. They generally appear as name-value pairs, separated by "=", and are written within the start tag of an element, after the element's name:

<"tag" "attribute"=""value"">(content to be modified by the tag)</tag>

Where tag names the HTML element, attribute is the name of the attribute, set to the provided value. An attribute customises or modifies HTML elements.

3.4 The structure of HTML Page

The basic structure for all HTML documents is simple and should include the following minimum elements or tags:

- <html>-This is the starting tag of the html which must be closed at the end of the page
- <head>-The author of the page can insert his/her details here</head>

- <title>-The is used for the title of the page which is published on the title page of your web browser</title>
- <body>-This is a container of the main body of the page</body>
- </html>

The <HTML> Element

The HTML element is considered the root and container element for the whole HTML document. That is, its sole purpose is to encapsulate all the HTML code and describe the HTML document to the web browser. Each HTML document should have one <html> and each document should end with a closing </html> tag.

Example 1: HTML Code:

<html>
…..the contents should be here in the order of the head, title and body

</html>

The <HEAD> Element

The HEAD tag marks the beginning of the document head element; its contains the title of the pages and other parameters that the browser will use. Thus, each <head> element should contain a <title> element indicating the title of the document, and may also contain any combination of the following elements, in any order:

- The <style> tag.

  This is used for declaring or including Cascading Style Sheets(CSS) codes inside your HTML document.

- The <script> tag

  This tag is used to declare or include JAVAScript or VBScript inside the document.

- The <meta> tag

  This is used to include information about the document such as keywords and a description, which are particularly helpful for search applications.

  - The <base> tag

    This is used to create a "base" universal resource location (url) for all links on the page.

  - The <object> tag

    This is designed to include multimedia such as images, Flash animations, MP3 files, QuickTime movies JavaScript objects, and other components of a page. The <param> tag is used along with this tag to define various parameters. Note the <embed> tag can also be used to include multimedia files as will be discuss later in this module.

- The <link> tag

  This is used to link to an external file, such as a style sheet or JavaScript file.

Example 2: Codes for HEAD element

```
<head>
<meta name="Keywords" content="NOUN, Web Pages" /> <meta
name="description" content="HTML Basic Tags" /> <base
href="http://www.nou.edu.ng " />
<link rel="stylesheet" type="text/css" href="noun.css" /> <script
type="text/javascript">
_uacct = "UA-232293";
urchinTracker();
</script>
</head>
```

The <title> Element

The <title> tag is usually placed within the <head> element to title your page. Whatever is written between the opening and closing <title></title> tags will be displayed in the title bar of the WEB browser. Search engines that use its content to help index pages use the title information. Therefore, it is excellent practice to use a title that really describes the content of your site.

Example 3: Code for Title element

```
<html>
<head>
<title>National Open University of Nigeria </title>
</head>
</html>
```

The <Body> Element

The <body> element appears after the <head> element. The purpose of the <body> element is to contain the text and HTML element that will display in the browser window. A <body> element may contain anything from a couple of paragraphs, links, images under a heading to more complicated layouts containing forms and tables. We will be looking at each of these elements in detail later in this unit. For now, it is only important to understand that the body element will encapsulate all of your webpage viewable content.

Example 4: Codes for Body Element

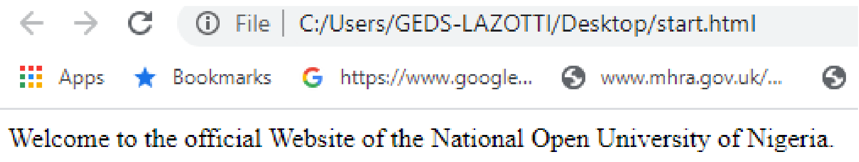
```
<html>
<head>
<title>National Open University Website!</title>
</head>
```

```
<body>
Welcome to the official Website of the National Open University of Nigeria
 </body>
</html>
```

Example 5: Codes for HTML, Head, Title and Body Tags put together

By putting all the tags together, we have a complete HTML document as follows:

```
<html>
<head>
<title>National Open University of Nigeria </title>
<meta name="Keywords" content="NOUN, Web Pages" />
<meta name="description" content="HTML Basic Tags" />
<base href="http://www.nou.edu.ng " />
<link rel="stylesheet" type="text/css" href="noun.css" />
<script type="text/javascript">
_uacct = "UA-232293";
urchinTracker();
</script>
</head>
<body>
   <P>Welcome to the official Website of the National Open University
of Nigeria.</p>
</body>
</html>
```



C:/Users/GEDS-LAZOTTI/Desktop/start.html

Welcome to the official Website of the National Open University of Nigeria.

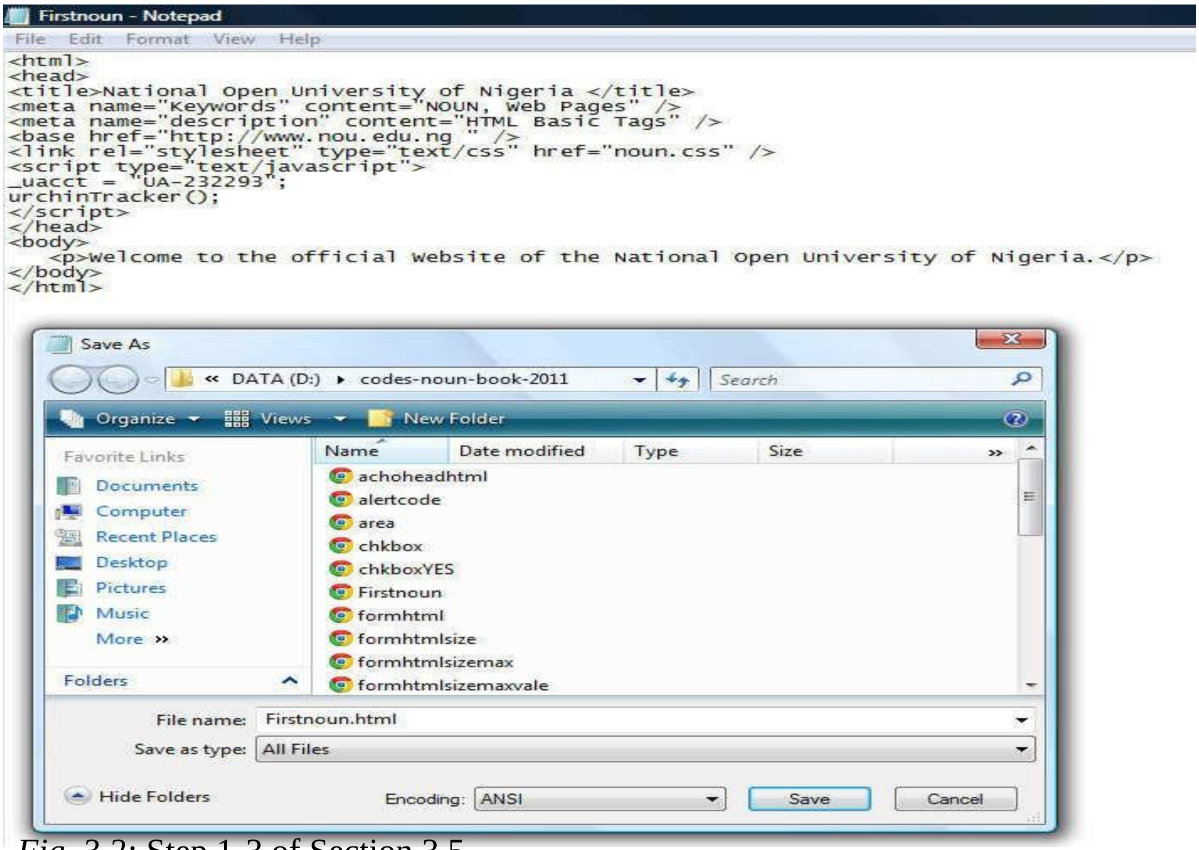
Figure 3.1 Expected output of the above tag

3.5 How to Create and Run HTML codes

Creating an HTML document is easy. HTML allows us use only ASCII characters for both the main text and formatting instructions. To begin coding HTML you need a standard text editor. Notepad is readily available on computer systems. You could also use Dreamweaver or other text editors if installed. It is not advisable to use a word processor.

Steps

1. Launch Notepad application on your computer
2. Type in your HTML codes
3. Save the document onto a location in your computer drive with a name and the extension “html or htm” (for example firstnoun.html)
4. To run the application, double-click on the file name in the location where it was saved.
5. The default browser will display the application as seen in figure 3.1.



```
<html>
<head>
<title>National Open University of Nigeria </title>
<meta name="Keywords" content="NOUN, Web Pages" />
<meta name="description" content="HTML Basic Tags" />
<base href="http://www.nou.edu.ng " />
<link rel="stylesheet" type="text/css" href="noun.css" />
<script type="text/javascript">
_uacct = "UA-232293";
urchinTracker();
</script>
</head>
<body>
   <p>Welcome to the official website of the National Open University of Nigeria.</p>
</body>
</html>
```

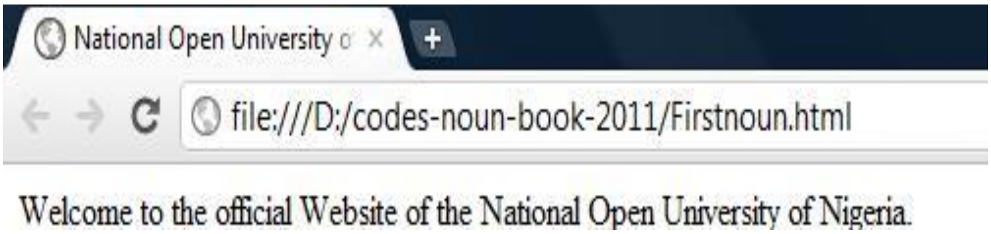
File name: Firstnoun.html
Save as type: All Files
Encoding: ANSI

Fig. 3.2: Step 1-3 of Section 3.5

Practice 1

Type in the HTML Codes in Example 5 and follow the steps in section 3.5. Run the codes by clicking on the filename and see how it appears

Answer to Practice Question 1



file:///D:/codes-noun-book-2011/Firstnoun.html

Welcome to the official Website of the National Open University of Nigeria.

Fig. 1.2: Display interpreted HTML code in a Google Chrome browser

4.0 CONCLUSION

HTML uses tags, which allow symbols or codes placed within the opening, and closing tags to be rendered to a browser. Examples of browsers include Internet explorer, Google Chrome, Firefox and so on. Browser interprets mark-up codes and renders the page to displays units. HTML permits the platform-independent display of information across network.HTML can be used to place media (such as graphics, video, and audio) on the Web page and to describe fill-in-forms.

5.0 SUMMARY

The development of Web pages is an interesting task. In this unit, we have been able to lay a good foundation for designing simple and complex Web pages. We can now identify HTML tags, put a few of them together and get the browser to interpret them.

6.0 TUTOR-MARKED ASSIGNMENT

i. What is the meaning of HTML?
ii. List the structure of HTML
iii. List and explain the meaning of five HTML tags
iv. Mention the names of five Web browsers
v. Write HTML codes to display your personal details to include name, address, department and year of study.

7.0 REFERENCES/FURTHER READING

Alex, L.& Matthew, L. (1999). *Fundamentals of Information Technology*. New Delhi:Vikas Publishing House PVT LTD.

Miller, D.(2006). *Data Communications and Network.* N.Y: McGraw Hill.

Terry, F-M.(N.D).*Web Development and Design Foundations with XHTML*. Pearson International Edition.

Idowu, S.A., Maitanmi, S.O. & Adetunji, O. O. (2020). Introductory to Web Technology and Development. Nigeria: Jamiro Press.
June, C. (2003). *The Unusually Useful Web Book*. USA: New Ride.

UNIT 2 XHTML AND HTML5 ELEMENTS

CONTENTS

1.0 INTRODUCTION

Just as in word processor software, one would want to make text bold, italicised, or underlined and apply other forms of formatting to texts. With HTML, these are also possible with the tags that will be discussed in this unit. The major differences between XHTML and HTML5 will be exposed with various examples. Similarly, we shall also cover how to embed multimedia (images, audio, and video) elements in Web pages.

2.0 OBJECTIVES

At the end of this unit, you should be able to:

- write HTML codes that enhance the appearance of web pages
- know the major differences between xhtml and html5
- identify multimedia elements
- describe different multimedia formats
- embed multimedia files in HTML codes
- create links to other web pages.

3.1 Major Differences between XHTML and HTML5

XHTML stands for Extensible HyperText Markup Language. It was developed to make HTML more extensible to work with other data format such as extensible markup language (XML). With this new feature, browsers ignore errors in HTML pages. While HTML has been updated to HTML5 to include more support for rich Internet application (RIA) such as graphics, video, mobile computing, and recent developments.

Basic rules

i. XHTML is stricter and more XML based version of HTML and it is supported by all major browsers.

ii. XHTML is case sensitive and must be in lower case while HTML5 is not (<body> <p> Welcome to National Open University of Nigeria</p></body>)

iii. XHTML must be started with <!DOCTYPE> declaration.

iv. With XHTML, all tags must be closed including the empty tags such as (<br/>, </hr> while empty tags need not to be closed with HTML5

v. XHTML must be properly nested while HTML may not (<b><i>Welcome</i></b>)

vi. XHTML attributes must be quoted
a href=https://www.w3schools.com/html/ visit our HTML tutorial </a>

Example 1. How to use of DOCTYPE in XHTML

```
<!DOCTYPE html PUBLIC “-//W3C//DTD XHTML 1.1//EN”
http://www.w3.org/TR/xhtml11/DTD/xhtml11.dtd>
<html xmlns=http://www.w3.org/2000/xhtml>
<html>
<title> xmlns=http://www.w3.org/1999/xhtml>
</head>
<body>
<p>This is an example of how to use the xhtml in starting page</p>
<br/>
</body>
</html>
```

3.2 Presentational Tags

Formatting is possible in HTML with the following tags:

Element	Example	Usage
<b>	**Bold** Text	Display text in bold
<i>	*Italic* text	Display text in italic
<big>	Big text	Display text larger than their normal size
<small>	Small text	Display text smaller than normal Size
<center>	</CENTER>	Centred
<sub>	Subscript Text	Display small text above the Baseline
<sup>	Superscript text	Display small text below the Baseline
<strike>	~~Strike~~	Displays text with a line through it

<u>	U text	Italic

The Paragraph Elements

Paragraph elements are used to group sentences and sections of text together. Texts to appear in the paragraph are contained between <p> and </p> tags.

```
<p> National Open University of Nigeria.</p>
<p> 14/16 Ahmadu Bello Way, Victoria Island
</p> <p> Lagos Nigeria.</p>
```

This will produce the following result:

National Open University of Nigeria
14/16 Ahmadu Bello Way, Victoria Island
Lagos, Nigeria.

The Line Break Element

The line break tag, <br>, is used to force a new line when a browser displays the text on the Web page document. The line break tag opening tag is used alone. It is not used as in pairs like many other tags.

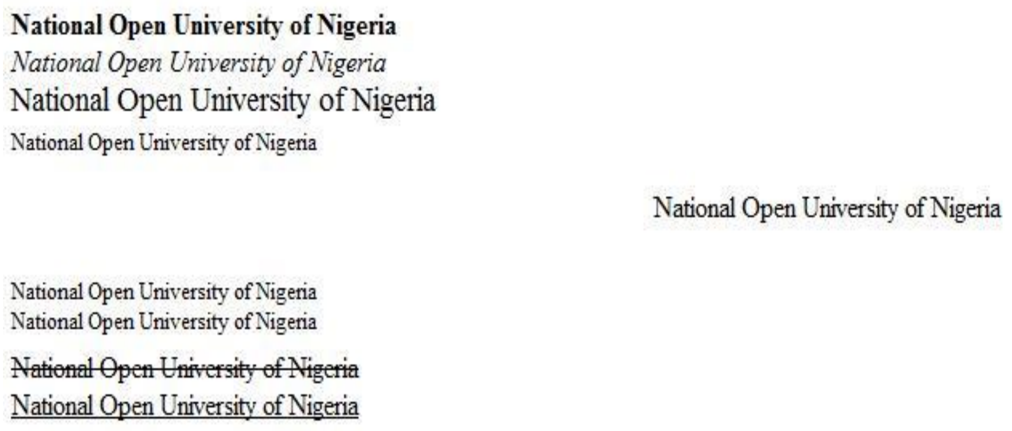
```
<br> National Open University of Nigeria.
```

This will produce the following result:

Example 2: HTML Codes: Putting all Together

```
<html>
<head>
<body>
<p><b> National Open University of Nigeria </b> <br><i> National
Open University of Nigeria</i> <br><big> National Open University of
Nigeria</big> <br><small> National Open University of
Nigeria</small> <br><center> National Open University of
Nigeria</center> <br><sub> National Open University of Nigeria</sub>
<br><sup> National Open University of Nigeria</sup> <br><strike>
National Open University of Nigeria</strike> <br><u> National Open
University of Nigeria</u>
</p>
</body>
</head>
</html>
```

This will produce:


National Open University of Nigeria
National Open University of Nigeria
National Open University of Nigeria
National Open University of Nigeria
National Open University of Nigeria
National Open University of Nigeria
National Open University of Nigeria
~~National Open University of Nigeria~~
National Open University of Nigeria

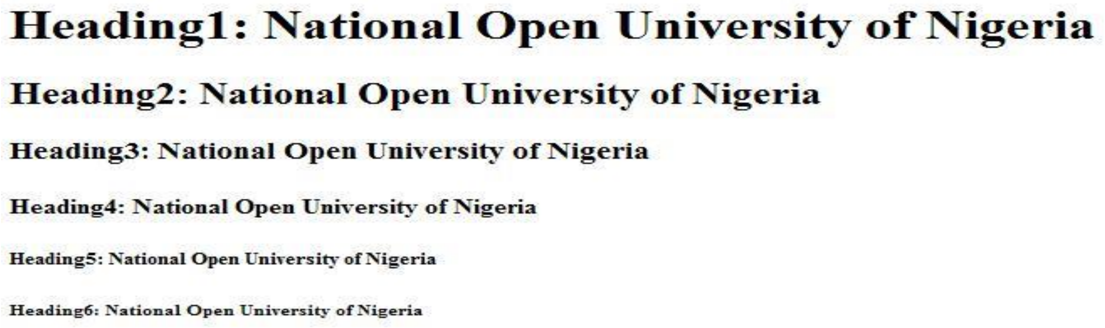
The Heading Elements

These are organised into levels <h1>, <h2>, <h3>, <h4>, <h5>, and <h6>. The size of the text is largest at <h1> and smallest for <h6>. Depending on the font being used, text contained in <h5> and <h6> may be displayed smaller than the default text size. You use different sizes for your headings. When headings are displayed by a browser, one line is added before and after each of the heading

Example 3: Codes for Heading Elements

```
          <html>
          <head>
          <body>
          <h1>Heading1: National Open University of Nigeria </h1>
<h2> Heading2: National Open University of Nigeria </h2>
<h3> Heading3: National Open University of Nigeria </h3>
<h4> Heading4: National Open University of Nigeria </h4>
<h5> Heading5: National Open University of Nigeria </h5>
<h6> Heading6: National Open University of Nigeria
</h6> </body>
</head>
</html>
```

This will display following result:


Heading1: National Open University of Nigeria
Heading2: National Open University of Nigeria
Heading3: National Open University of Nigeria
Heading4: National Open University of Nigeria
Heading5: National Open University of Nigeria
Heading6: National Open University of Nigeria

3.3 Multimedia Elements

Multimedia is media and content that uses a combination of different content forms. It includes a combination of text, audio, images, animation, video, and interactivity content forms. Multimedia elements such as sounds or videos are stored in media files. Multimedia comes in many different formats and are usually embedded in web pages to enhance their appearance and interactivity. Most web browsers today have support for a number of multimedia formats. The ones that are commonly supported are are Graphic Interchange Format (GIF), Joint Photographic Expert Group (JPEG or JPG), and Portable Network Graphics (PNG).

3.3.1 GIF Image File Format

The Graphics Interchange Format (GIF) is a bitmap image format bitmap image format that was introduced by CompuServe in 1987 and has since come into widespread usage on the World Wide Web due to its wide support and portability. It is best used for banners, clip art, and buttons. The main reason for this is that gifs can have a transparent background, which is important for web design. GIF files are usually larger files, which lead to low download times and large transfer rates. Gifs are also limited to the 256 colour scheme. The colour limitation makes the GIF format unsuitable for reproducing colour photographs and other images with continuous colour, but it is well suited for simpler images such as graphics or logos with solid areas of colour.

3.3.2 The JPEG Image File Format

The Joint Photographic Experts Group (JPEG) developed the JPEG image file format. As the name implies, JPEG is suitable for photographic images containing many colour, as supports over 16.7 million colors (with 24-bit colour). They are easier to download than gifs files and saves hard drive space because of the support of high compression. It is best to use Jpegs for photo galleries, or artwork to allow the viewer to catch that extra bit of detail.

3.3.3 The PNG Image File Format

The Portable Network Graphics (PNG - pronounced “ping”) image file format was developed in response to several needs such as more efficient format and the compression algorithm patent controversy associated with GIF format. It was created as the free, open-source successor to the GIF. It combines the best of GIF and JPEG. The PNG file format supports true colour (16 million colours) while the GIF supports only 256 colours. The PNG file excels when the image has large, uniformly coloured areas.

3.4 Image Element

Images can be included within HTML documents with the IMG element tag <img>. Images are used to enhance the appearance of web pages. Commonly used attributes with the image element are shown in the table below:.

Attribute	Value	Description
Alt	Text	Specifies an alternative text for an Image
Src	*URL*	Specifies the URL or source of an Image

Image Src

From the above, src attribute stands for *source*; that is, the source where the image file is located. The image may be directly available on your local system, a Web server or any standard URL. The src attribute must be properly pointed to the local or external source.

Example: HTML Code
`<img src="noun.gif" />`

Image:

HTML –Alternative Attribute

The *alt* attribute specifies alternate text to be displayed if for any reason, the browser does not locate the specified image or if a user has image files disabled. For browsers that support only text, the alternate attribute becomes very relevant also.

Example HTML Code:

```
<img src="http://universitymedia.com/brokenlink/noun.gif" alt="Noun
Logo" />
```

3.5 HTML Music (Audio) and Video Codes

Music and video can easily be inserted onto web page in a relatively easy way by using the embed <embed> tags. In the past, multiple tags had to be used because browsers did not have a uniform standard for defining embedded media files. A *src* attribute is required to define the correct URL (local or global) of the audio or video file in order for it to be displayed correctly. Other attributes can be set in order to customise your web pages.

Example: HTML Code for Music (Audio)

`<embed src="nounanthems.mid" />`

Example: HTML Code for Video

`<embed src="http:// universitymedia.com /files/noun.mpeg"/>`

Listing of Video Media Types

Below is the list of the most commonly used file formats for the internet.

- .swf files - are the file types created by Macromedia's Flash program.
- .wmv files - are Microsoft Window's Media Video file types.
- .mov files - are Apple's Quick Time Movie format.
- .mpeg files - set the standard for compression movie files created by the Moving Pictures Expert Group.

Flash movies (.swf), AVI's (.avi), and MOV's (.mov) file types are supported by the embed tag. The ".mpeg" files and Macromedia's ".swf" files are the most compact and widely used for the design of web pages.

3.6 Anchor Element

The HTML *anchor <a>* element is used to create a link or hyperlink reference (href) to a resource such as another web page, a file, a multimedia element, and so on or to a specific place within a web page. Each hyperlink begins with an <a> tag and ends with an </a> tag. The opening and closing tags surround the text to click to perform hyperlink. The anchor tag requires some attribute and value to work.

Hypertext Reference (href)

The *href* attribute defines reference that the link refers to. This is where the user will be taken if they wish to click this link. Hypertext references can be Internal, Local, or Global.

Example

```
<a href="http://nou.edu.ng">NOUN</a>
<a href="home.html">Home</a>
<a href="contactus.html">Contact Us</a>
<a href="register.html">Register</a>
<a href="login.html">Login</a>
```

In this example, when you click on NOUN, it will take you to the website nou.edu.ng

3.7 HTML E-mail Tag

To create e-mail link use a standard HTML anchor tag <a> and set the *href* property equal to the email address, rather than specifying a Web URL.

Example: HTML Code

```
<a href= "mailto:myaddress@gmail.com">Email </a>
```

4.0 CONCLUSION

The appearance of Web pages goes a long way in determining the number of visitors and how long they spend in a site. It is therefore necessarily for Web developers to be conversant with how to use relevant HTML tags for enhancing the presentation of text and display of images in Web Pages. Many of these tags have been covered in this unit. The three most common methods of representing images are Graphic Interface Format (GIF), Joint Photographic Experts Group (JPEG) and Portable Network Graphics (PNG). These formats allow images to be compressed for easy download from Web pages.

5.0 SUMMARY

This unit described the origin of HTML and the various advancement till date. It shows how the text content of an HTML document could be formatted with relevant HTML tags. By formatting, we mean layout and some presentation details. We also discussed how multimedia elements can be embedded into HTML documents to enhance their interactivity or appearance.

6.0 TUTOR-MARKED ASSIGNMENT

i. Describe three types of Image Format.
List and explain the meaning of five HTML tags.
ii. Name five popular Web browsers.
iii. Describe three of the rules guiding the use of XHTML
iv. Write HTML codes to display your personal details to include name, address, department and year of study.

7.0 REFERENCES/FURTHER READING

Barbara, K. K. & Norman, J. M. (2001). *The World Wide Web: A Mass Communication Perspective*. USA: Mayfield Publishing Company.

Barrie, S. & Valda, H. (2004). *Programming the Web: An Introduction*. USA McGraw-Hill.

Behrouz, A. F. (2003). *Data Communications and Networking*. (3rd ed.). N.Y, USA: McGraw-Hill/Osborne.

Deitel, P. J., & Deitel, H. M. (2008). *Internet and World Wide Web: How to Program*. (4th ed.). New Jersey, USA: Pearson Prentice Hall.
Idowu, S.A., Maitanmi, S.O. & Adetunji, O. O. (2020). Introductory to Web

Technology and Development. Nigeria: Jamiro Press.

June, C. (2003). *The Unusually Useful Web Book*. USA:New Rider.

Marc, D. M. & Thomas, C. P. (2003) .*Web Design Using Macromedia Dreamweaver*. USA: McGraw-Hill/ Irwin.

Peter, W .(2001). *Instant Web Pages*. USA: Sybex Inc.

Raymond, G. & Ellen, H. (2001). *In-Line/On-Line Fundamentals of the Internet and World Wide Web.* USA: McGraw-Hill Learning Center.

Terry, F-M. (2009). *Web Development and Design Foundations with XHTML*. USA: Pearson.

UNIT 3 TABLES

CONTENTS

1.0 INTRODUCTION

A table is layout mechanism that allows contents, which include text, images, links, forms, and even other tables to be arranged into vertical columns and horizontal rows. The rows and columns do not have to be visible, nor do they have to be equal in size. Each block of space within a table is called a table data cell. The size, background colours or background images, border colours and other formatting can easily be achieved by using Cascading Style Sheet (CSS). In HTML, tables are created using the table tag, in conjunction with the tr and td as the basic tags. Major formatting can be done on table using CSS.

2.0 OBJECTIVES

At this end of this unit, you should be able to:

- design tables and organise information using a table
- create tables using table tags
- format table by using appropriate table attributes.

3.0 MAIN CONTENT

3.1 HTML Tables

The <table> tag is used to begin a table. Within a table element are the <tr> (table rows) and <td> (table columns) tags. Tables are can be used to create site layout and server as container for forms or other elements. To create a simple table of two rows and two columns the following HTML code can be used:

HTML Code:

```
<table border="1">
<tr><td>Row 1 Col 1</td><td>Row 1 Col
2</td></tr> <tr><td>Row 2 Col 1</td><td>Row 2
Col 2</td></tr> </table>
```

Basic Table

Row 1 Col 1	Row 1 Col 2
Row 2 Col 1	Row 2 Col 2

Table rows are defined by <tr> and </tr> while table data is defined by <td> and </td>.

Let us now examine some attributes that are commonly used with tables.

3.2 Table Attributes

Commonly used table <table> attributes include align, border, border colour, with, height, cellspacing, cellpadding, bgcolour

3.2.1 The Border Attribute

This attribute is used to specify whether and what type of visible border the table will have. The values specifies range from 0 to 100 with 0 indicating no border will be visible and 1(relatively thin bother) -100 (very thick border). In the HTML code above, we specified the value of the border attribute as 1.

3.2.2 Spanning Multiple Rows and Cell

The rowspan attribute is used to span multiple rows and while the colspan is used to span multiple columns. To set headers table you need the <th> tags. Headers are by default bold to make them different from other content of the table.

HTML Code:

```
<table border="1">
<tr>
<th>Column1</th>
<th>Column 2</th>
<th>Column 3</th>
</tr>
```

```
<tr><td rowspan="2">Row 1 Col 1</td>
<td>Row 1 Column2</td><td>Row 1 Column3</td></tr>
<tr><td>Row 2 Column2</td><td>Row 2 Column3</td></tr>
<tr><td colspan="3">Row 3 Column1</td></tr>
</table>
```

Colspan and Rowspan

Column 1	Column 2	Column 3
Row 1 Column1	Row 1 Column2	Row 1 Column3
	Row 2 Column2	Row 2 Column3
Row 3 Column1		

3.2.3 Table Heading – The <th> Element

Table heading can be defined using <th> element The <th>, or table heading element is used to add heading to tables and distinguish column heading from table content. Figure 3.1 shows a table that uses <td> element.

```
<table border="1">
<tr>
<th>Name</th>
<th>Department</th>
</tr>
<tr>
<td>Okeke Ayo Sule</td>
<td>Economics</td>
</tr>
<tr>
<td>Etinosa Wada Erujeje </td>
<td>Computer Science</td>
</tr>
</table>
```

This will produce the following result.

Name	Department
Okeke Ayo Sule	Economics
Etinosa Wada Erujeje	Computer Science

Fig. 3.1 : Table

3.2.4 CellPadding and Spacing

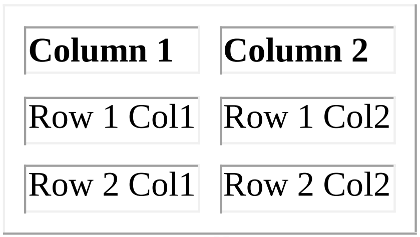
The cellspacing Attribute: This attribute specifies the distance between the cells in pixels. If a value is not specified for the cellspacing attribute, the default value (usually around 2 pixels) is assumed by the browser.

The cellpadding attribute: This attribute specifies the distance in pixel between the cell content and the edge of the cell. If you do not specify a value for the cellpadding attributes, the default value which is 1 pixel is assumed by the browser. An example with cellpadding set to 10 is shown in Figure 3.2.

HTML Code:

```
<table border="1" cellspacing="10"
<tr>
<th>Column 1</th>
<th>Column 2</th>
</tr>
<tr><td>Row 1 Col1</td><td>Row 1
Col2</td></tr> <tr><td>Row 2 Col1</td><td>Row
2 Col2</td></tr> </table>
```

Cellspacing and Padding



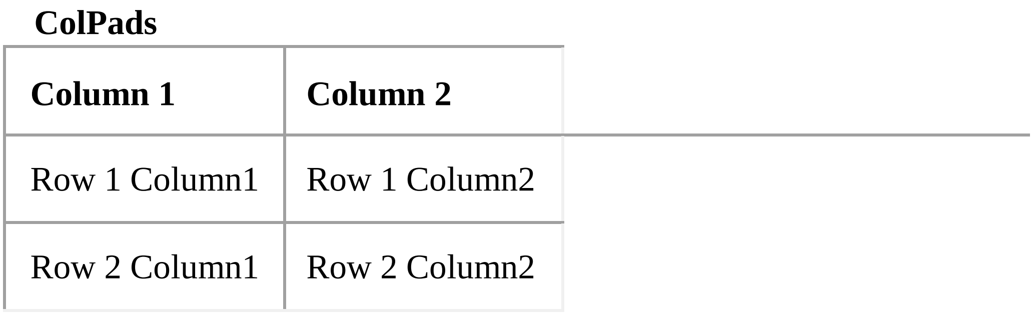
Column 1	**Column 2**
Row 1 Col1	Row 1 Col2
Row 2 Col1	Row 2 Col2

Fig. 3.2a: Cellspacing

Let us now specify a value of 10 for the cellpadding of the table and remove the cellspacing from the previous example.

HTML Code:

```
<table border="1" cellpadding="10"
<tr>
<th>Column 1</th>
<th>Column 2</th>
</tr>
<tr><td>Row 1 Column1</td><td>Row 1
Column2</td></tr> <tr><td>Row 2 Column1</td><td>Row
2 Column2</td></tr> </table>
```



ColPads

Column 1	Column 2
Row 1 Column1	Row 1 Column2
Row 2 Column1	Row 2 Column2

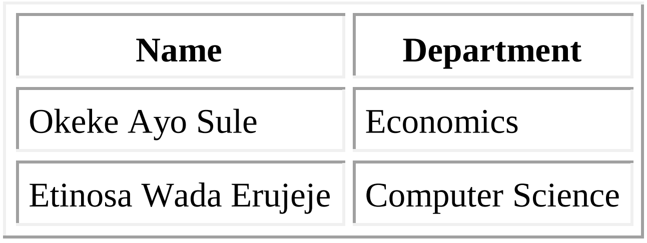
Fig. 3.2b: Padding

The value you specify for padding and spacing is interpreted by the browser as a pixel value. The value 10 specified in the two examples are simply means 10 pixels wide. That is, attributes that use numeric values for their measurements use pixels.

Figure 3.3 shows a table with cellpading of five and cellspacing of five.

```
<table border="1" cellpadding="5"
cellspacing="5"> <tr>
<th>Name</th>
<th>Department</th>
</tr>
<tr>
<td> Okeke Ayo Sule </td>
<td>Economics</td>
</tr>
<tr>
<td> Etinosa Wada Erujeje </td>
<td>Computer Science</td>
</tr>
</table>
```

This will produce the following result:



Name	Department
Okeke Ayo Sule	Economics
Etinosa Wada Erujeje	Computer Science

Fig. 3.3: Cellspacing and Padding

3.2.5 Colspan and Rowspan Attributes

The colspan Attribute: This attribute specifies the number of columns that a cell will occupy. It is also used to merge two or more columns into a single column.
The rowspan attribute: This attribute specifies the number of rows that a cell will occupy. The *rowspan* attribute is used to merge two or more rows.

Figure 3.3 shows a table with rowspans=2 and colspan=3

```
<table border="1">
<tr>
<th>Column 1</th>
<th>Column 2</th>
<th>Column 3</th>
</tr>
<tr><td rowspan="2">Row 1 Column1</td>
<td>Row 1 Column2</td><td>Row 1 Column3</td></tr>
<tr><td>Row 2 Column2</td><td>Row 2 Column3</td></tr>
<tr><td colspan="3">Row 3 Column1</td></tr>

</table>
```

This will produce the following result:

Column 1	Column 2	Column 3
Row 1 Column1 Row 1		Column2 Row 1
Column3	Row 2 Column2	Row 2 Column3
Row 3 Column1		

3.2.6 The Width and Height Attributes

The Width Attribute: This attribute specifies the width of the table in pixel or in percentage of the Web page. The table will stretch to fit the entire width of the page if 100% is used. If width is not specified, the browser assumes the width of a particular table automatically by using the width of the element and text it contains. The essence of this attribute is to customize or control the display of tables on the Web page.

The Height Attribute: This attribute specifies the height of the table in either pixel or the percentage of the Web page. This is more commonly used on <tr> and <td> tags.

You can specify table width or height in terms of integer value or in terms of percentage of available screen area. Study example 3.1 for a better understanding of the use of these attributes

Example 3.1

```
<table border="1" width="300"
height="120"> <tr>
<td>Row 1, Column 1</td>
<td>Row 1, Column 2</td>
</tr>
<tr>
<td>Row 2, Column 1</td>
<td>Row 2, Column 2</td>
</tr>

</table>
```

This will produce following result:

Row 1, Column 1	Row 1, Column 2
Row 2, Column 1	Row 2, Column 2

4.0 CONCLUSION

One of the most important features of HTML is its support for tables. Tables help to present data in more understandable ways in a browser. HTML tables work in a fashion similar to tables in Microsoft Word in that it allows authors to build columns and rows containing most of the other HTML elements and contents.

5.0 SUMMARY

HTML allows a designer to create table by using table tags in conjunction with the tr and td as the basic tags. Important attributes, which provide additional information about table elements, were also covered in this unit.

6.0 TUTOR-MARKED ASSIGNMENT

How are tables more useful as layout mechanism than plain HTML?
Give an example that show their usefulness in this regard.

7.0 REFERENCES/FURTHER READING

Barbara, K. K. & Norman, J. M. (2001). *The World Wide Web: A Mass Communication Perspective*. USA: Mayfield Publishing Company.

Barrie, S. & Valda, H. (2004). *Programming the Web: An Introduction*.

USA McGraw-Hill.

Behrouz, A. F. (2003). *Data Communications and Networking*. (3rd ed.). N.Y, USA: McGraw-Hill/Osborne.

Deitel, P. J., & Deitel, H. M. (2008).*Internet and World Wide Web: How to Program*. (4th ed.). New Jersey, USA: Pearson Prentice Hall.
Idowu, S.A., Maitanmi, S.O. & Adetunji, O. O. (2020). Introductory to Web Technology and Development. Nigeria: Jamiro Press.

June, C. (2003). *The Unusually Useful Web Book*. USA:New Rider.

Marc, D. M. & Thomas, C. P. (2003) .*Web Design Using Macromedia Dreamweaver*. USA: McGraw-Hill/ Irwin.

Peter, W .(2001). *Instant Web Pages*. USA: Sybex Inc.

Raymond, G. & Ellen, H. (2001). *In-Line/On-Line Fundamentals of the Internet and World Wide Web*. USA: McGraw-Hill Learning Center.

Terry, F-M. (2009). *Web Development and Design Foundations with XHTML*. USA: Pearson.

UNIT 4 INPUT TAGS AND FORM PROCESSING

CONTENTS

1.0 INTRODUCTION

A form is an HTML element that contains and organises other objects or controls. Form elements are like text fields, textarea fields, drop-down menus, radio buttons, checkboxes, and so on, which are used to take information from the user. Generally, a user completes a form by entering text, selecting menu items and so on before submitting it to an agent such as Web server, a mail server, etc. for processing. Within the server, the back-end application such as CGI, ASP Script or PHP script, and so on does the required processing on the users data as specified in the scripts. Results are rendered to the Web browsers if there is a need to do so.

2.0 OBJECTIVES

At this end of this unit, you should be able to:

- explain the meaning of Forms
- write HTML codes to process form information
- discuss how to use Form action and Methods
- use various form elements.

3.0 MAIN CONTENT

3.1 The Form Element

The <form> tag is used to specify the beginning of a form area on a Web page while the </form> tag is used to specify the end of a form area. Common attributes used to the <form> tags are name, method, and action. These attributes are used to specify what server-side program or file will process the form, how the form information will be sent to the server, and the name of the form.

Name Attribute: This is an optional attribute and names the form. The name of the form is required for it to be easily access by client scripting languages, such as JavaScript (will be treated latter) to edit and verify the form prior to sending its information for server-side processing

Method Attribute: This attribute is optional. The values GET or POST may be specified. When GET is specified, it causes the form data to be appended to the URL and sent to the Web server. When the value POST is specified for the attribute, it transmits the form data in the body of the HTTP response. This is a more referred and acceptable method.

Example 4.1 is a sample HTML fragment that defines a simple form that allows the user to enter a first name, last name, department and the gender of a user. When the submit button is activated, the form will be processed based on the instructions contained a PHP script (process.php).

Action Attribute: This attribute is optional. However, when a value is specified it determines the server-side program or script that will process your form data using CGI. The value should be a valid file name on a Web server. Examples PHP script(.php extension), Sun JavaServer Pages(.jsp extension), Microsoft Active Server Pages(.asp extension).

Example 1: HTML Code for Form

```
<FORM action="process.php”
   method="post"> <P>
First name: <INPUT type="text" name="firstname"><BR>
Last name: <INPUT type="text" name="lastname"><BR>
Department: <INPUT type="text" name="dept"><BR>
<INPUT type="radio" name="gender" value="Male">
Male<BR> <INPUT type="radio" name="gender"
value="Female"> Female<BR>
<INPUT type="submit" value="Send"> <INPUT type="reset"
value=”Reset”>
   </P>
```

```
    </FORM>
```

Output

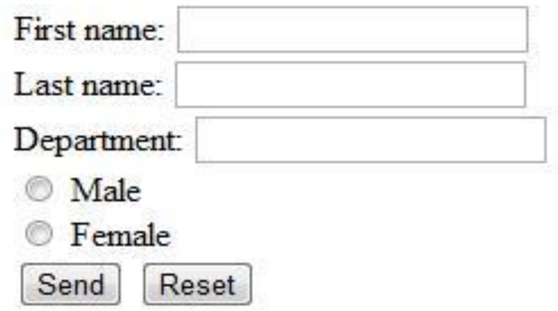


Fig. 4. 1: Form Display on Browser

3.2 HTML – Text Field/Box

Text fields are small rectangles that allow a user to simply input some text or numeric information, such as names, e-mail addresses, phone number, and other text and submit the information to the web server. The form element is configured by the <input /> tag. Common attributes of a text box are type, size, maxlength, value and password.

HTML – Text Field Size

You can control the size of the text area by specifying the size attribute.
The example below provides three different sizes for your text fields.
The default size is usually around 20 characters long. See example 2

Example 2: HTML Code for Input / Text Box Element

```
<html>
<body>
<FORM action="process.php”
   method="post"> <P>
   First name: <INPUT type="text" name="firstname" size="20"><BR>
   Last name: <INPUT type="text" name="lastname" size="19"><BR>
   Department: <INPUT type="text" name="dept"
   size="15"><BR> </P>
 </FORM>
<body>
<html>
```

Output

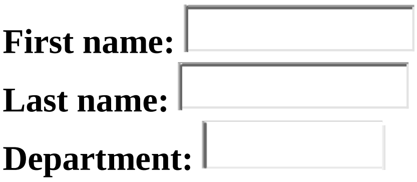


Fig. 4. 2: Textbox

HTML –TextField Maxlength

When the value is specified, TextField Maxlength is used to limit the number of characters a user can type into fields. It is a good programming practice to specify the maxlength; generally, this should match the size of your field.

Example 3: HTML Code Textbox with Maxlength Attribute <html>

```
<body>
<FORM action="process.php” method="post"> <P>
First name: <INPUT type="text" name="firstname" size="20"
maxlength="20"><BR>
Last name: <INPUT type="text" name="lastname" size="20" maxlength="20"><BR>
Department: <INPUT type="text" name="dept" size="15" maxlength="15"><BR>
                    </P>
                     </FORM>
                    <body>
                    <html>
```
[1]

Practice 1

Run the program and attempt to enter data more than the maximum length specified. Write down your experience in a sentence

HTML –TextField Value

The value attribute is used to pre-populate your text fields with some information. This can then be manipulated with any scripting language such as PHP, PERL, etc. See example 4 below.

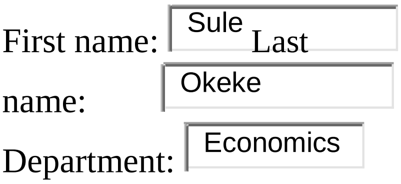
Example 4: HTML Code for TextField value

```
<html>
<body>
```

---

[1]

```
<FORM action="process.php”
   method="post"> <P>
First name: <INPUT type="text" name="firstname" size="20"
maxlength="20" value="Sule"><BR>
Last name: <INPUT type="text" name="lastname" size="20"
maxlength="20" Value="Okeke"><BR>
Department: <INPUT type="text" name="dept" size="15"
maxlength="15" Value="Economics"><BR>
 </P>
 </FORM>
<body>
<html/>
```



First name: Sule Last
name: Okeke
Department: Economics

Fig. 4. 3: Text Field with Values

3.3 HTML – Password Field

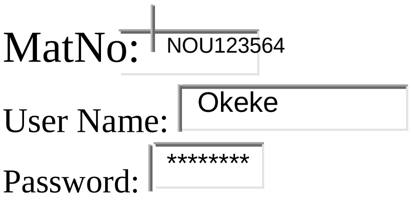
Password fields are a special type of <input /> tag. To implement them, change the type attribute from text to password. Password field is used to accept information that need to be hidden as it is entered. When a user types in information in a password box, asterisks (i.e *) are displayed instead of the characters that are being typed. This does not mean that the data entered is encrypted. To encrypt data one must use a scripting language to process the data captured. See example 5 below.

Example 5: Password Attribute

HTML Code:

```
<html>
<body>
<FORM action="process.php”
   method="post"> <P>
 MatNo: <INPUT type="text" name="matno" size="11"
maxlength="11"><BR>
User Name: <INPUT type="text" name="username" size="20"
maxlength="20"><BR>
```

```
Password: <INPUT type="password" name="pass" size="8"
maxlength="8"><BR>
        </P>
 </FORM>
<body>
<html>
```



MatNo: NOU123564

User Name: Okeke

Password: ********

Fig. 4. 4: Password Fields

3.4 HTML – Checkbox Forms

Checkboxes are another type of <input /> form. They are used for instances where a user may wish to select some or all-multiple options. The “type” attribute must be set to *checkbox* and set the name and value attributes. A sample checkbox code and the corresponding form are shown in Example 6 and Figure 5 below.

Example 6: HTML Code for Checkbox

```
<html>
<body>
<FORM action="process.php”
method="post"> <P>
<p>Please select the courses to register for the semester.</p>
CIT313: <input type="checkbox" name="courses"
value="CIT313" /><br />
CIT314: <input type="checkbox" name="courses"
value="CIT314" /><br />
CIT315: <input type="checkbox" name=" courses"
value="CIT315" /><br />
EDU325: <input type="checkbox" name="courses" value="EDU325"
/><br/>
</P>
</FORM>
<body>
<html>
```

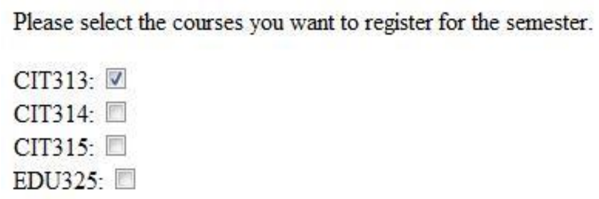
HTML –Checkboxes selected

With checkboxes, it is possible to pre-check the input boxes for viewers using the *checked* attribute. For example if the course CIT 313 is a compulsory course for all the students, it can be pre-checked. To implement this, simply set the checked attribute to “yes”. The codes for implementing checkboxes selected and the corresponding output are shown in example 7 and Figure 5 respectively.

Example 7: HTML Cod for Checkbox selected

```
<p>Please select the courses you want to register for the semester.</p>
CIT313: <input type="checkbox" checked =”yes” name="courses"
value="CIT313" /><br />
CIT314: <input type="checkbox" name="courses" value="CIT314"
/><br />
CIT315: <input type="checkbox" name=" courses" value="CIT315"
/><br />
EDU325: <input type="checkbox" name="courses" value="EDU325"
/><br/>
</P>
```

Output



Please select the courses you want to register for the semester.

CIT313: ☑
CIT314: ☐
CIT315: ☐
EDU325: ☐

Fig. 4.5: Checkbox selected

3.5 HTML –Radio Form

Radios are types of input forms that allow a user to select exactly one item from a group of predetermined items. In order to achieve this, we must properly name each radio button selection accordingly. Thus, each radio button in a group is given the same name and a unique value. The codes for implementing radio buttons and the corresponding output are shown in example 8 and Figure 6 respectively.

Example 9: HTML Code for Radio Button

```
Economics: <input type="radio" name="department” />
Computer Science: <input type="radio" name="department" />
Accounting: <input type="radio" name="department" />
```

Economics: ○

Computer Science: ○

Accounting: ○

Fig. 4.6: Radio Button

By naming these three radios “department,” they are identified as being related by the browser.

HTML –Radio Checked

By using the checked attribute, you will be able to configure the radio button to be selected by default when displayed by the browser. Example 10 shows the codes for Radio Checked while Figure 7 shows the corresponding output on a browser.

Example 10: HTML Code for Radio Checked

```
Economics: <input type="radio" name="department” checked="yes" />
Computer Science: <input type="radio" name="department" />
Accounting: <input type="radio" name="department" />
```

Economics: ●

Computer Science: ○

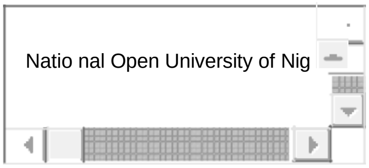
Accounting:

3.6 HTML - Textarea

This allows multi text field to be entered by a user. Paragraphs, essays, questions, descriptions or memos can be cut and pasted into textareas and submitted. Textareas have an opening tag <textarea> and a closing tag </textarea>. Example 11 and Figure 8 show the codes and the output of textarea on a browser respectively.

Example 3: HTML Code for Textarea

```
<textarea>National Open University of Nigeria!</textarea>
```



Natio nal Open University of Nig

Fig 4.7: Textarea

HTML – Text area Col and Rows

To adjust the size of the appearance of the text area requires two attributes, *cols* and *rows* with numeric values. The larger the value the larger the field will appear. Example 12 is a sample code for implementing text area col and row.

Example 12: HTML Code for col and row in textarea

```
<textarea cols="30" rows="10">Text Area!</textarea>
<textarea cols="10" rows="2">Text Area!</textarea>
<textarea cols="25" rows="5">Text Area!</textarea>
```

3.7 HTML –Selection Forms and Drop Down Lists

Drop down lists are the basic selection forms. The <select> container tag along with the <option> tags are used to configure the select list. Other names used to describe the drop down list are select list, select box, drop-down box, and option box. Drop down lists have several options a user can select. A sample code to implement drop down list and the corresponding output on a browser are shown in example 13 and Figure 9.

Example 13: HTML Code for Drop Down List

```
<select>
<option>Economic </option>
<option>Computer Science </option>
<option>Accounting </option>
</select>
```

By default, the first coded <option> will be displayed or selected as the default. We can change this using the *selected* attribute.

Example 14: HTML Code for Drop Down list with selected attribute

```
<select>
<option>Economic </option>
<option>Computer Science </option> <option
selected=”yes”>Accounting </option>
</select>
```

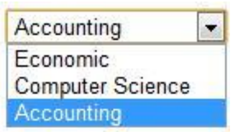


Fig.4.8: Drop Down list

3.8 Submit Button

Submit
Submission buttons are a type of <input /> tag and is used to submit the form. To achieve form submission, set the type attribute to submit. This creates a special type of button in forms that will cause the browser to send the form data to the web server provided there are available server scripting codes to achieve this.

HTML – Reset Button

Reset
Reset buttons exist to reset the fields of a form to its initial vales. See the codes in example 14.

Example 14: HTML Code for Submit and Reset Buttons

```
<input type="submit" value="Submit" /><br />
<input type="Reset" value="Reset" /><br />
```

Notice that in the above example we also changed what was written on our button using the *value* attribute. This can be changed to any value you wish.

Practice 2
Review the code in example 4.1, make some changes to it and run it.

4.0 CONCLUSION

Forms are important because they help Web page visitors to view data and gather or submit information to a server. Form elements include text boxes, labels, lists, check boxes and radio buttons. HTML tags that configure these form elements include the <input/> and <textarea> tag. Forms usually need to call some type of server-side processing scripts to perform functions such as updating a database, submitting form data and sending a mail. The server side script can be written in PHP (Hypertext Pre-processor) language, JSP (Java Server Pages), ColdFusion, and so on.

5.0 SUMMARY

In this unit, we discussed how to create and use form on Web Pages. You also learnt how to configure elements of form such as text boxes, labels, checkboxes etc. In the next module we shall be looking at how to design more attractive and interactive Web pages.

6.0 TUTOR-MARKED ASSIGNMENT

i. Which attribute of the <form> tag is used to specify the name and the location of the script that will process the form field values?

   (a) action
   (b) process
   (c) method
   (d) none of the above

ii. If you are to conduct a survey and ask your Web page visitors to vote for their favourite candidate in an election, which of the following form elements is best to use for this purpose?

   a. check box b.
   radio button c.
   text box
   d. scrolling text box.

7.0 REFERENCES/FURTHER READING

Barbara, K. K. & Norman, J. M. (2001). *The World Wide Web: A Mass Communication Perspective*. USA: Mayfield Publishing Company.

Barrie, S. & Valda, H. (2004). *Programming the Web: An Introduction*. USA McGraw-Hill.

Behrouz, A. F. (2003). *Data Communications and Networking*. (3rd ed.). N.Y, USA: McGraw-Hill/Osborne.

Deitel, P. J., & Deitel, H. M. (2008).*Internet and World Wide Web: How to Program*. (4th ed.). New Jersey, USA: Pearson Prentice Hall.

June, C. (2003). *The Unusually Useful Web Book*. USA:New Rider.

Idowu, S.A., Maitanmi, S.O. & Adetunji, O. O. (2020). Introductory to Web Technology and Development. Nigeria: Jamiro Press.

Marc, D. M. & Thomas, C. P. (2003) .*Web Design Using Macromedia Dreamweaver*. USA: McGraw-Hill/ Irwin.

Peter, W .(2001). *Instant Web Pages*. USA: Sybex Inc.

Raymond, G. & Ellen, H. (2001). *In-Line/On-Line Fundamentals of the Internet and World Wide Web.* USA: McGraw-Hill Learning Center.

Terry, F-M. (2009). *Web Development and Design Foundations with XHTML*. USA: Pearson.

MODULE 3

UNIT 1 CASCADING STYLE SHEET (CSS) FUNDAMENTALS

CONTENTS

1.0 INTRODUCTION

CSS stands for Cascading Style Sheets. It defines how to display HTML elements. HTML was never intended to contain tags for formatting a document. It came as an after thought. HTML was intended to define the content of a document, like:

```
<h1>This is a heading</h1>
<p>This is a paragraph.</p>
```

The use of CSS enables all formatting to be removed from HTML document, and stored in a separate CSS file. CSS helps to control the look and feel of HTML documents in an organised and efficient manner. Thus with the use of CSS, a designer will be able to achieve the following:

- add new looks to HTML pages
- completely restyle a web site with only a few changes to your CSS code

- use the “style” you create on any webpage you wish.

2.0 OBJECTIVES

At the end of this unit, you should be able to:

- explain the meaning and importance of CSS
- use CSS to format web pages
- add CSS to HTML files.

3.0 MAIN CONTENT

3.1 Advantages of Cascading Style Sheets

- Typography and page layout can be better controlled. These features include font size, line spacing, letter spacing, indent, margin, and element positioning
- Style is separable from structure. The format of the text and colours used on the page can be configured and stored separately from the body section of the Web page document
- Document are potentially smaller: The formatting is separate from the documents, therefore, the actual document should be smaller
- Style can be stored. This makes site maintenance easier. Again, if the style needs to be changed, it is possible to complete the modifications by changing the style sheet only.

3.2 Syntax of CSS

Cascading style sheets are made up of rules that describe the styling to be applied. A CSS rule has two main parts: a selector, and one or more declarations:

Selector	**Declaration**	**Declaration**
h1	{Colour: green ;	text-align:center;}
	Property Value	Property Value

In the above format, the selector is the heart and soul of CSS. It defines which HTML elements that will be manipulated with CSS code. Each declaration consists of a property and a value. The property is the style attribute you want to change and each property has a value. A CSS declaration always ends with a semicolon, while curly brackets surround declaration groups.

Example 1a: CSS Code

```
p {colour:green;text-align:center;}
```

The codes in example 1a can also be written as in example 1b to make the CSS more readable by putting one declaration on each line, as follows:

Example 1b: CSS Code

```
p
{
colour:green;
text-align:center;
}
```

3.3 CSS Comments

Comments are used to explain code, and may help when editing the source code later. Comments are not executed or run by browsers. The use of comment in CSS is demonstrated in example 2.

A CSS comment begins with "/*", and ends with "*/", like this:

```
/*This is a comment*/
```

Example 2: Comments in CSS

```
p
{
text-align:center;
/*This is another comment*/

color:black;
font-family:arial;
}
```

3.4 Types of Cascading Style Sheets

Cascading style sheets come in mainly in three flavours: internal, external, and inline. We shall study these flavours in this unit.

3.4.1 Internal CSS

This is defined in the header of a Web page and applies to the Web page document. To use it, you must add a new tag, <style>, inside the <head>

tag. The HTML code below contains an example of internal style <style>'s usage.

Example 3: Internal CSS code

```
<html>
<head>
<style type="text/css">
</style>
</head>
<body>
<p> National Open University of Nigeria
</p> </body>
</html>
```

The effect of the use of CSS is not easily noticed in this code. The code style tag just tells the browser that we will be defining some CSS to be used on this page.

Creating Internal CSS code

CSS codes are written to manipulate existing HTML codes. Example 4 represents the use of Internal CSS code in a simple and functional Web page design.

Example 4: Internal CSS Code

```
<html>
<head>
<style type="text/css">
p {colour: white; }
body {background-colour: black; }
</style>
</head>
<body>
<p>This displays National Open University using white text on a black
background!</p>
</body>
</html>
```

Display:

This displays National Open University using white texts on a black background!

Fig. 1.1: Internal CSS

What we did in the above example was to manipulate <p> and <body> which are common HTML tags. In addition, we went through the following steps:

Stage 1

- We chose the HTML element we wanted to manipulate. **- p{ : ; }**
- Then we chose the CSS attribute color. - p { **color:** ; **}**
- Next we choose the font color to be white. - p { color: **white**; **}**

Stage 2

- We choose the HTML element Body **- body { : ; }**
- Then we chose the CSS attribute. - body { **background-colour:** ; **}**
- Next we chose the background colour to be black. - body { background-colour:**black**; **}**

3.4.2 Using External CSS

The use of external CSS helps to keep CSS separate from HTML codes. Placing CSS in a separate file allows for easier maintenance of web pages. To use external CSS file, we need to first create the required CSS code and save it with a ".css" file extension. This CSS file is then referenced in your HTML codes using the <link> **instead** of <style> tag.

Example 5: Creating and using external CSS

```
h3{ colour: red; }
p {colour: white; }
body {background-colour: black; }
```

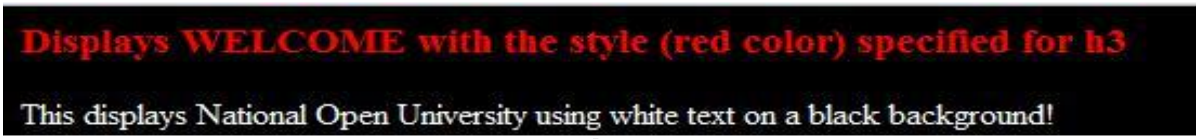
Practice 1

Open up notepad.exe, or any other plain text editor and type the CSS codes in Example 5 above. Now save the file as a CSS (.css) file. Make sure that you are not saving it as a text (.txt) file, as notepad likes to do by default. Name the file “test.css” (without the quotes). Now create a new HTML file and fill it with the following code. Then save this file as “example6.html” (without the quotes) in the same directory as your CSS file. Now open your HTML file in your web browser and compare with what we have in Figure 2.

Example 6: HTML Code

```
<html>
<head>
<link rel="stylesheet" type="text/css" href="test.css" />
```

```
</head>
<body>
<h3> Displays WELCOME with the style (red colour) specified for h3
</h3>
<p>This displays National Open University using white text on a black
background!</p>
</body>
</html>
```



Displays WELCOME with the style (red color) specified for h3

This displays National Open University using white text on a black background!

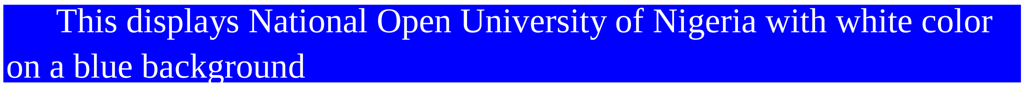
Fig.1.2: HTML with CSS

3.4.3 Using CSS Inline

Inline style can be coded in the body of the Web pages as an attribute of HTML tag. Inline CSS has the highest priority out of external, internal, and inline CSS. This means the inline style can override styles that are defined in external or internal. To add a style inside an HTML element, all we need to do is specify the desired CSS properties with the style HTML attribute. Study Example 7 and Figure 3.

Example 7: CSS Inline Code

```
<p style="background: blue; colour: white;"> This displays National
Open University with white color on a blue background </p>
```



This displays National Open University of Nigeria with white color on a blue background

Fig.1.3: Inline CSS

3.5 Using CSS Classes

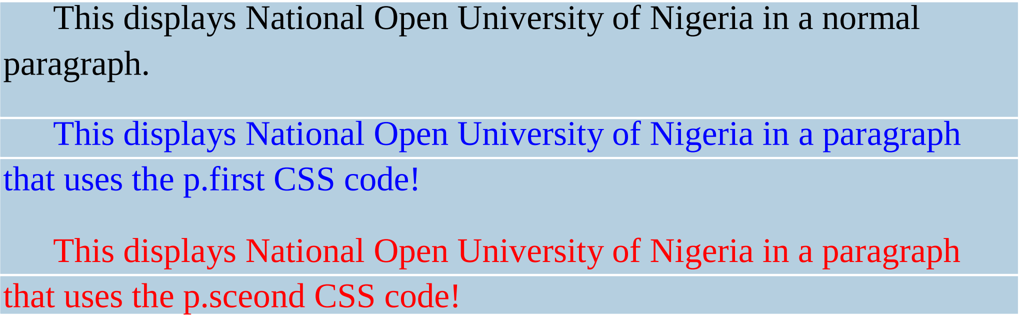
CSS classes makes it possible to apply a CSS rule to certain class of elements on a Web page and not necessarily tie the style to a particular HTML tag. It also makes it possible to give an HTML element multiple looks with CSS. For example, sometimes, one may wants the font to be large and white, while at another, one would prefer the font to be small and black for a particular HTML items for instance, Paragraph <p>.

Using classes is simple; it requires need to add an extension to the typical CSS code and make sure to specify this extension in the HTML codes. We can demonstrate this by making two paragraphs that behave differently. First we begin with the CSS, note the red text.

```
p.first{ colour: blue; }
p.second{ colour: red; }
```

Example 9: HTML Code using CSS Classes

```
<html>
<body>
<p>This displays National Open University of Nigeria in a normal
paragraph.</p>
<p class="first">This displays National Open University of Nigeria in a
paragraph that uses the p.first CSS code!</p>
<p class="second"> This displays National Open University of Nigeria
in a paragraph that uses the p.second CSS code! </p> ...
```



This displays National Open University of Nigeria in a normal paragraph.

This displays National Open University of Nigeria in a paragraph that uses the p.first CSS code!

This displays National Open University of Nigeria in a paragraph that uses the p.sceond CSS code!

Fig.1 4: Display

3.6 Background and Color Properties

background

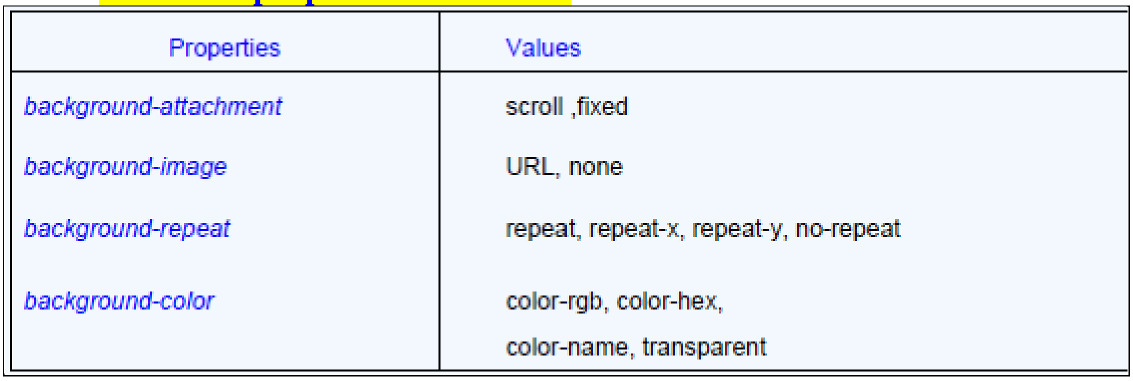
– background: “color” / “#rrggbb” / url(“*.gif”)

color

– color : “color name” / “#rrggbb”

Eg. **BODY{ Background:”red”;} // kindly use the American spelling for your color or follow the keyboard settings**

Table 1.1. properties and values



Properties	Values
background-attachment	scroll ,fixed
background-image	URL, none
background-repeat	repeat, repeat-x, repeat-y, no-repeat
background-color	color-rgb, color-hex, color-name, transparent

Background Properties –Example

Font Properties

Properties	**Values**
Font-family	Arial, Monospace,
Font-style	Normal, italic, oblique
Font-variant	normal, small-caps
Font-size	x-small, small, medium, large
Font-weight	normal, bold, bolder, light, x-large

CSS measurements

When you manipulate text and other objects with a style sheet, you often must specify a length or size. CSS supports measurements such as

1) inches (in)
2) centimeters (cm)
3) millimeters (mm)
4) point size (pt)
5) pixels (px)

3.7 Understanding CSS Background

The background of your website is very important. With CSS, you are able to set the background color of image of any CSS element. With CSS, a colour is most often specified by:
a HEX value - like "#ff0000"

- an RGB value - like "rgb(255,0,0)"
- a color name - like "red"

Colours can be produced for a number of page elements using the color names or RGB, or hexadecimal codes as demonstrated in example 10.

Example 10: CSS Background Code

```
h4 { background-colour: white; }
p { background-colour: #1078E1; }
body { background-colour: rgb( 149, 206, 145);

}
```

3.8 Using CSS Background Image

The background- image property specifies an image to use as the background of an element. By default, the image is repeated so it covers the entire element. The background image for a page can be set as demonstrated in Example 11.

Example 11: CSS Background image Code

```
p { background-image: url(myPic.jpg); }
body {background-image:url('logo.gif');}
h4{ background-image: url(http://www.nou.ed.ng /pics/cssPic/logo.gif);
}
```

3.9 CSS Fonts

CSS allows a great control over the way text is displayed. Thus, one can change the text size, color, style, and lots more can be achieved.

3.9.1 CSS Font Colour

CSS colours are defined using a hexadecimal (hex) notation for the combination of Red, Green, and Blue colour values (RGB). The lowest value that can be given to one of the light sources is 0 (hex 00). The highest value is 255 (hex FF). This is illustrated in Example 12.

Example 12: CSS Font Color Code

```
h4 {color: red ;}
h5 {color: #9000A1 ;}
h6 {color: rgb (0,220, 60) ;}
```

3.9.2 CSS Font Family

A specific font name or a generic font family may assign font families. This is illustrated in Example 13.

Example 13: CSS Font Family Code

```
h4{font-family:sans-serif;}
h5{font-family:serif;}
h6 { font-family: arial;}
```

3.9.3 CSS Font Size

Font sizes are used to manipulate the size of fonts by using values, percentages, or key terms. Key terms are not very useful; the common terms are xx-large, x-large, large, medium, small, x-small, and xx-small. One may use values if one does not want the user to be able to increase the size of the font. Percentages are used when one wants to change the default font, but does not want to set a static value. This is illustrated by Example 14.

Example 14: CSS Font Size Code

```
p{font-size:120%;}
body{font-size:10px;}
h1{ font-size: x-large;}
```

3.9.4 Text Align

By default, texts on any website are aligned to the left. If a different alignment is required, specify it by using the text-align attribute. This is illustrated in Example 15.

Example 15: CSS Text Align Code

```
p {text-align: right; }
h2 {text-align: justify; }
```

3.9.5 Text Transform

Text-transform is used to modify the capitalisation of text. This is illustrated in Example 16.

Example 16: CSS Text Transform Code

```
p { text-transform: capitalize; }
h5{ text-transform: uppercase; }
h6{ text-transform: lowercase; }
```

4.0 CONCLUSION

Cascading style sheets refers to a unique processing approach by which multiple style sheets may be applied to single Web page but is supported by all style sheet languages. There are many advantages, which result from using style sheets to apply stylistic formatting to Web pages. For example, a Web developer can separate documents available to many pages in a Web site, thus enabling the developer to apply similar styles across many documents while having to modify styles in one file when there is a need.

5.0 SUMMARY

Cascading style sheets were discussed in this unit as means of providing a uniform and consistent presentation detail in HTML documents. The three levels i.e Internal, External and Inline in which style sheet can appear were covered. CSS classes were also covered.

6.0 TUTOR-MARKED ASSIGNMENT

i. What is a style sheet selector?
ii. How can styles be referenced in an HTML page?
iii. What property can be set in order to change the colour of paragraph text to red (from the default black)?
iv. Write an internal style for producing an <H2> heading tag that has a blue background, is in italics, and is centred.

7.0 REFERENCES/FURTHER READING

Barbara, K. K. & Norman, J. M. (2001). *The World Wide Web: A Mass Communication Perspective*. USA: Mayfield Publishing Company.

Barrie, S. & Valda, H. (2004). *Programming the Web: An Introduction*. USA McGraw-Hill.

Behrouz, A. F. (2003). *Data Communications and Networking*. (3rd ed.). N.Y, USA: McGraw-Hill/Osborne.

Deitel, P. J., & Deitel, H. M. (2008).*Internet and World Wide Web: How to Program*. (4th ed.). New Jersey, USA: Pearson Prentice Hall.

Idowu, S.A., Maitanmi, S.O. & Adetunji, O. O. (2020). Introductory to Web Technology and Development. Nigeria: Jamiro Press.

June, C. (2003). *The Unusually Useful Web Book*. USA:New Rider.

Marc, D. M. & Thomas, C. P. (2003) .*Web Design Using Macromedia Dreamweaver*. USA: McGraw-Hill/ Irwin.

Peter, W .(2001). *Instant Web Pages*. USA: Sybex Inc.

Raymond, G. & Ellen, H. (2001). *In-Line/On-Line Fundamentals of the Internet and World Wide Web.* USA: McGraw-Hill Learning Center.

Terry, F-M. (2009). *Web Development and Design Foundations with XHTML*. USA: Pearson.

UNIT 2 FUNDAMENTALS OF JAVASCRIPT FOR DYNAMIC STATEMENTS

CONTENTS

1.0 INTRODUCTION

JavaScript is a scripting language that was developed by Netscape Communicator to provide interactivity to static Web pages. The language was originally developed by Netscape under the name LiveScript in late 1995. Netscape and Sun in December 1995 later released LiveScript under the name JavaScript. Many people seem to be confused about the relationship of JavaScript and Java, which is a separate programming language. JavaScript is a simple, interpreted language while Java is a compiled object-oriented programming language. In this unit, we shall describe basic facts about JavaScript and how to incorporate the most commonly used JavaScript elements into pages.

2.0 OBJECTIVES

At the end of this unit, you should be able to:

- explain JavaScript
- write simple JavaScript programs
- use arithmetic operators
- use JavaScript to make web pages interactive.

3.0 MAIN CONTENT

3.1 Inserting a JavaScript into an HTML page

To insert a JavaScript into an HTML page, we use the <script> tag. Inside the <script> tag, we use the type attribute to define the scripting language.

So, we have <script type="text/javascript"> and </script> to connote where the JavaScript starts and ends. Example 1 is a simple JavaScript code that displays on a browser “Welcome to National Open University of Nigeria” – without the quotes.

Example 1: Simple JavaScript Code

```
<html>
<body>
<scripttype=”text/javascript”>
document.write(“Welcome to National Open University of Nigeria”);
</script>
</body>
</html>
```

In this example, the “document.write” command is a standard JavaScript command for writing output to a page. By entering the document.write command between the <script> and </script> tags, the browser will recognise it as a JavaScript command and execute the code line.

3.2 JavaScript Comments

Comments are added to JavaScripts codes to make them more readable. JavaScript allows the use of single line or multiple lines comments. To put a comment on a single line use //. Example 2 illustrates the use of a comment in JavaScript codes.

Example 2: Comments in JavaScript

```
<script type="text/javascript"> // My details are as displayed in the
following three paragraphs document.write("<p>My Name is Ade Musa
Okeke </p>");
document.write("<p>I am in the School of Science and
Technology.</p>");
document.write("<p>My Matriculation Number is NOU031111.</p>");
</script>
```

To use multi line comments start with /* and end with */. Example 4 is used to illustrate the use of multi line comments (/* */).

Example 3: Multi line comments

```
<scripttype="text/javascript">
                /*
My details as a student of the National Open University of Nigeria are displayed
in the next three paragraphs
                */
document.write("<p>My Name is Ade Musa Okeke </p>"); document.write("<p>I am
in the School of Science and Technology.</p>");
document.write("<p>My Matriculation Number is NOU031111.</p>"); </script>
```

3.3 Where to Locate JavaScript in a Program

JavaScript code can be located internally within the program or externally. If it is to be within the program, then it has to be located in the body or head section of an HTML page. Since program instructions are executed sequentially, scripts that are to be executed latter or when a user clicks a button are better placed in as a function. For easy maintenance of programs, it is better to separate function from the main page content by locating them in the head section.

Example 4: JavaScript Codes located in the head section

```
<html>
<head>
<scripttype="text/javascript">
functionmessage()
{
alert("This alert box was called with the onload event");
}
</script>
</head>

<bodyonload="message()">
</body>
</html>
```

If one does not want a script to be placed inside a function, or if one’s script should write page content, it should be placed in the body section.

Example 5: JavaScript Codes located in the body section

```
<html>
<head>
</head>
<body>
<scripttype="text/javascript">
document.write("This message is written by JavaScript");
</script>
</body>
</html>
```

3.3.1 Using an External JavaScript

To use JavaScript as external file, first it has to be written and saved with a .js file extension. Then point to the .js file in the “src” attribute of the <script> tag. Example 4 illustrates the use of JavaScript as an external file.

Example 4: External JavaScript

```
<html>
<head>
<scripttype="text/javascript"src="extfile.js"></script>
</head>
<body>
</body>
</html>
```

3.4 JavaScript Statements

JavaScript is a sequence of statements to be executed by the browser. Each statement must be separated by a semi colon. Example 5 is used to illustrate how JavaScript program can be used to display the details of student to the Web page

Example 5: Student details

```
<scripttype="text/javascript">
document.write("<p>My Name is Ade Musa Okeke </p>");
document.write("<p>I am in the School of Science and
Technology.</p>");
document.write("<p>My Matriculation Number is NOU031111.</p>");
</script>
```

3.4.1 JavaScript Blocks

JavaScript statements can be grouped together in blocks. Blocks start with a left curly bracket {, and ends with a right curly bracket}. The purpose of a block is to make the sequence of statements execute together. In Example 6, the three lines of the student's details are treated as a block.

Example 6: Block Statements

```
<scripttype="text/javascript">
{
document.write("<p>My Name is Ade Musa Okeke </p>");
document.write("<p>I am in the School of Science and Technology.</p>");
document.write("<p>My Matriculation Number is NOU031111.</p>");
}
</script>
```

3.5 JavaScript Variables

Variables are "containers" for storing information. As with algebra, JavaScript variables are used to hold values or expressions. A variable can have a short name, like amt, or a more descriptive name, like amount

Rules for JavaScript variable names

Variable names are case sensitive (the variable amt and AMT are two different variables)

Variable names must begin with a letter or the underscore character.

Declaring (Creating) JavaScript Variables

A variable is declared by preceding it with the keyword var. Example 7 shows valid declaration of variables in JavaScript.

Example 7: Declaration Statements

```
var x; var myname;

var examscore
var radius
var greetings;
```

Assignment Statement

As long as no values are assigned to variable, they will remain empty. To assign values to the variables use the assignment operator (=). We will learn about other operators later in this module. In Example 8, we combine both the declaration and assignment statements.

Example 8: Assignment and declaration

Statement varx=5;

var myname=”Adebola”;
var examscore=89;
var radius=1.0;
var greetings=”Welcome”;

In Example 8, variable **x** holds the value **5**, myname holds the value Adebola, examscore holds the value 89, radius holds the value 1.0 while greetings holds the value Welcome. Note the use of quotes in the assignment of a text value to variables myname and greetings and the use of semicolon after each variable declaration. Semicolon is used in JavaScript to mark the end of a statement

JavaScript also makes it possible to assign a value to variable that has not been declared. See the example below:

amt =10;

This is the same as

var Amt =10;

3.6 Arithmetic Operators and Expressions

An **arithmetic expression** is one, which is evaluated by performing a sequence of arithmetic operations to obtain a numeric value to replace the expression. Arithmetic operators are used to perform arithmetic between variables and/or values. Table 1 shows a list of arithmetic operator and expressions.

Given that **Y=10**, the table below explains the arithmetic operators:

Table 2.1: Arithmetic Operators and Expressions

Operators	Meaning	Example	Result
+	Addition	X=Y+2	X=12
-	Subtraction	X=Y-2	X=8
*	Multiplication	X=Y*2	X=20
/	Division	X=Y/2	X=5
%	Modulus	X=Y%2	X=0
++	Increment	X++	X=11
--	Decrement	X--	X=9

The list above is similar to that of basic mathematics. The only symbol that might look new is the modulus ("%"), which divides one operand by another and returns the remainder as its result. In addition, the + operator can be used to add string variables or text values together.

To add two or more string variables together, use the + operator.

```
txt1="NationalOpen";
txt2="UniversityofNigeria";
txt3=txt1+txt2;
```

After the execution of the statements above, the variable txt3 will contain "National Open University of Nigeria."

Practice 1
The program below computes the area of a circle. Type the codes ising a text editor preferably notepad.exe. Save an HTML file and open it with a browser. What is the result?

```
<html>
<body>
<scripttype="text/javascript">
varradius=5;
var area =radius * radius * 3.14159
 document.write("The Area of the Circle with
 radius=5"+area); document.write("<br/>");
 }
</script>
</body>
</html>
```

4.0 CONCLUSION

JavaScript statements are typically embedded directly with HTML. A single HTML document can include any number of embedded scripts. When used properly, JavaScript has the capacity to improve the look and enhance user's interactivity with Web pages. Some statements that will enable one to write simple JavaScript codes have been covered in this unit.

5.0 SUMMARY

JavaScript is the most popular scripting language of the Internet. It is majorly used as a client-side scripting language to add interactive functionality, validate forms, detect browsers, etc. in Web design. Some of its constructs have been covered in this unit. It is supported by major browsers, such as Internet Explorer, Firefox, Chrome, Opera, and Safari.

6.0 TUTOR-MARKED ASSIGNMENT

i. Who developed JavaScript and when? Which browser support JavaScript?
ii. Locate a JavaScript calculator and explain how it works.
iii. Using JavaScript, design a Web page that converts temperature reading in Celsius to Fahrenheit scale.

7.0 REFERENCES/FURTHER READING

Barbara, K. K. & Norman, J. M. (2001). *The World Wide Web: A Mass Communication Perspective*. USA: Mayfield Publishing Company.

Barrie, S. & Valda, H. (2004). *Programming the Web: An Introduction*. USA McGraw-Hill.

Behrouz, A. F. (2003). *Data Communications and Networking*. (3rd ed.). N.Y, USA: McGraw-Hill/Osborne.

Deitel, P. J., & Deitel, H. M. (2008).*Internet and World Wide Web: How to Program*. (4th ed.). New Jersey, USA: Pearson Prentice Hall.

Idowu, S.A., Maitanmi, S.O. & Adetunji, O. O. (2020). Introductory to Web Technology and Development. Nigeria: Jamiro Press.

June, C. (2003). *The Unusually Useful Web Book*. USA:New Rider.

Marc, D. M. & Thomas, C. P. (2003) .*Web Design Using Macromedia Dreamweaver.* USA: McGraw-Hill/ Irwin.

Peter, W .(2001). *Instant Web Pages*. USA: Sybex Inc.

Raymond, G. & Ellen, H. (2001). *In-Line/On-Line Fundamentals of the Internet and World Wide Web.* USA: McGraw-Hill Learning Center.

Terry, F-M. (2009). *Web Development and Design Foundations with XHTML*. USA: Pearson.

UNIT 3 USING STATEMENTS IN CONTROL JAVASCRIPT

CONTENTS

1.0 INTRODUCTION

JavaScript programs will be executed in the order in which statements are written except for the use of control statements with the scripts. The use of control statements can lead to the conditional, repeated and alteration of the normal sequential flow of control. Control statements in JavaScript are similar to their counterparts in C/C++ and Java. They are thus easy to learn.

2.0 OBJECTIVES

At the end of this unit, you should be able to:

- implement logical construct with JavaScript
- apply decision statements with JavaScript
- use loops with JavaScript.

3.0 MAIN CONTENT

3.1 Logical Statement

When writing a program, it may become necessary that some sets of statements to be executed are based on the outcome of a logical expression iN Comparsion and logical operators would be required. As the names connote, they allow for comparison of values. They are used with **if**, **while**, **switch**, and **for** statements to accomplish decision or iterative constructs in programming. One may be interested in testing if one operand is greater than, less than, equal to, or not equal to another operand. The majority of these operators will probably look familiar as in other programming languages. The result obtained is usually a true or false which further determines which statement the computer should execute.

Comparison Operators

Comparison operators are used in logical statements to determine equality or difference between variables or values. Given that Y=10.

***Table* 3.1**:explains the comparison operators:

Operators	Meaning	Example	Result
==	Equal to	Y==8	False
===	Equivalent to	Y=== 10 Y==="10"	True False
!=	Not Equal to	Y!=8	True
>	Greater than	Y>8	True
<	Less than	Y<8	False
>=	Greater or Equal to	Y>=8	True
<=	Less or Equal to	Y<=8	False

Logical Operators

Logical operators are used to determine the logic between variables or values. Given that X=5 and Y=10, the Table 3 explains the results of the use of logical operator in the expressions.

Table 3.2: Logical Operators

Operators	Meaning	Example	Result
&&	And	(x < 7 && y > 6)	False
||	Or	(x==5 || y==6)	True
!	Not	!(x==7)	True

3.2 Decision Making

One may wish to test the value of a variable, and perform different tasks based on the outcome of the test. For instance, one may need to check the examination score of a student to know whether he passed or failed and what grade he made. One can use conditional statements in one's code to achieve this. Conditional statements are used to perform different actions based on different conditions. The "**if** and **switch"** commands are commonly used to implement the conditional statement.

We shall briefly examine the different construct of the “**if** and **switch”** statements.

If Statement: This is used to execute some code only if a specified condition is true.

Syntax

```
If(condition)
{
 code to be executed if condition is true
}
```

Example 1

```
<script type="text/javascript">
var examscore = 80;
var result;
if (examscore >= 70)
 {
  result = “Pass”;
  document.write ("<b>Congratulation, You Passed </b>");
   }
</script>
```

If...else Statement

This is used to execute some codes if the condition is true and another code if the condition is false.

Syntax

```
if(condition)
 {
 code to be executed if condition is true
 }
else
 {
 code to be executed if condition is not true
  }
```

Example 2

```
<script type="text/javascript">
var examscore = 80;
var result;
if (examscore >= 45)
```

```
 {
 result = “Pass”;
  document.write("<b>Congratulation, You Passed </b>");
 }
 else
{
 result = “Fail”;
document.write("<b> You Failed, Try again</b>");
}
</script>
```

This will display the information “Congratulation, You passed.” **Practice 1**

If the value of exam score is 35, what message will be displayed on the web browser?

Switch Statement

This is used to select one of many blocks of code to be executed. The syntax of the switch statement is:

Syntax

```
switch(m)
{ case1:
executecodeblock1
break;
case2:
 executecodeblock2
 break;
.
.
.
casem:
executecodeblockm
break;

default:
code to be executed if m is different from case 1,Case 2, ...Case m }
```

It works by evaluating a single expression m (most often a variable). The value of the expression is then compared with the values for each case in the structure. If there is a match, the block of code associated with that case is executed. The **break** command is used to prevent the code from running into the next case automatically. We examine this by looking at a program that displays the day of the week based on a user selection.

Example 3

```
<script type="text/javascript">
      var dayoftheWeek;
      switch (dayoftheWeek)
      {
         Case 1: document.write("<b>Today is Sunday </b>");
          break;

      }
          Case 2: document.write("<b>Today is Monday </b>");
          break;
      {
          Case 3: document.write("<b>Today is Tuesday </b>");
          break;
      }
      {
          Case 4: document.write("<b>Today is Wednesday</b>");
          break;
      }

      {
          Case 5: document.write("<b>Today is Thursday </b>");
          break;
      }
      {
          Case 6: document.write("<b>Today is Friday</b>");
          break;
      }
      {
          Case 7: document.write("<b>Today is Saturday </b>");
          break;
      }
      {
         Default: document.write("<b>There are 7 Days in a week
          </b>"); break;
      }
       </script>
```

3.3 Iteration on JavaScript

Some statements in JavaScript are known as iterative statements. Instead of adding several almost equal lines in a script we can use loops to perform the tasks. Loop statements have control structures that delimit them and which determine how many times (zero or more) the delimited code is executed, based on some conditions.

We will look at two structures here:

- the “for statement”
- the “while statement” and its variants

The for Loop

The syntax of the “for statement” is

```
for (startvalue ; condition; increment ) {
statements;
}
```

Notice that there are three variables inside the for statement conditional expression.

They are

Startvalue : This holds the value of the initial state of the variable to be tested. It is usually done as an assignment.

Condition: The condition to be tested for. The statement keeps processing as long as it remains true.

Increment: The increment by which the variable being tested changes.

Example 4

```
<html>
<body>
<scripttype="text/javascript">
varnum=0;for(i=0;num<=100;num+)
{
document.write("The Next No is " + num);
document.write("<br />");
}
</script>
</body>
</html>
```

Example 4 defines a loop that starts with i=0. The loop will continue to run as long as **i** is less than, or equal to 100. **i** will increase by 1 each time the loop runs. The loop will generate integer numbers from 0 to 100 numbers.

The "while statement"

The "while statement" test a condition, and when true**,** repeatedly runs a block of code until the condition is no longer true.

The syntax is given as follows:

```
While (expression) {
Statements;
}
```

Another way to accomplish the task in example 4 is by using a while loop statement as shown in Example 5. The loop starts with i=0. The loop will continue to run as long as **i** is less than, or equal to 100. **i** will increase by 1 each time the loop runs:

Example 5

```
<html>
<body>
<scripttype="text/javascript">
varnum=0;
while(num<=100)
 {
 document.write("The Next number is " +num); document.write("<br
 />");;

 }
</script>
</body>
</html>
```

Example 6

The "do… while statement"

This is required when a block of code is to be run at least once. After running a block of code once, "do… while statement" evaluates the conditional expression. If the conditional expression is true, then it loops back to the beginning of the statement and starts again.

The syntax is as follow:

```
do {
statements;
}
While (expression);
```

Example 7

```
<html>
<body>
<scripttype="text/javascript">
varnum=0;
do
{
document.write("The next number is " + num);
document.write("<br />");
                    }
while (num<=10);
</script>
</body>
</html>
```

4.0 CONCLUSION

The normal execution of statements in a program is one after the other in the order in which they are written. This process is called sequential execution. Programmer can however, specify the order in which statements should be executed by using control constructs/statement. Some of these constructs have been covered in this units.

5.0 SUMMARY

In this unit, we have covered the basic statements required to implement Control Constructs in JavaScript. In the next unit, we shall cover events and events handlers.

6.0 TUTOR-MARKED ASSIGNMENT

i. Identify and correct the errors in following segments of code:
if (age >= 30); document.write (“Age greater than or equal to 30); else document.write(“Age is less than 30);

ii. Write a script that outputs HTML text that keeps displaying in the browser
window the multiples of the integer 2, namely 2,4, 8, 16, 32, 64, 128, etc. Ensure that your loop terminates when the value 2048576 is printed.

7.0 REFERENCES/FURTHER READING

Barbara, K. K. & Norman, J. M. (2001). *The World Wide Web: A Mass Communication Perspective*. USA: Mayfield Publishing Company.

Barrie, S. & Valda, H. (2004). *Programming the Web: An Introduction*. USA McGraw-Hill.

Behrouz, A. F. (2003). *Data Communications and Networking*. (3rd ed.). N.Y, USA: McGraw-Hill/Osborne.

Deitel, P. J., & Deitel, H. M. (2008).*Internet and World Wide Web: How to Program*. (4th ed.). New Jersey, USA: Pearson Prentice Hall.

Idowu, S.A., Maitanmi, S.O. & Adetunji, O. O. (2020). Introductory to Web Technology and Development. Nigeria: Jamiro Press

June, C. (2003). *The Unusually Useful Web Book*. USA:New Rider.

Marc, D. M. & Thomas, C. P. (2003) .*Web Design Using Macromedia Dreamweaver*. USA: McGraw-Hill/ Irwin.

Peter, W .(2001). *Instant Web Pages*. USA: Sybex Inc.

Raymond, G. & Ellen, H. (2001). *In-Line/On-Line Fundamentals of the Internet and World Wide Web.* USA: McGraw-Hill Learning Center.

Terry, F-M. (2009). *Web Development and Design Foundations with XHTML*. USA: Pearson.

Dave, D. (2002). *Schaum's Outline of HTML: Introduction to Web Page Design and Development*. USA: McGraw Hill

UNIT 4 EVENTS HANDLERS IN JAVASCRIPT

CONTENTS

1.0 INTRODUCTION

The word “event” as used in relation to computer programming usually signifies some sort of action or occurrence. As will be further discussed in this unit, an event refers to a repositioning of the mouse cursor, a mouse click, the filling of a form, or the pressing of the enter key. JavaScript lets one reacts to these events by specifying the relevant attribute in the object’s HTML tag called an event handler. To use an event handler, it has to be included in the HTML tag. Most times, a function is created to handle an event. A function is lines of JavaScript code that perform some action or action(s).

2.0 OBJECTIVES

At the end of this unit, you should be able to:

- implement JavaScript Popup Boxes
- explain the meaning of event and event handlers
- apply JavaScript Functions
- use JavaScript to implement events and event handlers.

3.0 MAIN CONTENT

3.1 JavaScript Popup Boxes

Popup boxes are used to display a message, along with an “OK” button. Depending on the popup box, it might also have a “Cancel” button, and one might also be prompted to enter some text JavaScript has three different types of popup box available for one to use. They are Alert box, Confirm box, and Prompt box.

a.) Alert Box

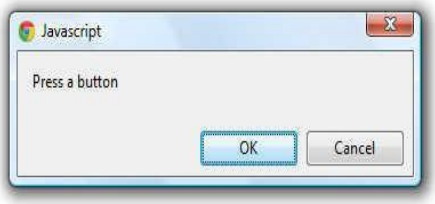
An alert box is often used if one wants to make sure information comes through to the user. When an alert box pops up, the user will have to click "OK" to proceed.

Syntax

```
alert("sometext");
```

Example 1

```
<html>
<head>
<script type="text/javascript">
function show_confirm()
{
var r=confirm("Press a button");
if (r==true)
{
alert("You pressed OK!");
}
else
{
alert("You pressed Cancel!");
}
}
</script>
</head>
<body>

<input type="button" onclick="show_confirm()" value="Show confirm
box" />

</body>
</html>
```



Press a button

Fig. 4.1: Alert

b.) Confirm Box

A confirm box is often used if one wants the user to verify or accept something. When a confirm box pops up, the user will have to click either “OK” or “Cancel” to proceed. If the user clicks “OK”, the box returns true. If the user clicks “Cancel”, the box returns false.

Syntax

```
confirm("sometext");
```

Example 2

```
<html>
<head>
<scripttype="text/javascript">
functionshow_confirm()
{
varr=confirm("Pressabutton");
if(r==true)
{
 alert("You pressed OK!");
}
else
{
 alert("YoupressedCancel!");
}
}
</script>
</head>
<body>
<input type="button" onclick="show_confirm()" value="Show confirm box"/>
</body>
</html>
```

c.) Prompt Box

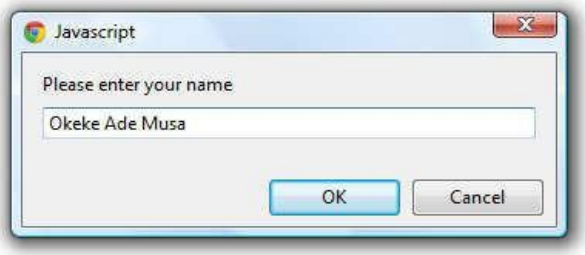
A prompt box is often used if the user is required to input a value before entering a page. When a prompt box pops up, the user will have to click either “OK” or “Cancel” to proceed after entering an input value. If the user clicks “OK”, the box returns the input value. If the user clicks “Cancel,” the box returns null.

Syntax

```
prompt("sometext","defaultvalue");
```

Example 3

```
<html>
<head>
<script type="text/javascript">
function show_prompt()
{
var name=prompt("Please enter your
name","Myname"); if (name!=null && name!="") {

 document.write("Hello " + name + "! You are Welcome!");
}
}
</script>
</head>
<body>
<input type="button" onclick="show_prompt()" value="Show prompt
box" />
</body>
</html>
```



Please enter your name

Okeke Ade Musa

Fig. 4.2: Prompt

3.2 JavaScript Functions

A function contains codes that will be executed by an event or by a call to the function. A function may be called from anywhere within a page (or even from other pages if the function is embedded in an external .js file). Functions can be defined both in the <head> and in the <body> section of a document. However, to assure that a function is read/loaded by the browser before it is called, it is wise to put functions in the <head> section.

How to Define a Function

Syntax

```
functionfunctionname(var1,var2,...,varX)
{
```

```
somecode
}
```

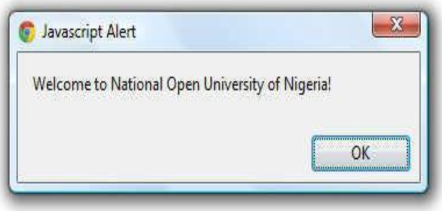
The parameters var1, var2, and so on are variables or values passed into the function. The {and the} defines the start and end of the function.

Note: A function with no parameters must include the parentheses () after the function name.

Note the word function is in lower case and when a call is made, it has to be spelt correctly.

Example 4

```
<html>
<head>
<scripttype="text/javascript">
functionnounmessage()
{
alert("Welcome to National Open University of Nigeria!");
}
</script>
</head>
<body>
<form>
<input type="button" value="Click me!" onclick=" nounmessage ()" />
</form>
</body>
</html>
```



Welcome to National Open University of Nigeria!

Fig. 4.3: Welcome

If the line: alert("Welcome to National Open University of Nigeria!!") in the example above had not been put within a function, it would have been executed as soon as the page was loaded. Now, the script is not executed before a user hits the input button. The function nounmessage () will be executed if the input button is clicked.

The Return Statement

The return statement is used to specify the value that is returned from the function. Therefore, functions that are going to return a value must use the return statement.

The example below returns the area of a rectangle that is, length * breadth

Example 5

```
<html>
<head>
<scripttype="text/javascript">
functionarea(length,breadth)
{
returnlength*breadth;
}
</script>
</head>
<body>
<scripttype="text/javascript">
document.write(area(10,15));
</script>
</body>
</html>
```

3.3 JavaScript Events

JavaScript programs do not have to be executed in sequence. We can make web pages more interactive by using events. These actions can be detected by JavaScript. A wide variety of events enables scripts to respond to the mouse, the keyboard, and other circumstances. Examples of events are:

- A web page or an image loading
- Mouse click
- Mouse over a hot spot on the web page
- Selecting an input field in an HTML form
- Submitting an HTML form
- A keystroke

The script that is used to detect and respond to an event is called an event handler. Event handlers are among the most powerful features of JavaScript.

3.4 Events Handlers

In JavaScript/HTML, an event handler attaches JavaScript to your HTML elements. Event handlers allow a web page to detect when a given “event” has occurred, so that it can run some JavaScript code. In one’s code, an event handler is simply a special attribute that one adds to an HTML element. For example, to run some JavaScript when the user clicks on an element, add the onClick attribute to the element. More examples of event handlers are presented in Table 4.1.

Table 4.1: More Examples of Event Handlers

Event	Description
onclick:	Use this to invoke JavaScript upon clicking (a link, or form boxes)
onload:	Use this to invoke JavaScript after the page or an image has finished loading.
onmouseover:	Use this to invoke JavaScript if the mouse passes by some link
onmouseout:	Use this to invoke JavaScript if the mouse goes pass some link
onunload:	Use this to invoke JavaScript right after someone leaves this page.
onSubmit	The onSubmit event is used to validate ALL form fields before submitting it.
onFocus, onBlur and onChange	The onFocus, onBlur and onChange events are often used in combination with validation of form fields.

4.0 CONCLUSION

One very simple response to an event is to display a dialog box. JavaScript provides three types of dialog boxes: alert box, confirmation box, and prompt box. Events allow scripts to respond to a user who is moving the mouse, entering form data or pressing keys. Events and event handlers help to make web application more responsive, dynamic and interactive.

5.0 SUMMARY

Event such as the onclick and onsubmit events can be used to trigger scripts. JavaScript events, which allow scripts to respond to users' interaction and modify the pages, accordingly have been discussed in this unit.

6.0 TUTOR-MARKED ASSIGNMENT

i. Name three JavaScript event handlers and describe how they are used. Create a Web page that incorporates them.
ii. What are some practical uses of alert boxes?

7.0 REFERENCES/FURTHER READING

Barbara, K. K. & Norman, J. M. (2001). *The World Wide Web: A Mass Communication Perspective*. USA: Mayfield Publishing Company.

Barrie, S. & Valda, H. (2004). *Programming the Web: An Introduction*. USA McGraw-Hill.

Behrouz, A. F. (2003). *Data Communications and Networking*. (3rd ed.). N.Y, USA: McGraw-Hill/Osborne.

Deitel, P. J., & Deitel, H. M. (2008). *Internet and World Wide Web: How to Program*. (4th ed.). New Jersey, USA: Pearson Prentice Hall.
Idowu, S.A., Maitanmi, S.O. & Adetunji, O. O. (2020). Introductory to Web Technology and Development. Nigeria: Jamiro Press

June, C. (2003). *The Unusually Useful Web Book*. USA:New Rider.

Marc, D. M. & Thomas, C. P. (2003) .*Web Design Using Macromedia Dreamweaver*. USA: McGraw-Hill/ Irwin.

Peter, W .(2001). *Instant Web Pages*. USA: Sybex Inc.

Raymond, G. & Ellen, H. (2001). *In-Line/On-Line Fundamentals of the Internet and World Wide Web*. USA: McGraw-Hill Learning Center.

Terry, F-M. (2009). *Web Development and Design Foundations with XHTML*. USA: Pearson.

Dave, D. (2002). *Schaum's Outline of HTML: Introduction to Web Page Design and Development*. USA: McGraw Hill

MODULE 4

UNIT 1 OVERVIEW OF XML

CONTENTS

1.0 INTRODUCTION

Extensible Markup Language (XML) is a set of rules for encoding documents in machine-readable form. It was developed in 1996 by the World Wide Web Consortium's (W3C's) XML working group. As a widely supported open technology for describing, storing and sharing data, Web application uses XML extensively and current versions of most browsers provide huge support for the language. The primary design goals of XML emphasise simplicity, generality, and usability over the Internet. Today, XML has grown to become the most common tool for data transmissions between all sorts of applications. It is a textual data format with strong support via Unicode for the world's languages. Though XML focuses on documents, it is widely used for the representation of arbitrary data structures for example, Web services. There are many applications programming interfaces (APIs) used to process XML data and several schema systems exist to aid in the definition of XML-based languages. We shall learn more about XML in this unit.

2.0 OBJECTIVES

At the end of this unit, you should be able to:

- explain the meaning of XML
- show how to create and modify XML document
- illustrate how to process and validate XML document
- describe how to view XML documents in a Web browser.

3.1 XML Basics

XML permits documents authors to create markup (that is, text-based notation for describing data) for virtually any type of information. This enables document authors to create entirely new markup languages for describing any type of data, such as mathematical formula, software configuration instructions, chemical molecular structures, music, new recipes and financial reports. XML describes data in a way that both human beings and computer can understand. XML is not a replacement for HTML. HTML is about displaying information, while XML is about carrying information. XML uses tags to structure data. The tags are not predefined- every developer is expected to define his/her tags. XML is designed to be self-descriptive. Tags are markup construct that begins with "<" and ends with ">". Tags come in three flavours: *start-tags*, for example <section>, *end-tags*, for example </section>, and *empty-element tags*, for example <line-break />. An element's start and end tags enclose text that represents a piece of data. Every XML document must have exactly one root element that contains all the other elements. XML documents may begin by declaring some information about themselves, as in the following example.

```
<?xml version="1.0" encoding=" ISO-8859-1" ?>
```

Now let us take a look at this simple XML code below:
Example 1:

```
<?xml version="1.0" encoding="ISO-8859-
 1"?> <MyPersonalDetails>
          <FullName>
                    <FirstName>Musa </FirstName>
                    <LastName>Okeke</LastName>
          </FullName>
          <BirthDate>
                    <Month>June</Month>
                    <Date>17</Date>
                    <Year>1970</Year>
          </BirthDate>
```

```
      <MailingAddress>
            <University>National Open University of Nigeria
</University>
            <School>Science and Technology</School>
            <Programme>Computer Science</Programme>
      </MailingAddress>
</MyPersonalDetails>
```

From the codes above, XML did nothing at all. It is just information wrapped in tags. Someone must write a piece of software to send, receive or display it. The first line of code tells the version and character encoding being used by this XML document. The second line of code tells what kind of information or XML document. The XML applications that will use the codes in example 1, will looked at the root or parent tag in the XML document. Here, it is <MyPersonalDetails >, which is not defined by XML. XML allows authors to create their own XML tag to be used in each document. XML, like any other languages, is capable of having two or more child tags or commonly known as nested tags. The <FullName> tag has three child tags, so on and so forth. Also, XML tags are case sensitive. Meaning we cannot declare < MyPersonalDetails > opening tag with a closing tag of </myPersonalDetails >. Opening and closing tags must be written with the same case:

3.2 Creating and Modifying XML Documents

XML allows one to describe data precisely in a well-formed format. XML document are highly portable. Any text editor such notepad of software that supports ASCII/Unicode characters can open XML documents for viewing and editing. An XML document is created by typing XML codes into a text editor and then save the document with a filename and a .xml extension. Most Web browsers can display XML documents in a formatted manner that shows the XML's structure.

3.3 Processing XML Documents

To process an XML document, you would need an XML parser (or XML processor). A parser is software that checks that the document follows the syntax rules specified by the W3C's XML recommendation and makes the document's data available to application. A parser would for example check an XML document to ensure that there is a single root element, a start tag for each element, and properly nested tags (that is, the end tag for a nested element must appear before the end tag of the enclosing element). Furthermore, XML is case sensitive, so the proper capitalisation must be used in elements as in Example 1. A document that conforms to this syntax is said to be a well-formed XML document and is syntactically correct. If an XML parser can process an XML document successfully, that XML document is well-formed. Parsers can provide access to XML-encoded data in well-formed document only. Often XML parsers are built into software or available for download over the Internet. Examples of parser include Microsoft XML Core Services (MSXML), Xerces Expat and so on.

3.4 Validating XML Documents

In addition to being well formed, an XML document may be valid. This means that it contains a reference to a Document Type Definition (DTD) and that its elements and attributes are declared in that DTD and follows the grammatical rules for them that the DTD specifies. A DTD is an example of a schema or grammar. Since the initial publication of XML 1.0, there has been substantial work in the area of schema languages for XML. Such schema languages typically constrain the set of elements that may be used in a document, which attributes may be applied to them, the order in which they may appear, and the allowable parent/child relationships.

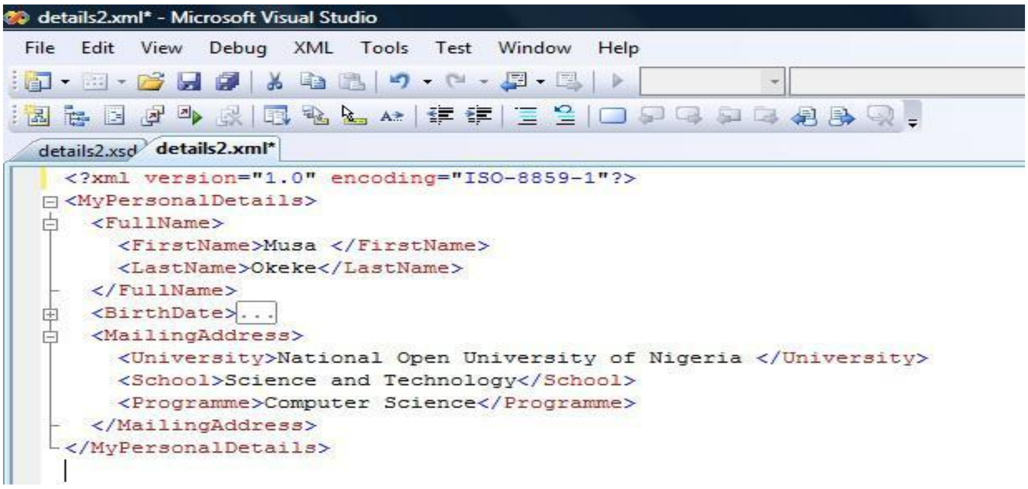
When an XML document references DTD or a schema, some parsers (called validating parsers) can read the DTD/Schema and check that the XML conforms to the DTD/Schema, the XML document is valid. For example, if in Figure 2.1 we were referencing DTD that specifies that BirthDate element must have Month, Date and Year, then the exclusion of Year element would invalidate the XML document detail2.xml. However, the XML document would still be well formed, because it follows proper XML syntax (that is, it has one root element, each element has a start tag and an end tag, and the element are nested properly). By definition, a valid XML document is well formed. Parsers that cannot check for document conformity against DTDs/schemas are nonvalidating parsers- they determine only whether an XML document is well-formed, not whether it is valid. Schema are XML documents themselves, whereas DTDs are not. XML processors are classified as validating or non-validating depending on whether or not they check XML documents for validity. A processor that discovers a validity error must be able to report it, but may continue normal processing.

3.5 Formatting and Manipulating XML Documents

XML document can be manipulated to appear differently on several devices. For example, the way XML document renders on Personal Digital Assistants (PDAs) is different from Desktop computers. Most XML documents contain only data. They do not include formatting instructions, so applications that process XML documents must look for how to process, manipulate or display the data. Extensible Stylesheet Language (XSL) can be used to specify rendering instructions for different platforms. XML-processing programs can also search, sort and manipulate XML data using XSL. Other popular XML-related technologies are: XPath XML Path Language (XPath), which is used for accessing parts of an XML document, XSL Formatting Objects (XSL-FO), which is a XML vocabulary used to describe document formatting, and XSL Transformations-language (XSLT) used for transforming XML documents into other documents.

3.6 Viewing an XML Document in Web Browser

Example 1 shows a simple listing of a text file for detail2.xml. This document does not contain formatting information for the detail2.xml. This is because XML is a tool for describing the structure, storage and transferring of data across disparate format/sources. Formatting and displaying data from an XML document is achieved in different ways within specific application platform. For instance, when the user loads detail2.xml in the Internet Explorer, MSXML (Microsoft XML Core Services) or Firefox, it will be parsed and display the document data. Each browser has a built-in style sheet to format the data. Note that the resulting format of the data in Example 1 is similar to the format of the listing in Figure 2.1. The XML document will be displayed with colour-coded root and child elements. A plus (+) or minus sign (-) to the left of the elements can be clicked to expand or collapse the element structure. To view the raw XML source (without the + and - signs), select “View Page Source” or “View Source” from the browser menu. Although these symbols are not part if the XML document, both browser place them next to every container element. A minus sign indicates that the browser is displaying the container element child element. Clicking the minus sign next to an element collapses that element (that is, it causes the browser to hide the container element’s children) and replace the minus sign with a plus). Conversely, clicking the plus sign next to an element expands the elements (that is, it causes the browser to display the container elements children and replace the plus sign with a minus sign). This behaviour is similar to viewing the directory structure on one’s system in Windows Explorer or another similar directory viewer. In fact, a directory structure often is modelled as a series of tree structure in which the root of the tree represents a disk drive for instance C: and nodes in the tree represent directories. Parsers often store XML data as tree structure to facilitate efficient manipulation as discussed.


details2.xml* - Microsoft Visual Studio
File Edit View Debug XML Tools Test Window Help
details2.xsd
details2.xml*

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<MyPersonalDetails>
  <FullName>
    <FirstName>Musa </FirstName>
    <LastName>Okeke</LastName>
  </FullName>
  <BirthDate>...
  <MailingAddress>
    <University>National Open University of Nigeria </University>
    <School>Science and Technology</School>
    <Programme>Computer Science</Programme>
  </MailingAddress>
</MyPersonalDetails>
```

Fig. 1.1: View of XML file in a Browser

4.0 CONCLUSION

Within the last two decades of the introduction of XML, it has been used to create hundreds of languages which include XHTML ,WSDL for describing available web services, WAP and WML as markup languages for handheld devices, RSS languages for news feeds, RDF and OWL for describing resources and ontology, SMIL for describing multimedia for the web etc. In addition, XML-based formats have become the default for most office-productivity tools, including Microsoft Office (Office Open XML) and Apple's iWork.

5.0 SUMMARY

XML describes data in a way that both human beings and computer can understand. It enhances the storage and exchange of data amongst disparate computer systems. In this unit, we covered how to create, modify, validate, format, process and view XML documents in a browser.

6.0 TUTOR-MARKED ASSIGNMENT

i. Discuss the importance of XML technologies.
ii. State which of the following statements is true(T) or false(F):
    a XML can be used to create other markup languages(T/F)
    b. Parsers are used to check the syntax of an XML document (T/F)
    c. XML document is not case sensitive (T/F)
    d. All XML start tags must have corresponding end tags (T/F)

7.0 REFERENCES/FURTHER READING

Barbara, K. K. & Norman, J. M. (2001). *The World Wide Web: A Mass Communication Perspective*. USA: Mayfield Publishing Company.

Barrie, S. & Valda, H. (2004). *Programming the Web: An Introduction*. USA McGraw-Hill.

Behrouz, A. F. (2003). *Data Communications and Networking*. (3rd ed.). N.Y, USA: McGraw-Hill/Osborne.

Deitel, P. J., & Deitel, H. M. (2008). *Internet and World Wide Web: How to Program*. (4th ed.). New Jersey, USA: Pearson Prentice Hall.

Idowu, S.A., Maitanmi, S.O. & Adetunji, O. O. (2020). Introductory to Web Technology and Development. Nigeria: Jamiro Press

June, C. (2003). *The Unusually Useful Web Book*. USA:New Rider.

Marc, D. M. & Thomas, C. P. (2003) .*Web Design Using Macromedia Dreamweaver*. USA: McGraw-Hill/ Irwin.

Peter, W .(2001). *Instant Web Pages*. USA: Sybex Inc.

Raymond, G. & Ellen, H. (2001). *In-Line/On-Line Fundamentals of the Internet and World Wide Web*. USA: McGraw-Hill Learning Center.

Terry, F-M. (2009). *Web Development and Design Foundations with XHTML*. USA: Pearson.

Dave, D. (2002). *Schaum's Outline of HTML: Introduction to Web Page Design and Development*. USA: McGraw Hill

UNIT 2 SEARCH ENGINES & TOOLS

CONTENTS

1.0 INTRODUCTION

Quite a number of collections of search tools are available today that allow users to find information on the Web quickly and easily. Two basic approaches have evolved in response to the need to organise and to locate information on the World Wide Web. These are directories and search engines. A directory offers a hierarchy representation of hyperlinks to Web pages and presentation broken down into topics and subtopics. On the other hand, a search engine is a set of programs that is used to search for information within a specific realm and collate that information in a database. Although search engine is really a general class of programs, the term is often used to specifically describe Internet search engines like Google, Alta Vista and Excite. They enable users to search for documents on the World Wide Web, FTP servers and USENET newsgroups. Search engines can also be devised for offline content, such as a library catalogue, the contents of a personal hard drive, or a catalogue of museum collections. Generally search engines help people to organise and display information in a way which makes it readily accessible.

2.0 OBJECTIVES

At the end of this unit, you should be able to:

- explain the meaning of Search Tools
- describe the components of a Search Engine
- discuss how search engines work
- show how to popularise a site.

3.0 MAIN CONTENT

3.1 Search Tools

A search tools is software that enables a user to quickly and easily gain access to information. The collection of search tools is constantly evolving with new ones coming on the scene and others disappearing. In this unit, we shall consider two basic approaches that have evolved in response to the need to organise and locate information on the World Wide Web: directories and search engines. Both approaches allow information about Web pages that is contained in some database that already has been created either manually or using special programs that search the Web pages to be assess quickly and easily. A request for information is answered by the search tool retrieving the information from its already-constructed database of indexed Web details. Other definitions that relate to searching information on the Web are as follows:

Search Terminology

Search tool: This refers to any mechanism for locating information on the Web. Examples include search or metasearch engine, and directory.

Metasearch engine: This refers to an all-in-one search engine that performs a search by calling on more than one other search engine to do the actual work.

Query: This refers to the information entered into a form on a search engine's Web page that describes the information being sought.

Query Syntax: This term is used to describe, the set of rules describing what constitute a legal query on some search engines, special symbols may be used in a query.
Query Semantic: This term is used to describe a set of rules that defines the meaning of a query.

Hit: This refers to a URL that a search engine returns in response to a query.

Match: This is a synonym for hit.

Relevancy score: This refers to a value that indicates how close a match, a URL was to a query; usually expressed as a value from 1 to 100, with the higher score meaning more relevant.

3.1.1 Directories

The first method of finding and organising Web information is stated earlier is the directory approach. A directory offers a hierarchy representation of hyperlinks to Web pages and presentation broken down into topics and subtopics. The hierarchy can descend many levels. The specific number of levels is determined by the taxonomy of topics.

Examples of popular general directories

www.google.com
www.looksmart.com
www.lycos.com
www.dmoz.com
www.yahoo.com

3.1.2 Search Engines

The second approach to organising information and locating information on the Web is a search engine, which is a computer program that does the following:

1. allows a user to submit a form containing a query that consists of a word or phrase describing the specific information of interest to be located from the Web

2. searches its database to try to match your query

3. collate and returns a list of clickable URLs containing presentations that match the user's query; the list is usually ordered with the better matches appearing at a the top

4. permits a user to revise and resubmit a query.

A recent survey ranking the market share of web search engine carried out by Rapid app 2020, showed

- Google is 90%,
- Bing is 75%
- Yahoo is 6.69%,
- Baidu is 3.39%,
- Yandex is 3.29% and
- Other is 1.98%.

3.2 Components of a Search Engine

Search engines have the following components:

a) User Interface
b) Databases

c) Robot or Spider Software

a) User Interface

The user interface is a mechanism by which users submit queries to the search engine by typing a keyword or phrases to search into the text box. When the form is submitted, the data typed into the text box is sent to a server-side script that searches the database using the keywords entered. Afterwards, search results are displayed in the browser containing a list of information, such as the URLs for Web pages that meet the users' criteria. This result set is formatted with a link to each page along with additional information that might include the page title, a brief description, the first few lines of text, or the size of the page and a relevancy score for each hit. This way, the user is able to make an informed choice as to which hyperlinks to follow. Hyperlinks to help files are usually displayed prominently, and advertisement should not hinder a reader's use of the search engine. The order in which pages are displayed may depend on paid advertisement, alphabetical order, and link popularity. Each search engine has its own policy for ordering the search results. The policies can change over time.

b) Database

A database is a collection of information organised so that its contents can easily be accessed, managed and updated. Databases management systems (DBMSs) such as Oracle, Microsoft SQL Server, Informix, MySQL or IBM DB2 are used to configure and manage the database. The databases associated with search engines are extremely large indexed pages that require a highly efficient search strategy to retrieve information from them. Computer scientists have spent years developing efficient several searching and sorting strategies, which are implemented in the search. The information displayed as results of your search is usually from the database accessed by the search engine site. Some search engines, such as AOL and Netscape use a database provided by Google.

c) Robot

A robot (sometimes called a spider) is a program that automatically traverses the hypertext structure of the Web by retrieving a Web page document and following the hyperlinks on the page. It moves like a robot spider on the Web, accessing and documenting Web pages. It requests pages from a website in the same way as Microsoft Explorer, or Firefox and any other browser does it. Spider does not collect images or formatting details. It is only interested in text and links and the URL from which they come. The spider categorises the pages and stores information about the Web site and the Web pages in a database. Various robots may work differently, but in general, they access and may store important information on web pages such as title, meta tag keyword, meta tag description, and some of the text on the page (usually either the first few sentences of the text contained in the heading tags). For multimedia elements in web pages to be indexed, the "alt" tag should be used in order to have values in the search engines. The spider software works in conjunction with the index software. This uses the information collected by the spider. The spider takes the information it has gathered about a web page and sends it to the index software where it is analysed and stored. The index makes sense of the mass of text, links and URLs using an algorithm, which refers to a complex mathematical formula that indexes the words, the pairs of words and so on. The algorithm analyses the pages and links for word combinations to determine what the web pages are all about that is, what topics are being covered. Then, scores are assigned that allow the search engine to measure how relevant or important the web pages (and URLs) might be to the user or visitor. Major search engines such as Google, Yahoo or Bing use proprietary algorithm for scoring.

3.3 Listing in a Search Engine and Search Index

The components of a search engine (robot, database and search form) work together to obtain information about Web pages, store information about Web pages, and provide a graphical user interface to facilitate searching for and displaying a list of Web pages relevant to given key words. In recent times, search engines have become one of the top methods used to drive traffic to ecommerce sites. Though very effective, it is not always easy to get listed in a search engine or search directory. Recently, there is a trend away from free listing in search engines. Current trends entail paying for listing consideration in a search engine or directory. These approaches include an express submit or express inclusion, paying for preferential placement in search engine displays (called sponsoring or advertising), and paying each time a visitor clicks the search engine's link to your site. Yahoo and Google use the terms Calls its Sponsor Results and Google AdWords respectively. In these programs, payment is made when the site is submitted for review. If accepted, the site has a listing usually at the top or right margin of the search results. In addition to the initial fee, the Web site owners must pay each time a visitor clicks on the search engine link to their site-this is called a cost-per-click (CPC).

4.0 CONCLUSION

A web search engine is designed to search for information on the World Wide Web, FTP servers USENET newsgroup, and so on. The search results, which may consist of web pages, images, information and other types of files, are generally presented in a list of results and are often called hits. Some search engines also mine data available in databases or open directories. Unlike web directories, which are maintained by human editors, search engines operate algorithmically or are a mixture of algorithmic and human input.

5.0 SUMMARY

Search engines use automated software programs to survey the Web and build their databases. Web documents are retrieved by these programs and analysed. Data collected from each web page are then added to the search engine index. Each search engine uses a proprietary algorithm to create its indices such that, ideally, only meaningful results are returned for each query. The best URLs are then returned to the user as hits, ranked in order with the best results depending on the algorithm used by the search engine at the top.

6.0 TUTOR-MARKED ASSIGNMENT

i. Describe the components of a search engine.
ii. Describe how Google search engine works.
iii. What is the meaning of Google AdWords?

7.0 REFERENCES/FURTHER READING

Barbara, K. K. & Norman, J. M. (2001). *The World Wide Web: A Mass Communication Perspective*. USA: Mayfield Publishing Company.

Barrie, S. & Valda, H. (2004). *Programming the Web: An Introduction*. USA McGraw-Hill.

Behrouz, A. F. (2003). *Data Communications and Networking*. (3rd ed.). N.Y, USA: McGraw-Hill/Osborne.

Deitel, P. J., & Deitel, H. M. (2008). *Internet and World Wide Web: How to Program*. (4th ed.). New Jersey, USA: Pearson Prentice Hall.
Idowu, S.A., Maitanmi, S.O. & Adetunji, O. O. (2020). Introductory to Web Technology and Development. Nigeria: Jamiro Press

June, C. (2003). *The Unusually Useful Web Book*. USA:New Rider.

Marc, D. M. & Thomas, C. P. (2003) .*Web Design Using Macromedia Dreamweaver.* USA: McGraw-Hill/ Irwin.

Peter, W .(2001). *Instant Web Pages*. USA: Sybex Inc.

Raymond, G. & Ellen, H. (2001). *In-Line/On-Line Fundamentals of the Internet and World Wide Web.* USA: McGraw-Hill Learning Center.

Terry, F-M. (2009). *Web Development and Design Foundations with XHTML*. USA: Pearson.

Dave, D. (2002). *Schaum's Outline of HTML: Introduction to Web Page Design and Development*. USA: McGraw Hill

UNIT 3 THE FUTURE WEB, TECHNOLOGIES AND DEVELOPMENT TOOLS

CONTENTS

1.0 INTRODUCTION

Advances in Internet technology have led to the release of several tools for Web development. Many of the tools are easy to use and made available to the public as open source to aid in development. A popular example is the LAMP (Linux, Apache, MySQL, PHP) stack, which is usually distributed free of charge. The availability of free tools has greatly influenced the rate at which many people around the globe setup new Web sites daily. Easy to use software for Web development include amongst others: Adobe Dreamweaver, Netbeans, WebDev, or Microsoft Expression Studio, Adobe Flex, and so on. By using these software, virtually anyone can develop a Web page in a matter of minutes. Knowledge of Hypertext Markup Language (HTML) or other programming language is not usually required, but is recommended for professional results. Newer generation of web development tools use the strong growth in LAMP, Java Platform, Enterprise Edition technologies and Microsoft .NET technologies to provide the Web as a way to run applications online. Web developers now help to deliver applications as Web services, which were traditionally only available as applications on a desk, based computer. Thus, instead of running executable code on a local computer, users can now interact with online applications to create new contents. This has enabled new methods in communication and allowed for many opportunities to decentralise information and media distribution. In this unit, we shall discuss other technologies, models and tools that enhance easy development of Web applications.

2.0 OBJECTIVES

At the end of this unit, you should be able to:

- describe the tools used for Web applications
- explain the meaning and usefulness of Artificial intelligence to web application
- classify and use Web development tools
- discuss current and emerging trends in Web application development.

3.0 MAIN CONTENT

3.1 ARTIFICIAL INTELLIGENCE

Artificial intelligence (AI) has been described as the future of the world. This has no restriction to field specification as showed in several decades. This is seen when physical strength is being replaced by robots, and the greatest minds began to work on the full automation of our lives.

It is possible to ignore the existence of this for a while in automating the production of products, consumer goods, cars and other things. Either we like it or not, AI is sneaking into our lives much deeper than we think. A typical example of the application of AI is the smart home, voice search, home cleaning robots, autopilot among others. Each of them already has its embodiment in the market.

3.2 PROGRAMMING LANGUAGES

This is essential for web developers both professionals and beginners. Their demand directly depends on what frameworks and languages they can work with. They need to understand the required knowledge needed to get the job going. Some of the current tools in web developments are discussed.

JavaScript, Python and PHP are the most popular programming languages in recent years. The most important in web development is JavaScript. Python follows in order of importance but really weak in mobile applications. However, for browser web services this language is one of the best. PHP is considered one of the best backend languages in web application. It is very easy to learn and use. It is also suitable for creating 80% of simple web services.

3.3 FRAMEWORKS

This is part of the development environment. Typical example includes the React and Vue which are the basis of all front-end developments. React is today the most sought-after framework in most companies. For a long time, it competed with Angular. But the developers claimed that Angular sags significantly in efficiency. Vue recently became popular. However, it has captured the Asian market. This is expected to meet up with React in times of popularity and use. Xiaomi is another well known brands that has completely switched to using Vue.

3.4 VOICE COMMANDS

Voice command is one of the applications of AI. It was not previously noticeable because the functionality was not perfect. Products with high quality voice input were disproportionally expensive. Voice search is not as fast as the text input because the browser needs time for a clear speech recognition. However, it takes on a new look and influences the development of most web services as seen in figure 3.4.

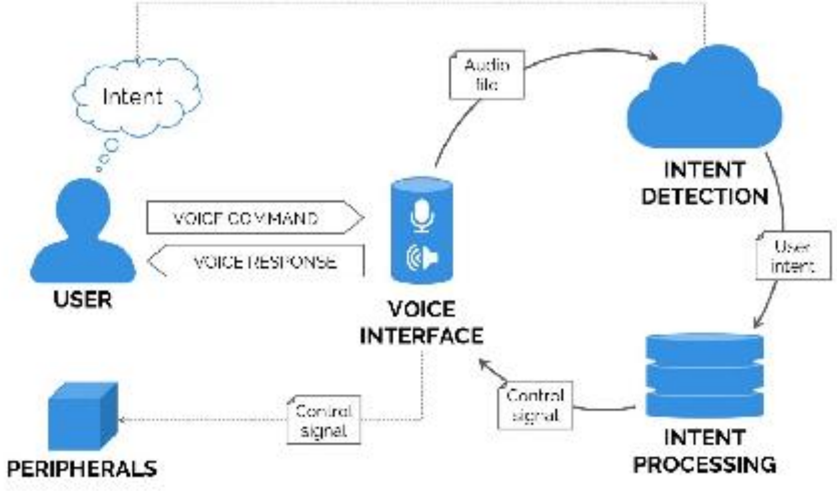


Fig 3.4 The mechanism of voice input

3.5 INTERACTIVITY

Interactivity if one of the influences of web development. The task of the developer is to make the most convenient and practical use of all applications. That is why web services will be more interactive.

One example of an interactive application is the Genesis car dealer website. Instead of flipping through hundreds of pages, the user can explore the car with a mouse click. The functionality allows you to collect your car through; colour selection, equipment selection and wheel selection. In short, the buyer will be able to change everything that does not affect the characteristics of the car.

3.6 ADAPTABILITY

Decades ago, we visit web pages only from a computer. Recently, the use of smartphones and tables has taken over because their functionality and performance is much higher. In web applications, mobile phones are not inferior to computers. The only difference is the screen size. Ease and convenience of the graphics should not suffer because of different monitors as graphically seen in figure 3.6.

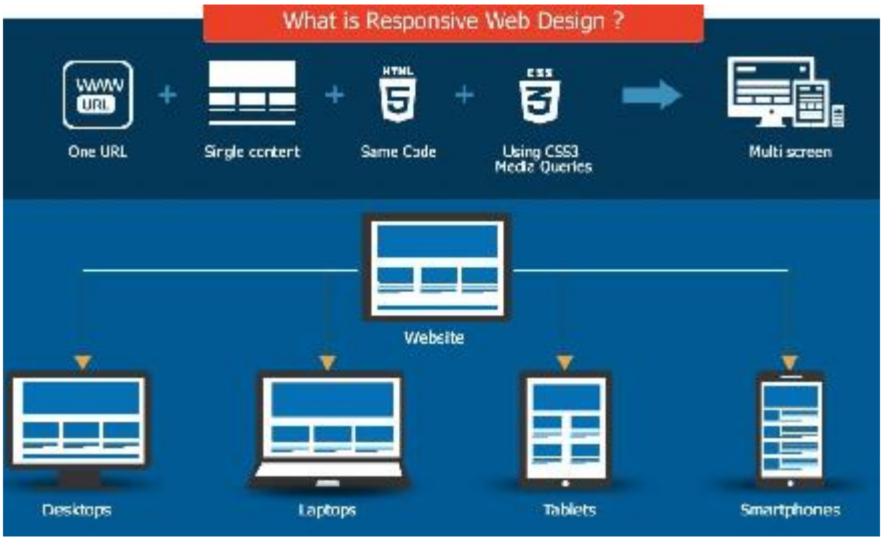


Fig 3.6 Principles of responsive web development

3.7 ONE-PAGE SITES

The effectiveness of single page site was noticed because of its benefits. Such benefit include ability to save development resources, its faster mode of loading among others. Modern users appreciate their time, so wandering around the site without achieving their aims may be frustrating. Also, the correct placement of data on a one-page website is an outstanding benefit in web development which attracts the attention of clients in a matter of seconds. Similarly, buyers are interested in products that they understand in 5 seconds thank products that they will study in several hours.

From the above, Service providers publish services to a service broker. Service requesters find required services using a service broker and bind to them.

3.2 Cloud Computing

Cloud computing refers to the use and access of multiple server-based computational resources via a digital network (WAN, Internet connection using the World Wide Web, and so on). Cloud users may access the server resources using a computer, netbook, pad computer, smart phone, PDA, or other devices. In cloud computing, applications are provided and managed by the cloud server and data are stored remotely in the cloud configuration. Users do not download and install applications on their own device or computer; all processing and storage is maintained by the cloud server. The on- line services are usually offered by a cloud provider or by a private organisation. Before the advent of cloud computing, tasks such as using word processing would not be possible without the installation of application software on a user's computer. A user would need to purchase a license for each application from a software vendor and obtained the right to install the application on one computer system. As computer technologies advanced, local area networks (LAN) and more networking capabilities, the client-server model of computing were born, where server computers with enhanced capabilities and large storage devices could be used to host application services and data for a large workgroup. In a client-server computing environment, a network-friendly client version of the application was required on client computers, which utilised the client system's resources (memory and CPU for processing), even though the resultant application data files (such as word processing documents) were stored centrally on the data servers. In this case, many users on a network purchased multiple user licenses of an application for use. Cloud computing differs from the classic client-server model discussed in module one of this course material, by providing applications from a server that are executed and managed by a client's web browser, with no installed client version of an application required. Cloud computing provides computation, software, data access, and storage services that do not require end-user knowledge of the physical location and configuration of the system that delivers the services. One may compare this scenario with the concept drawn from the electricity grid, wherein end-users consume power without needing to understand the component devices or infrastructure required to provide the service. The reason behind centralisation is to give cloud service providers complete control over the versions of the browser-based applications provided to clients, which removes the need for version upgrades or license management on individual client computing devices. In addition, cloud computing describes a new supplement, consumption, and delivery model for information technology service based on Internet protocols, and in most cases typically involves provisioning of dynamically scalable and often virtualised resources. It is a by-product and consequence of the ease-of-access to remote computing sites provided by the Internet. This may take the form of web-based tools or applications that users can access and use through a web browser as if they were programs installed locally on their own computers. The term “Software as a service” (SaaS) is at times used to describe application programs offered through cloud computing.

3.3 Blogs

A blog is a blend of the term “Web log.” It is a type of Website or part of a Website. Many blogs provide commentary or news on a particular subject; others function as more personal online diaries. A typical blog combines text, images, and links to other blogs, Web pages, and other media related to its topic. The ability of readers to leave comments in an interactive format is an important part of many blogs. Most blogs are primarily textual, although some focus on art (art blog), photographs (photoblog), videos (video blogging), music (MP3 blog), and audio (podcasting). Microblogging is another type of blogging, featuring very short posts. Most blogs are interactive, allowing visitors to leave comments and even communicate with each other via widgets on the blogs. This interactivity distinguishes them from other static websites. Entries are commonly displayed in reverse-chronological order. Many blogs are hosted at blog communities such as http://blogspot.com.

3.4 RSS

Really Simple Syndication or Rich Site Summary (RSS) is commonly used to create newsfeed from blog postings and other Web sites. The RSS feeds contain a summary of new items posted to the site. Web feeds benefit publishers by letting them syndicate content automatically. They benefit readers who want to subscribe to timely updates from favoured websites or to aggregate feeds from many sites into one place. RSS feeds can be read using software called an “RSS reader”, “feed reader”, or “aggregator”, which can be web-based, desktop-based, or mobile-device-based. Some browser, such as Firefox, Safari, and Internet 7 can display RSS feeds. A standardised XML file format allows the information to be published once and viewed by many different programs. The user subscribes to a feed by entering into the reader the feed’s

URL or by clicking a feed icon in a Web browser that initiates the subscription process. The RSS reader checks the user's subscribed feeds regularly for new work, downloads any updates that it finds, and provides a user interface to monitor and read the feeds. RSS allows users to avoid inspecting all of the websites they are interested in manually, and instead subscribe to Websites such that all new content is pushed onto their browsers when it becomes available. By providing up-to-date, linkable content for anyone to use, RSS enables website developers to draw more traffic. It also allows users to get news and information from many sources easily and reduces content developers time. RSS simplifies importing information from portals, weblogs and news sites. Any piece of information can be syndicated via RSS, not just news.

3.5 Podcasts

Podcasts are typically audio files, delivered by an RSS feed on the Web. They may also be made available by recording an MP3 file and providing a link on a Web page. They usually would take the format of an audio blog, interview or radio show. These files can be saved to your computer or to an MP3 player (such as iPod) for later listening.

3.6 Wiki

A wiki is a Web site that allows immediate update by visitors using a simple form on a Web page at any time. Some wikis are designed to serve a small group of people such as the members of an organisation. The most powerful and popular wiki is Wikipedia which is accessible at the URL (http:://Wikipedia.org). It is an online encyclopaedia, which can be updated by any registered user at anytime. Wiki is a form of social software in action where visitors sharing their collective knowledge can create a resource freely used by all. Though there have been isolated cases of practical jokes and occasionally inaccurate information posted at Wikipedia, the information and resources provided is still good enough as starting point when exploring a topic.

3.7 Microformat

Microformat is a standard format for representing information aggregate that can be understood by computers thereby enabling easier access and retrieval of information. It could also lead to new types of applications/services on the Web. Some people consider the web as containing loose information while others see logical aggregates, business cards, resume, events, etc. The need to organise information on the Web cannot be overemphasised. Microformat standard encourage sites to organise their information such that its increases interoperability and accessibility. For example, if one wants to create an event or an events calendar, one could use the hCalalender microformat. Some other available microformats are the adr for address information, hresume for resume and xfolk for collections of bookmarks. These all allow new services to be created with ease.

3.8 Resources Description Framework (RDF)

The Resource Description Framework (RDF), developed by the World Wide Web consortium (W3C) is one way of making the Web more meaningful. It is based on XML and used to describe content in a way that is understood by computers. RDF helps connect isolated databases across the web with consistent semantics. The structure of any expression in RDF is a collection of triples. RDF triples consist of two pieces of information (subject and object) and linking fact (predicate).

3.9 Ontologies

Advances in Internet technologies makes items on the Web to be organised in such a way that meaning can be easily derived from them. Ontologies are ways of organising and describing related items, and are used to represent semantics. It serves as a means of cataloguing Internet content in a way that can be understood by the computers. RDF and OWL (Web Ontology Language) are designed for formatting ontologies.

3.10 Application Programming Interface (APIs)

Application Programming Interface (APIs) provides application with access to external services and databases. For example, a traditional programming API, like the Sun’s Java API, allows programmers to use already-written methods and functions in their programs. In addition, Web services have APIs that permit their functionality and information to be shared or used across the internet. Most major Web 2.0 companies (for example, eBay, Amazon, Google, Yahoo! and Flickr) provide APIs to encourage use of their services and data in the development of mashups, widgets or gadgets.

3.11 Mashups

Mashups is a means of combining contents or functionality from existing Web services, Websites and RSS feeds or other solutions to serve a new purpose. For example, a skilled developer could mashup Google Maps with a tourist site to create more exciting services/sites on the Internet. The use of APIs helps to save lots of time and money in mashups processes of combining two or more applications to create others. Its possible to build great mashups in a day. Please, note that the mashup may rely on one or more third parties software. Thus, if the API provider experiences downtime, the mashup will be unavailable as well because of the dependence. The way out will be to use mashup that are programmed to avoid sites that could be down. It is also recommended that a developer check the “term of service” for using third party software for the purpose of mashup.

3.12 Widgets and Gadgets

Widgets are commonly referred to as gadgets. They are mini applications designed to run either as stand alone or as add-on features in Web pages. Widgets can be used to for the personalization of a user's Internet experience. Some personalised services may include the display of real-time weather conditions, viewing of maps, receiving event reminder, providing easy access to search engines, aggregating RSS feeds, and so on. The robustness of web services, APIs and other related tools make it easy to develop Widgets. Several catalogs of widgets exist online with the most all-inclusive being Widgipedia which provides an extensive widgets and gadgets for a variety of platform.

3.13 Web 2.0

The term "Web 2.0" is associated with Web applications that facilitate participatory information sharing, interoperability, user-centred design, and collaboration on the World Wide Web. A Web 2.0 site allows users interact and collaborate with each other in a social media dialogue as creators (prosumers) of user-generated content in a virtual community, in contrast to websites where users (consumers) are limited to the passive viewing of content that was created for them. Examples of Web 2.0 include social networking sites, blogs, wikis, video sharing sites, hosted services, web applications, mashups and folksonomies. Web 2.0 websites allow users to do more than just retrieve information. By increasing what was already possible in Web 1.0, they provide the user with more user-interface, software and storage facilities, all through their browser. Users can provide the data that is on a Web 2.0 site and exercise some control over that data. These sites may have an "Architecture of participation" that encourages users to add value to the application as they use it. The Web 2.0 offers all users the same freedom to contribute.

3.14 Web 2.0 Tools

The client-side/web browser technologies used in Web 2.0 development are Asynchronous JavaScript and XML (Ajax), Adobe Flash and the Adobe Flex framework, and JavaScript/Ajax Dojo Toolkit, MooTools, jQuery, and so on. Ajax programming uses JavaScript to upload and download new data from the web server without undergoing a full page reload. To allow users to continue to interact with the page, communications such as data requests going to the server are separated from data coming back to the page (asynchronously). Otherwise, the user would have to routinely wait for the data to come back before they can do anything else on that page, just as a user has to wait for a page to complete the reload. This also increases overall performance of the site, as the sending of requests can complete quicker independent of blocking and queuing required sending data back to the client. The data fetched by an Ajax request is typically formatted in XML or JSON (JavaScript Object Notation) format, which constitute the two widely, used structured data formats. Since both of these formats are natively understood by JavaScript, a programmer can easily use them to transmit structured data in their web application. When this data is received via Ajax, the JavaScript program then uses the Document Object Model (DOM) to dynamically update the web page based on the new data, allowing for a rapid and interactive user experience. In short, using these techniques, Web designers can make their pages function like desktop applications. For example, Google Docs uses this technique to create a Web based word processor. Adobe Flex is another technology often used in Web 2.0 applications. Compared to JavaScript libraries like jQuery, Flex makes it easier for programmers to populate large data grids, charts, and other heavy user interactions.[Applications programmed in Flex, are compiled and displayed as Flash within the browser. Flash is capable of doing many things which were not possible pre-HTML5, the language used to construct web pages. Out of the many capabilities, of Flash, the most commonly used in Web 2.0 is its ability to play audio and video files. This has allowed for the creation of Web 2.0 sites where video media is seamlessly integrated with standard HTML. In addition to Flash and Ajax, JavaScript/Ajax frameworks have recently become a very popular means of creating Web 2.0 sites. At their core, these frameworks do not use technology any different from JavaScript, Ajax, and the DOM. What frameworks do is smooth over inconsistencies between web browsers and extends the functionality available to developers. Many of them also come with customisable, prefabricated “widgets” that accomplish such common tasks as picking a date from a calendar, displaying a data chart, or making a tabbed panel. On the server side, Web 2.0 uses many of the same technologies as Web 1.0. New languages such as PHP, Ruby, Perl, Python, JSP and ASP are used by developers to dynamically output data using information from files and databases. What has begun to change in Web 2.0 is the way this data is formatted. In the early days of the Internet, there was little need for different websites to communicate with each other and share data. In the new “participatory web”, however, sharing data between sites has become an essential capability. To share its data with other sites, a website must be able to generate output in machine-readable formats such as XML (Atom, RSS, etc) and JSON. When a site’s data is available in one of these formats, another website can use it to integrate a portion of that site's functionality into itself, linking the two together. This is one of the hallmarks of the philosophy behind the Web 2.0 movement.

3.15 XHTML

eXtensible Hypertext Markup Language (XHTML) is the newer version of HTML, which was covered extensively in Module two of the course material. XHTML combines the formatting strengths of HTML and the data structures and extensibility strengths of XML to deploy applications for device-independent Web access. XHTML uses the tags and attributes of HTML along with the syntax to XML. Using HTML to write application that runs on electronic devices with fewer resources such as a personal digital assistant (PDA) or mobile phone could be an issue. However, this can be accomplished in XHTML since it is more of a descriptive language (unlike HTML) than a structure language.

4.0 CONCLUSION

The Internet is playing a great role in the delivery of contents to users all across the world. Many researches are going on every day to make it more accessible, available, interactive, meaningful and responsive to users' needs. Most of the information in this unit has been presented for you to keep up-to-date with current Internet and Web programming developments.

5.0 SUMMARY

In this unit, we covered current and emerging topics in Web development, which are aimed at making web contents easier to develop and having the responsiveness, look and feel of traditional desktop applications.

6.0 TUTOR-MARKED ASSIGNMENT

Briefly discuss the meaning of the following terms:

i. Web services
ii. Blogs
iii. Cloud Computing
iv. Ontologies
v. Widgets and Gadgets

7.0 REFERENCES/FURTHER READING

Barbara, K. K. & Norman, J. M. (2001). *The World Wide Web: A Mass Communication Perspective*. USA: Mayfield Publishing Company.

Barrie, S. & Valda, H. (2004). *Programming the Web: An Introduction*. USA McGraw-Hill.

Behrouz, A. F. (2003). *Data Communications and Networking*. (3rd ed.). N.Y, USA: McGraw-Hill/Osborne.

Deitel, P. J., & Deitel, H. M. (2008). *Internet and World Wide Web: How to Program*. (4th ed.). New Jersey, USA: Pearson Prentice Hall.

Idowu, S.A., Maitanmi, S.O. & Adetunji, O. O. (2020). Introductory to Web Technology and Development. Nigeria: Jamiro Press

June, C. (2003). *The Unusually Useful Web Book*. USA:New Rider.

Marc, D. M. & Thomas, C. P. (2003) .*Web Design Using Macromedia Dreamweaver.* USA: McGraw-Hill/ Irwin.

Peter, W .(2001). *Instant Web Pages*. USA: Sybex Inc.

Raymond, G. & Ellen, H. (2001). *In-Line/On-Line Fundamentals of the Internet and World Wide Web.* USA: McGraw-Hill Learning Center.

Terry, F-M. (2009). *Web Development and Design Foundations with XHTML*. USA: Pearson.

Dave, D. (2002). *Schaum's Outline of HTML: Introduction to Web Page Design and Development*. USA: McGraw Hill