

# COMPUTER APPLICATION PLUS TWO QUICK NOTES

## Chapter One

### Review of C++ Programming

**Character Set:**-Fundamental unit of C++ language. Classified into letters(a-z,A-Z),digits(0-9),special characters(#, ;, : > { + etc.),white spaces(space bar, tab, new line) and some ASCII characters ranges from 0 to 255.

**Tokens:**-Basic building blocks of C++ programs. Classified into keywords, identifiers, literals, punctuators and operators.

**Keywords:**-Reserved words that convey specific meaning to the language compiler.

**Identifiers:**-User defined words to identify memory locations(variables), statements(labels),functions(function names) data types etc.

**Literals:**-Constants that do not change their value during the program run. Classified into integer constants(digits preceded by +(plus) or -(minus) sign. ), floating point constants(expressed in fractional form and exponential form) , character constants(single character enclosed within single quotes) and string constants(group of characters enclosed within double quotes).

**Operators:**-Symbols that trigger a specific operation. Based on the number of operands, they are classified into unary(that requires only one operand), binary(that requires two operands), and ternary(which requires three operands). Based on the type of operation, they are classified into arithmetic operators (+,-,\*,/,%), relational operators (<,<=,>,>=,==,!=), logical operators(&&(AND),|| (OR),!(NOT)), get from operator(>>), put to operator(<<), assignment operator(=), increment operator(++), decrement operator(- -), arithmetic assignment operators(+,+=,-,=-,\*=,/=,%=) and conditional operator(?:).

**Punctuators:**-Special characters like comma(;), semi colon(;), hash(#), braces({}) etc.

**Data types:**-These are means to identify the type of data and associated operations. Classified into **fundamental data types** ( int, char, float, double, void(represents empty set of data, hence size is zero. )) , **derived data types** (array, function)and **user defined data types**(structure, class, union, enumeration).

**Type Modifiers:**-The keyword signed, unsigned, short and long are type modifiers.

**Expressions:**-Expressions are constituted by operators and required operands to perform an operation. Based on the operators used, they are classified into arithmetic expression, relational expression(uses numeric or character data as operands and returns True or False value as outputs) and logical expression(uses relational expressions as operands and returns True or False value as outputs). Arithmetic expression is divided into integer expression(uses integer data as operands and returns an integer value) and real expression(uses floating point data as operands and returns a floating point value).

**Type Conversion:**-It is the process of converting the current data type of a value into another type. It may be implicitly(type promotion) or explicitly(type casting) converted. In implicit type conversion, the compiler converts a lower type into higher type. In explicit type conversion, user is responsible for the conversion. A **type cast operator()** is used for this purpose.

### **Various statements in a C++ Program**

#### **1) Declaration Statement:-**

Variables should be declared prior to their use in the program and data types are required for this.

Eg: int n,sum;

#### **2) Input Statement:-**

C++ provides the operator >>, called extraction operator or get from operator for input statement.

Eg:- cin>>a>>b>>c;

#### **3) Output Statement:-**

To perform output operation, C++ gives the <<, called insertion operator or put to operator.

Eg: cout<<"Hello";

cout<<a+b+c;

#### **4) Assignment Statement:-**

A specific data is stored in memory locations assignment operator(=).The statement that contains " = " is known as assignment statement.

Eg:- area=3.14\*radius\*radius;

#### **5) Control Statements:-**

##### **a) Selection Statements:-**

##### **1) if(test\_expression)**

**{statements;}**

- 2) `if(test_expression)`  
`{statement1;`  
`}`  
`else`  
`{statement2;`  
`}`
- 3) `if(test_expression1)`  
`{statement1;}`  
`else if(test_expression2)`  
`{statement2;}`  
`else`  
`{statement3;}`
- 4) `switch(expression)`  
`{case value 1:statement1;`  
`break;`  
`case value2:statement2;`  
`break;`  
`default: statement3;`  
`}`
- 5) **conditional operator(?:)**  
 It is a ternary operator of C++ and it requires three operands and It can substitute if- else statement.  
`expression1?expression2:expression3;`

**b) Looping Statements:-**

1. `for(initialization_expression;test_expression;updation_expression)`  
`{statements;`  
`}`
- 2) `initialization_expression;`  
`while(test_expression)`  
`{statements;`  
`updation_expression;`  
`}`
- 3) `initialization_expression;`  
`do`  
`{statement;`  
`updation_expression;`  
`}while(test_expression);`
- 4) **Nesting of Looping Statements:-**  
 Placing a loop inside the body of another loop is called nesting of a loop.  
 Eg:`for(i=1;i<=2;++i)`  
`{for(j=1;j<=3;++j)`  
`{cout<<"\n"<<i<<" "<<j;}}`

**c) Jump Statements:-**

The statements that facilitate the transfer of program control from one place to another place are called jump statements. C++ provides four jump statements that perform unconditional control transfer in a program. They are **return**, **goto**, **break** and **continue** statements.

- 1) **return statement:**  
 The return statement is used to transfer control back to the calling program or to come out of a function.
- 2) **goto statement:**  
 The goto statement can transfer the program control to anywhere in the function. The target destination of a goto statement is marked by a label, which is an identifier.
- 3) **break statement:-**  
 When a break statement is encountered in a loop, it takes the program control outside the immediate enclosing loop.
- 4) **continue statement:-**

The statement continue is another jump statement used for skipping over a part of the code within the loop body and forcing the next iteration.

## Chapter Two

### Arrays

An array is a collection of elements of the same type placed in contiguous memory locations. Each element in an array can be accessed using its position in the list, called **index number** or **subscript**.

#### **Declaring Arrays:**

**data\_type array\_name[size];**

eg:- int num[10];

#### **Memory Allocation for Arrays:-**

The amount of storage required to hold an array is directly related to its type and size. The memory space allocated for an array can be computed using the following formula :

**total\_bytes=sizeof(array\_type)\*size\_of\_array**

#### **Array Initialisation:-**

Array elements can be initialized in their declaration statements.

Eg:-int score[5]={98,87,92,79,85};

#### **Accessing elements of arrays:-**

Accessing each element of an array at least once to perform any operation is known as **traversal** operation.

#### **String handling using arrays:-**

Eg:- char my\_name[10];

```
for(int i=0;i<6;i++)
```

```
cin>>my_name[i];
```

```
for(int i=0;i<6;i++)
```

```
cout<<my_name[i];
```

**NOTE:-**A null character '\0' is stored at the end of each string. This character is used as the string terminator and added at the end automatically. So we can say that memory required to store a string will be equal to the number of characters in the string plus one byte for null character.

#### **Input/Output Operations on strings [The supporting header file is #include<stdio>]**

Input function for string operation: **gets(character\_array\_name);**

Output function for string operation: **puts(string\_data);**

## Chapter Three

### Functions

In programming, the entire problem will be divided into small sub problems that can be solved by writing separate programs. This kind of approach is known as modular programming. The process of breaking large programs into smaller sub programs is called **modularization**.

#### **Merits of modular programming:-**

- 1) Reduces the size of the program.
- 2) Less chance of error occurrence.
- 3) Reduces programming complexity
- 4) Improves reusability

#### **Demerits of modular programming:-**

- 1) Proper breaking down of the problem is a challenging task.
- 2) Each sub problem must be independent of others.

#### **Functions in C++:-**

A function is a named unit of statements in a program to perform a specific task .Functions can be categorized into two:

- 1) **Predefined Functions** or **Built-in Functions**
- 2) **User defined functions**

#### **1. Predefined Functions:-**

##### **1) Console Functions for character I/O:-[The supporting header file is #include<stdio>]**

- a) **getchar():** returns the character that is input through the keyboard.
- b) **putchar():**-displays the character given as the argument on the monitor.
- c) **gets(character\_array\_name):** Input function for string operation.

- d) **puts(string\_data):** Output function for string operation.
- 2) **Stream Functions for I/O Operations:-[The supporting header file is #include<iostream>]**
- a) **cin.get():** It can accept a single character or multiple characters(String) through the keyboard.
- b) **cin.getline() :** It accepts a string through the keyboard.
- c) **cout.put():** It is used to display a character constant or the content of a character variable given as argument.
- d) **cout.write():**This function displays the string contained in the argument.
- 3) **String Functions[The supporting header file is #include<cstring>]**
- a) **strlen():**This function is used to find the length of a string .**Syntax: int strlen(string);**
- b) **strcpy():** This function is used to copy one string into another. **Syntax: strcpy(string1,string2);**
- c) **strcat():**This function is used to append one string into another string. The length of the resultant string is the total length of the two strings.**Syntax:strcat(string1,string2);**
- d) **strcmp():** This function is used to compare two strings. In this comparison, the alphabetical order of characters in the strings are considered . **Syntax:strcmp(string1,string2);**

The function returns any of the following values in three different situations:

- 1) **Returns 0 if string1 and string2 are same.**
- 2) **Returns a –ve value if string1 is alphabetically lower than string2.**
- 3) **Returns a +ve value if string1 is alphabetically higher than string2.**
- e) **strncmpi():**This function is used to compare two strings ignoring the cases. In this comparison, the alphabetical order of characters in the strings are considered . **Syntax:strncmpi(string1,string2);**

The function returns any of the following values in three different situations:

- i. **Returns 0 if string1 and string2 are same.**
- ii. **Returns a –ve value if string1 is alphabetically lower than string2.**
- iii. **Returns a +ve value if string1 is alphabetically higher than string2.**

#### 4) **Mathematical Functions:- [The supporting header file is #include<cmath>]**

- i. **abs():** used to find the absolute value of a number. **syntax: int abs(int);**
- ii. **sqrt():** used to find the square root of a number. **syntax: double sqrt(double);**
- iii. **pow():** used to find the power of a number. **double pow(double,double);**

#### 5) **Character Functions:[The supporting header file is #include<cctype>]**

- i. **isupper():** This function carries out a test “ Is the character an upper case ?” . It is used to check whether a character is in upper case or not. It returns a 1 if the character is in uppercase and 0 otherwise. **syntax: int isupper(char c);**
- ii. **islower():** This function carries out a test “ Is the character a lower case ?” . It is used to check whether a character is in lower case or not. It returns a 1 if the character is in lowercase and 0 otherwise. **syntax: int islower(char c);**
- iii. **isalpha():** This function carries out a test “ Is the character an alphabet ?” . It is used to check whether a character is an alphabet or not. It returns a 1 if the character is an alphabet and 0 otherwise. **syntax: int isalpha(char c);**
- iv. **isdigit():** This function carries out a test “ Is the character a digit?” . It is used to check whether the character is a digit or not. It returns a 1 if the character is a digit and 0 otherwise. **syntax: int isdigit(char c);**
- v. **isalnum():** This function carries out a test “ Is the character an alphabet or a number?” . It is used to check whether a character is alphanumeric or not. It returns a 1 if the character is alphanumeric and 0 otherwise. **syntax: int isalnum(char c);**
- vi. **toupper():** This function is used to convert a lower case character to an upper case. **Syntax: char toupper(char c);**
- vii. **tolower():** This function is used to convert an upper case character to a lower case. **Syntax: char tolower(char c);**

## 2. User defined functions:-

### A. **Function definition:-**

The syntax of a function definition is given below:-

```
data_type function_name(argument_list)
{statements in the body;}
```

The **data\_type** is any valid data type of C++. The **function\_name** is a user defined word. The **argument\_list** is a list of parameters , i.e is a list of variables preceded by data types and separated by commas. The body comprises of C++ statements required to perform the task assigned to the function.

**B. Prototype of functions:-**

A **function prototype** is the declaration of a function by which compiler is provided with the information about the function such as the name of the function, its return type, the number and type of arguments, and its accessibility. The following is the format:

**data\_type function\_name(argument\_list);**

**C. Arguments of functions:-**

Arguments or parameters are the values from the calling function to the called function. The variables used in the function definition as arguments are known as **formal arguments**. The constants, variables or expressions used in the function call are known as **actual(original ) arguments**.

**D. Methods of calling function:-**

Based on the method of passing arguments, the function calling methods can be classified as **Call by Value** method and **Call by reference** method.

**a. Call by Value (Pass by Value ) method:-**

In this method , the values of the actual parameters are copied into the formal parameters . So any changes made inside the formal parameters are not reflected back to the actual parameters.

**b. Call by Reference (Pass by Reference ) method:-**

In this method , the values of the actual parameters are shared into the formal parameters . So any changes made inside the formal parameters are reflected back to the actual parameters. Here a **reference variable** is used as the formal argument. A reference variable is an alias name of another variable. An **ampersand(&)** symbol is placed in between the data type and the variable in the function header.

**E. Scope and life of variables and functions:-**

The concept of availability or accessibility of variables and functions is termed as their scope and life time.

1. **Local variable:-** A variable which is declared inside a function is known as a **local variable**.
2. **Global variable:-** A variable which is declared outside all other functions is known as a **global variable**.
3. **Local function:-** A function which is declared inside the function body of another function is known as a **local function**.
4. **Global function:-** A function which is declared outside the function body of any another function is known as a **global function**.

**Chapter Four**  
**Web Technology**

**Communication the Web**

- a. **Client to Web Server Communication**
- b. **Web Server to Web Server Communication**

**Web Server :-**1. A server computer that hosts web sites. 2. A web server software that is installed in a server computer .

**Web Server Packages**

Apache Server, MicroSoft Internet Information Server (IIS), Google Web Server(GWS),nginx

**Software Ports:-**FTP(20 & 21),SSH(22),SMTP(25),DNS(53),HTTP(80),POP3(110),HTTPS(443)

**NASA:-**National Aeronautics and Space Administration

**ICANN:-**Internet Corporation for Assigned Names and Numbers

SINo	Static Web Page	Dynamic Web Page
1	The content and layout is fixed	The content and layout may change during run time.
2	Never use databases	Database is used
3	Directly run on the browser.	Runs on the server side application programs
4	They are easy to develop.	Development requires programming skills.

**Scripts:-**Scripts are program codes written inside html pages. A script is written inside the **<SCRIPT>** and **</SCRIPT>** tags .

SINo	Client Side Scripting	Server Side Scripting
1	Script is copied to the client browser	It remains in the Web Server
2	Executed in the client browser	Executed in the Web Server
3	Mainly used for validation of data	Used to connect to databases and return data from the web server
4	Users can block client side scripting.	Server scripting cannot be blocked by a user.
5	Features of the webserver affects the coding	Features does not affects the coding.

### Scripting Languages:

A. **JavaScript:** Developed by **Brendan Eich**. **Ajax(Asynchronous JavaScript and Extensible Markup Language )** technology is used in JavaScript.

B. **VBScript:** Developed by **Microsoft Corporation**.

C. **PHP:-**Stands for “**PHP-Hyper text Preprocessor**”, developed by **Rasmus Lerdorf**.

D.**ASP:-**Stands for **Active Server pages**.

E. **JSP:-**Stands for **Java Server Pages**

F.**CSS:-**Stands for **Cascading Style Sheets**.

### Basic Structure of an HTML Document

```
<html><head><title></title></head><body></body></html>
```

**Container Tags:-**requires pair tags – i.e opening tag and closing tag. Eg. <html></html>,<head></head>

**Empty Tags:-**Used opening tags only. Eg. <br>,<hr>,<img>

**Attribute:-**Parameters included within the opening tag. Eg <body bgcolor = ” red ”>

### Important Tags and Attributes

1.<html> **Tag:- a). dir-**specifies the direction of the text to be displayed on the web page. Two values **ltr**(left to right) and **rtl**(right to left).

**b).lang-**specifies the language used within the document. Different lang values are **en-** english, **fr-** french,**de-**german, **it-**italian,**el-**greek,**es-**spanish,**ar-**arabic,**ja-**japanese,**hi-**hindi and **ru-**russian.

2.<head>:-declares the head section.

3.<title>:-mentions the document title.

4.<body> **Tag:** specifies the document body section.

1. **background:-**sets a background image. The important attributes are

2. **bgcolor:-**specifies the background colour.

3.**text:-**specifies the foreground colour.

4. **link:-**colour of the hyperlink that are not visited by the viewer. Default colour is **blue**.

5.**alink:-**specifies the colour of the active hyperlink. Default colour is **green**.

6.**vlink:-**specifies the colour of the hyperlink which is already visited by the viewer. Default colour is **purple**.

7.**leftmargin:-**leftside margin.

8. **topmargin:-**topside margin.

5.**Heading Tags:-**<h1></h1>,<h2></h2>,<h3></h3>,<h4></h4>,<h5></h5>,<h6></h6>. Attribute **align** – values are **left,right** and **center**.

6. <P> **Tag:-**paragraph tag. Attribute **align-** values are – **left,right,center** or **justify**.

7.<br>:-inserting line breaks.

8. <hr>:-produces horizontal line. Attributes **size,width,color,noshade**

9.<center>:- brings the content to the center .

10.<b>:- making the text bold.

11. <i>:-Italicizing the text .

12. <u>:- Underlining the text.

13. <s> and <strike> :- Striking through the text.

14.<big>:- Making the text big sized.

15.<small>:-Making the text small sized.

16.<strong>:-Making the text bold text.

17.<em>:- Emphasizing the text

18.<sub>:-creating subscripts.

19. <sup>:-creating super scripts.

20.<q>:-used for short quotations

21. <blockquote>:-used for long quotations

22. **<pre>**:- Displaying preformatted text.  
 23. **<address>**:- Displaying the address  
 24. **<marquee>**:-Displaying the text in a scrolling marquee.

**Attributes of <marquee>**:-

**height,width,direction**(up,down,left,right),**behaviour**(scroll,slide,alternate),**scrolldelay,scrollamount,loop,bgcolor,hspace**(horizontal space),**vspace**(vertical space).

25.**<div>**:-defining a section. Attributes are **align,id**(identifier),**style**.

26.**<font>**:-specifying the font characteristics. Attributes are **color,face**(type of the font),**size**(values ranges from 1 to 7,default value is 3)

27. **<img> Tag**:- To insert images in HTML pages. Attributes are

**src**(source),**width,height,vspace,hspace,align**(bottom,middle,top),**border**(border line around the image)

**HTML ENTITIES FOR RESERVED CHARACTERS**:-1) **&nbsp;** - non breaking space,2) **&quot;** - Double quotation mark

3) **&apos;** - Single quotation mark 4) **&amp;** - Ampersand 5) **&lt;** - Less than 6) **&gt;** - Greater than 7) **&copy;** - Copy

right Symbol 8) **&trade;** – Trade mark symbol 9) **&reg;** – Registered Symbol

**Comments in HTML**:- HTML comments are placed within **<!-- -->** tag.

## Chapter Five

### Web Designing using HTML

#### Lists in HTML

1).**Ordered list**(**<ol><li></li></ol>**):- Attributes – **start** and **type**(Arabic Numerals,Upper Case Alphabets,Lower Case Alphabets,Roman Numeral Upper,Roman Numeral Lower).

2).**Unordered list**(**<ul><li></li></ul>**):-Attributes - **type**(disc,circle,square)

3).**Definition List**(**<dl><dt></dt><dd></dd></dl>**)[dl-definition list,dt- definition term,dd- definition description]

#### Creating Links

1).**Internal Linking**:-Link within the same document.**<A>**(anchor tag) is used. **Name** attribute is used for the target specification and **href** – (means hyper reference)(# symbol is essential) attribute is used for linking purpose.

2).**External Linking**:-Link from one web page to another web page. **<A>**(anchor tag) and **href** attribute is used.

#### URL:-Uniform Resource Locator

1).**Relative URL** :- **<a href="http://www.scertkerala.gov.in">** - represents the complete path.

2).**Absolute URL**:- **<a href="image.html">** - represents the specified file name only.

**Creating email linking**:-We can create an email hyperlink to a web page using the hyper link protocol **"mailto:"** .

eg:-**<a href="mailto: scertkerala@gmail.com">scert </a>**

**<embed>** tag is used for Inserting Music and Video:-Attributes are **src,height,width,align**.

**<bgsound>** tag is used for inserting audio files only. Attributes are **src,loop**.

#### <table>tag and its attributes:-

**border**(thickness of the border line),**bordercolor,align,bgcolor,background,cellspacing**(amount of space between the cells),**cellpadding**(amount of space between the cell border and content),**width,height,frame**(void-no border,**above**-top border,**below**-bottom border,**hsides**-top and bottom borders,**vsides**-right and left borders,**lhs**-left side border,**rhs**-right side border,**box & border**-border on all sides),**rules**(values - **none,cols,rows,groups,all**)

**NOTE:** When both **bgcolor** and **background** are specified , the **background** will override the **bgcolor**.

**Attributes of <tr> Tags:-**

**align**(values- **left,right,center**),**valign**-vertical alignment( values - **top,middle,bottom,baseline**(aligns the baseline of the text across the cells in the row))

**Attributes of <th> and <td> Tags:-align, valign,bgcolor, rowspan**(number of rows to be spanned by the cell),**colspan**(number of columns to be spanned by the cell)

**<caption> Tag**:-we can provide a heading to a table.

**<frameset> Tag**:- Used for dividing the browser window. **Attributes** are **cols,rows,border,bordercolor**

**<frame> Tag**:- **Attributes** are **src,scrolling**(displays whether a scrollbar or not-values are **yes,no,auto**),**noresize**(prevents the resizing behaviour),**marginwidth,marginheight,name**(gives a name to a frame).

#### Forms in Web pages:-

**<form> Tag**:- Used for creating a form with various form controls. **Attributes** are **action**(specifies the URL),**method**(get method and **post** method),**target**(specifies the target window).

**Form Controls:-**

1) **<input> Tag**. **Attributes** are a) **type** :- Values are **text,password,checkbox,radio,reset,submit,button**

b) **name**:- used to give a name to the control.

- c) **value**:-used to provide an initial value.
- d) **size**:- sets the width of the input texts.
- e) **maxlength**:-specifies the maximum size.
- 2) **<textarea> Tag**:- Used for creating a multiline entry text box. **Attributes** are **name,rows,cols**.
- 3) **<select> Tag**:- Used to create a drop down list box . **Attributes** are **name,size,multiple**.
- 4) **<fieldset> Tag**:- Used for grouping related data in a form.

### Chapter Six

#### Client Side Scripting Using JavaScript

**<script> Tag**:- This tag is used to include scripting code in an HTML page.

```
<script Language="JavaScript">
</script>
```

**Functions in JavaScript**:-

```
function function_name()
{
body of the function;
}
```

**Data types in JavaScript**:-**Number**(All positive and negative numbers),**String**(Characters enclosed within double quotes),**Boolean**(true and false values)

**variables**:-In JavaScript , variables can be declared by using the keyword **var**.

```
var x,y;
x=25;
y="Kerala";
```

**Operators in JavaScript**:-

- 1) **Arithmetic Operators**(+,-,\*,/,%,++,--)
- 2) **Assignment Operators**(= ,+ =,- =,\*=,/=,%=)
- 3) **Relational Operators**(<,<=,>,>=,==,!=)
- 4) **Logical Operators**&&(AND),|| (OR),NOT(!)
- 5) **String Addition Operator**(+)

**Control Structures in JavaScript**:-

- 6) **if**(test\_expression)

```
{statements;
}
```
- 7) **if**(test\_expression)

```
{statement1;
}
else
{statement2;
}
```
- 8) **switch**(expression)

```
{case value 1:statement1;
break;
case value2:statement2;
break;
default: statement3;
}
```
- 9) **for**(initialization;test\_expression;updation)

```
{statements;
}
```
- 10) **while**(test\_expression)

```
{statements;
}
```

**Built in Functions** :-

- 1) **alert()**:-Used to display a message on the screen.
- 2) **isNaN()**:- Used to check whether a value is number or not.



- 3) **toUpperCase():**-Converts a lowercase character to an upper case.
- 4) **toLowerCase():**-Converts an uppercase character to a lower case.
- 5) **charAt():**- returns the character at a particular position. The counting starts at zero.
- 6) **length property:**-This property returns the length of the string.

**For accessing values in a Textbox, JavaScript uses the following format:-**  
**document.form\_name.textbox\_name.value**

The various **JavaScript Events** are:

- 1) **onClick:**- It occurs when the user clicks over an object.
- 2) **onMouseEnter:**-It occurs when the mouse pointer is moved onto an object.
- 3) **onMouseLeave:**-It occurs when the mouse pointer is moved out of an object.
- 4) **onKeyDown:**-It occurs when the user is pressing a key on the keyboard.
- 5) **onKeyUp:**-It occurs when the user releases a pressed key.

**External JavaScript file:-** We can place the scripts into an external file and then link to that file from within the HTML document. This file is saved with the extension **‘.js’**. The file can be linked to HTML file using the **<script>** tag. The **type** attribute specifies that the linked file is a JavaScript file and the **src** attribute specifies the location and file name of external JavaScript file.

## Chapter Seven

### Web Hosting

**Web Hosting:-** The service of providing storage space in a web server.

**1)Shared Hosting:-**Multiple websites are shared on a single web server. Cheaper and easy to use. But heavy traffic slows the webserver.

**2)Dedicated Hosting:-**Uses a single , powerful web server for hosting. Advantage are very speed and performance is stable. But it is highly expensive.

If the client is allowed to place their own purchased web server in the service providers facility, then it is called **co-location** .

**3) Virtual Private Server:-**It is a physical server that is virtually partitioned into several servers using the virtualization technology. Some popular server virtualization software’s are **VMware,Virtualbox,FreeVPS,User mode Linux, Microsoft Hyper-V.**

**FTP Client Software’s :-**It transfers files from one computer to another on the internet. The popular FTP Client Software’s are **FileZilla,CuteFTP,SmartFTP.**

**Free Hosting:-**provides web hosting services free of charge.

**Content Management System:-** Refers to a web based software system which is capable of creating, administering and publishing web sites.

**Responsive Web design:-**It is the custom of building a website suitable to work on every device and every screen size. The term **Responsive Web design** was coined by **Ethan Marcotte.**

**ICANN:-**Internet Corporation for Assigned Names and Numbers

**WHOIS:-** It is a domain name availability checker platform.

## Chapter Eight

### Database Management System(DBMS)

**Database:** An organized collection of inter related data items stored together with minimum redundancy.

**DBMS(Database Management System):-**A system which has some set of rules and relationships which allows for the definition, creation, retrieval, updation, maintenance and protection of the databases.

**Advantages:-**

- 1) **Database reduces data redundancy.**
- 2) **Database can control data inconsistency.**
- 3) **Efficient data access.**
- 4) **Data integrity can be maintained.**
- 5) **Data security can be ensured.**
- 6) **Database facilitates the sharing of data.**
- 7) **Enforces the necessary standards.**
- 8) **Offers a facility to do the backup and recovery.**

**Components of DBMS environment:-**

- 1) **Hardware:-** The actual computer system used for the storage and retrieval of the database.
- 2) **Software:-** consists of the actual DBMS, Application Programs and Utilities.

- 3) **Data**:-The operational data and the meta- data(data about data).
  - a) **Fields**:- The smallest unit of stored data.
  - b) **Record**:-A collection of related fields.
  - c) **File**:-A collection of all occurrences of the same type of data.
- 4) **Users**:-
  - a) **Database Administrator(DBA)**:-Person who is responsible for the control of the centralized and shared database.
  - b) **Application Programmer**:-Computer professionals who interact with the DBMS through application programs.
  - c) **Sophisticated users**:- person who interact with the DBMS through their own queries.
  - d) **Naive Users**:-who interact with the DBMS through application programs that were written previously.
- 5) **Procedures**:-refers to the instructions and rules that govern the design and use of the database.

**Data Abstraction**:- The developers hide the complexity of the database from users through several levels of abstraction. They are

- 1) **Physical Level**:- The lowest level of database abstraction. This level describes that how the data are actually stored on the storage devices.
- 2) **Logical Level**:-The next higher level of database abstraction. This level describes that what data are actually stored in the database.
- 3) **View Level**:-The highest level of database abstraction. This level is concerned with the way in which data are viewed by individual users.

**Schema**: Overall design of the database.

**Instances**:-Collection of information stored in the database at a particular moment.

**Data Independence**:- The ability to modify the schema followed at one level without affecting the schema followed at the next higher level.

- a) **Physical Data Independence**:- The ability to modify the schema followed at physical level without affecting the schema followed logical level.
- b) **Logical Data Independence**:- The ability to modify the schema followed at logical level without affecting the schema followed view level.

**RDBMS:-Relational Database Management System.- invented by Edgar Frank Codd.**

**Terminologies in RDBMS:-**

- a) **Entity**:-a person or a thing in the real world that is distinguishable from others.
- b) **Relation(Table)**:- collection of data elements organized in terms of rows and columns.
- c) **Tuple**:- The row(records) of a relation.
- d) **Attribute**:-The columns of a relation.
- e) **Degree**:- The number of attributes in a relation.
- f) **Cardinality**:- The number of rows or tuples in a relation.
- g) **Domain**:- a pool of values from which actual values appearing in a given column are drawn.
- h) **Keys**:-An attribute or a collection of attributes in a relation that uniquely distinguishes each tuple from other tuples in a given relation.
  - 1) **Candidate key**:- The minimal set of attributes that uniquely identifies a row in a relation.
  - 2) **Primary key**:- one of the candidate keys chosen to be the unique identifier for a table by the database designer.
  - 3) **Alternate keys**:- A candidate key that is not the primary key .
  - 4) **Foreign keys**:- A candidate key which is the primary key of another table.

**Relational Algebra:-**

- 1) **SELECT( $\sigma$ )** Operation: Unary operator . Selects rows from a relation that satisfies a given predicate(condition).
- 2) **PROJECT( $\pi$ )**:- Unary operator. Selects certain attributes from the table and forms a new relation.
- 3) **UNION( $\cup$ )**:- Binary operator. Combines two relations.
- 4) **INTERSECTION( $\cap$ )** : Binary operator. It returns the common elements only.
- 5) **SET DIFFERENCE( $-$ )**:- Binary operator. It returns a relation containing the tuples appearing in the first relation but not in the second.
- 6) **CARTESIAN PRODUCT( $\times$ )**:- returns a relation consisting of all possible combinations of tuples from two relations.

## Chapter Nine

### Structured Query Language(SQL)

The original version of SQL was developed in the 1970's by Donald D Chamberlin and Raymond F Boyce at IBM's San Jose Research Laboratory(now Almanden Research Centre). It was originally called Sequel(Structured English Query Language) and later its name was changed to SQL.

Components of SQL:-

- a) Data Definition Language(DDL):-CREATE TABLE,ALTER TABLE,DROP TABLE.
- b) Data Manipulation Language(DML):-SELECT,INSERT INTO,UPDATE,DELETE FROM.
- c) Data Control Language(DCL):-GRANT,REVOKE.

Creating database:- CREATE DATABASE <database\_name>;

Opening database:-USE <database\_name>;

DATA TYPES IN SQL:-

- 1) TINY INT ,SMALL INT,MEDIUM INT,INT,BIG INT
- 2) FLOAT(M,D),DOUBLE(M,D),DECIMAL(M,D)  
[M means maximum size and D means the number of digits to the right of the decimal point]
- 3) CHAR(SIZE),VARCHAR(SIZE):- includes letters, digits and special symbols. CHAR is a fixed length character data type. It consumes the declared size. VARCHAR represents variable length strings. It consumes only the actual size of the string, not the declared size.
- 4) DATE:-Used to store dates. The YYYY-MM-DD is the standard format. The supported range is from 1000-01-01 to 9999-12-31.
- 5) TIME:-displays the current time in the format HH:MM:SS.

### SQL COMMANDS

#### a. DATA DEFINITION LANGUAGE (DDL COMMANDS)

##### 1. CREATE TABLE COMMAND:-

CREATE TABLE <table\_name>(<column1> < data\_type1> (size)[<constraint>],.....);

Constraints:- rules enforced on data that are entered into the column of a table.

- a) Column constraints:-are applied only to individual columns.
  - 1) NOT NULL:- Specifies that a column can never have null values.
  - 2) AUTO\_INCREMENT:-Performs an automatic increment feature.
  - 3) UNIQUE:-It ensures that no two rows have the same value .
  - 4) PRIMARY KEY:-This constraint declares a column as the primary key of the table.
  - 5) DEFAULT:-a default value can be set for a column.
- b) Table Constraints:-It can be used not only on individual columns, but also on a group of columns. Eg:UNIQUE.
  2. ALTER TABLE ADD COMMAND:-  
ALTER TABLE <table\_name> ADD <column\_name> <data\_type>[<size>] [<constraint>] [FIRST] / AFTER <column\_name>;
  3. ALTER TABLE MODIFY COMMAND:-  
ALTER TABLE <table\_name> MODIFY <column\_name> <data\_type> [<size>] [<constraint>];
  4. ALTER TABLE DROP COMMAND:-  
ALTER TABLE <table\_name> DROP <column\_name>;
  5. ALTER TABLE RENAME COMMAND:-  
ALTER TABLE <table\_name> RENAME TO <new\_table\_name>;
  6. DROP TABLE COMMAND  
DROP TABLE <table\_name>;
- b. DATA MANIPULATION LANGUAGE COMMANDS(DML):-
  1. INSERT INTO COMMAND:-  
INSERT INTO <table\_name> (<column1>,<column2>,....) VALUES(<value1><value2>,....);
  2. SELECT FROM COMMAND:-  
SELECT <column1>,<column2>,.... FROM <table\_name> WHERE <condition>;  
The DISTINCT keyword is used for avoiding the duplicate values.

MySQL provides a variety of operators or clauses attached with SELECT command.

SLNO	OPERATOR	MEANING/RESULT
1	=	Equal to relational operator
2	< > or !=	Not Equal to relational operator
3	<	Less than relational operator
4	<=	Less than or equal to relational operator
5	>	Greater than relational operator
6	>=	Greater than or equal to relational operator
7	NOT	True when condition is False
8	AND	True if both the conditions are True
9	OR	True if either of the condition is True
10	BETWEEN...AND	Used to specify a range including lower and upper values
11	IN	It provides a list of values
12	IS	retrieves NULL values with this operator
13	LIKE	It is a pattern matching operator. % (percentage )and _(under score ) are used with this keyword
14	ORDER BY	It is used for sorting ascending (ASC) or descending(DESC) order
15	GROUP BY	used to form groups
16	HAVING	specifies conditions and it is used along with GROUP BY clause

#### Aggregate Functions:-

- 1) **SUM()**:-Calculates the sum of a specified column.
- 2) **AVG()**:-Calculates the average value in a specified column.
- 3) **MAX()**:-Calculates the maximum value.
- 4) **MIN()**:-Finds the minimum value.
- 5) **COUNT()**:-Counts the number of non null values.

#### 3. UPDATE COMMAND

**UPDATE <table\_name> SET <column\_name>=<value> WHERE <condition>;**

#### 4. DELETE FROM COMMAND

**DELETE FROM <table\_name> WHERE <condition>;**

**Nested Query:-** One query contains another query is called nested query. Then the inner most query is called **sub query** and the outer most query is called **outer query**.

**VIEWS:-** A view is a virtual table that does not really exist in the database, but is derived from one or more tables.

**CREATE VIEW <viewname> AS SELECT <column1>,<column2>,... FROM <table\_name> WHERE <condition>;**

A view can be removed by

**DROP VIEW <viewname>;**

## Chapter Ten

### Enterprise Resource Planning(ERP)

ERP combines all the business requirements of an enterprise or company together into a single , integrated software that runs off a single database so that the various departments of an enterprise can share information and communicate each other more easily.

#### Functional units of ERP:-

- A. Financial Module:-** It can collect financial data from various functional departments and generate valuable financial reports.
- B. Manufacturing module:-** It contains necessary business rules to manage the entire production process.
- C. Production Planning module :-** It is used for optimizing the utilization of available resources and helps the organization to plan their production.
- D. HR Module:-** Human Resource module of ERP focuses on the management of human resources and human capital.
- E. Inventory Control Module:-** This module covers processes of maintaining the appropriate level of stock in the ware house.
- F. Purchasing Module:-** It is used for making the required raw materials available in the right time and at the right price.

**G. Marketing Module:-** It is used for monitoring and tracking customer orders, increasing customer satisfaction, and for eliminating credit risks.

**H. Sales and distribution module:-** It includes inquiries , order placement, order scheduling , dispatching and invoicing.

**I. Quality Management Module:-**This module is used for managing the quality of the product.

**Business Process Re- engineering(BPR):-** It is the analysis and redesign of work flow within an enterprise. A business process consists of three elements.

- **Inputs**
- **Processing**
- **Outputs**

**Implementation of ERP:-**

1. **Pre Evaluation screening:-** This step limits the number of packages to be evaluated.
2. **Package Selection:-** selects an appropriate package.
3. **Project Planning:-** This phase decides when to begin the project, how to do it and when to complete it.
4. **Gap Analysis:-**Gap analysis means the steps to find the remaining needs of the enterprise.
5. **Business Process Re- engineering:-** rethinking and redesign of the business process to achieve improvements in process..
6. **Installation and Configuration:-**installs a new ERP package.
7. **Implementation team training:-** The company trains their employees to implement and run the new ERP package.
8. **Testing:-** The software is tested to ensure that it performs properly.
9. **Going live:-** in this step, the system becomes live to perform the enterprise operations.
10. **End user training:-**this step provides the training for end users.
11. **Post implementation:-**In this phase, errors may be corrected and necessary steps may be taken to improve the processing efficiency.

**ERP Solution Providers/Packages:-**

- Oracle
- SAP(Systems Applications and Products)
- Odoo
- Microsoft Dynamics
- Tally ERP

**Benefits and risks of ERP:-**

**Benefits:-**

1. Improved resource utilization
2. Better Customer Satisfaction
3. Provides accurate information
4. Decision making capability
5. Increased flexibility
6. Information Integrity

**Risks of ERP implementation:-**

1. High cost
2. Time consuming
3. Requirement of additional trained staff
4. Operational and maintenance issues

**ERP and Related Technologies:-**

- **Product Life Cycle Management(PLM):-** It is the process of managing the entire life cycle of a product.
- **Customer Relationship Management(CRM):-**It is a comprehensive approach for creating, maintaining and expanding customer relationships.
- **Management Information Systems(MIS):-** It is an integrated system of man and machine for providing the information to support the operations, the management and the decision making function in an organization.
- **Supply Chain Management(SCM):-**It consists of all the activities associated with moving goods from the supplier to the customer.

- **Decision Support System(DCS):-** Decision Support Systems are interactive, computer based systems that aid users in judgment and choice activities.

## Chapter Eleven

### Trends and Issues in ICT

ICT means Information and Communication Technology.

**Mobile Computing:-**

**Generations in Mobile Communication:-**

1. **First Generation networks(1G):-** developed around 1980. 1G mobile phones were based on the analog system and provided basic voice facility only.
2. **Second Generation networks(2G):-** It follows digital system for communication. The popular standards introduced by 2G systems are **Global System for Mobiles(GSM)** ,**Code Division Multiple Access(CDMA)**,**General Packet Radio Services(GPRS)** and **Enhanced Data rates for GSM Evolution(EDGE)**.
3. **Third Generation networks(3G):** It is also referred to as wireless broadband and it uses the **WCDMA(Wideband Code Division Multiple Access)** technology.
4. **Fourth Generation networks(4G):-** It is also called **LTE(Long Term Evolution)** and it uses the **Orthogonal Frequency Division Multiple Access(OFDMA)** technology.
5. **Fifth Generation networks(5G):-** It will be the future generation of mobile networks.

**Mobile Communication Services:-**

- a. **Short Message Services(SMS):-** SMS messages are exchanged using the protocol called **Signaling System No. 7(SS7)**.
- b. **Multimedia Message Services(MMS):-**MMS supports contents such as text, graphics, music, video clips and more.
- c. **Global Positioning System(GSM):-** It is a satellite based navigation system that is used to locate a geographical position anywhere on earth , using its longitude and latitude.
- d. **Smart cards:-**It is a plastic card embedded a computer chip that stores and transacts data.

**Android Operating Systems:-** Android operating systems was developed by **Andy Rubin** in 2003.The various versions are **Cupcake(1.5)**, **Donut(1.6)**, **Éclair(2.0)**, **Froyo(2.2)**, **Gingerbread(2.3)**, **Honeycomb(3.0)**, **Ice cream Sandwich(4.0)**, **Jelly Bean(4.1 to 4.3.1)**, **KitKat(4.4 to 4.4.4)**, **Lollipop(5.0 to 5.1.1)** ,**Marshmallow(6.0 to 6.0.1)**, **Nougat(7.0 to 7.1)**, **Oreo(8.0 to 8.1)** and **Pie(9.0)**.

**ICT in Business:-**

1. **Social networks and Big data analytics:-**Big data analytics is the process of examining large data sets containing a variety of data types to uncover hidden patterns, market trends , customer preferences and other useful business information.
2. **Business Logistics:-**It is the management of flow of goods/resources in a business between the point of origin and the point of consumption in order to meet the requirements of customers or corporations. **Radio Frequency Identification(RFID)** technology is used for business logistics.

**Information Security:-**

1. **Intellectual Property Right(IPR)**
  - a) **Patent:-** exclusive right granted for an invention.
  - b) **Trade Mark:-** distinctive sign that identifies certain goods or services produced or provided by an individual or a company.
  - c) **Industrial Designs:-**refers to the ornamental or aesthetic aspects of an article.
  - d) **CopyRight:-**legal right given to the creators for an original work, for a limited period of time.

**Infringement:-** Unauthorized use of intellectual property rights such as patents, copyrights, and trademarks .

**Cyber space:-** a virtual environment created by computer systems connected to the internet.

**Cyber Crimes:-** It is a criminal activity in which computers or computer networks are used as a tool, target or a place of criminal activity.

**A. Cyber crimes against individual:-**

1. **Identity Theft**
2. **Harassment**
3. **Impersonation and Cheating**
4. **Violation of Privacy**
5. **Dissemination of obscene material**

**B. Cyber Crime against Property**

1. credit card fraud
2. intellectual property theft
3. internet time theft

**C. Cyber Crime against Government**

1. Cyber Terrorism
2. Website Defacement
3. Attack against e-governance websites

**Cyber Laws:-** refers to the legal regulatory aspects of the internet.

**Information Technology Act 2000(Amended in 2008):-** In May 2000, the Indian Parliament passed Information Technology Bill and it is known IT Act 2000.

**Cyber Forensics:-** It is the process of using scientific knowledge for identifying, collecting, preserving, analyzing and presenting evidence to the courts .

**Infomania:-** It is the state of getting exhausted with excess information.