**DJACM WEBX DOCS**

**HTML**

**1.What is HTML?**

HTML or **H**YPER**T**EXT **M**ARKUP **L**ANGUAGE is standard set of markup tags used to create Web Pages.A web browser can read HTML files and compose them into visible pages.

The filename extension for files containing HTML is .html or .htm

**2.Tags in Html**

Tags are diffrent elements of the WebPage.A Webpage can have text,images,links,etc.All of these items have tags to define them on WebPage.Tags begin with **<tag\_name>** and end with **</tag\_name>**.

Some tags have attributes(extra properties) which can be defined.Say for example, you want to change size and color of your text.For this we use <font> tag and specify color and size as attributes

<font **size**=”4” **color**=”red”>Some text of size 4 and color red</font>

Similarly for an image we can have **height** and **width** as attributes

**Some general attributes**

**class**

It is used to classify elements to a particular class and apply all properties of that class to the element which has that class in its attribute.

**id**

The id selector uses the id attribute of an HTML tag to find the specific element.An id should be unique within a page, so you should use the id selector when you want to find a single, unique element.

**3.Basic Tags**

**<html>**

The HTML document itself begins with <html> and ends with </html>.

**<head>**

The <head> element must include a title for the document, and can include scripts, styles, meta information, and more.

**<title>**

Title for the webpage that will appear on the browser tab

**<body>**

The visible part of the HTML document is between <body> and </body>.

**<br>**

Breaks the current line and goes to next line

**<a>**

Anchor tag is used to create links between web pages.The href attribute is used to provide the link

<a href=”[www.link\_to\_other\_page.com](http://www.link_to_other_page.com/)”>Click me</a>

**<h1>** to **<h6>**

Headings of different sizes from size 1(largest size) to size 6 (smallest size)

**<form>**

The <form> tag is used to create an HTML form for user input.It can have various input elemenst in it.

**Attributes:**

**action-**  Specifies where to send the form-data when a form is submitted

**method-**  Specifies the HTTP method to use when sending form-data

**<input>**

Defines any input object with which the user can interact and provide data to Webpage.Examples are buttons,radioboxes,checkboxes,text input,etc

E.g.

TextBox

**<input type="text">**

Password field(All characters will appear as asterisks(\*))

**<input type="password">**

Radio Buttons( Choose one from many)

**<input type="radio">**

Button

**<input type="submit>**

**Attributes:**

**name**

The name attribute is used to reference elements in a JavaScript, or to reference form data after a form is submitted.

**value**

The value attribute is used differently for different input types:

* For "button", "reset", and "submit" - it defines the text on the button
* For "text", "password", and "hidden" - it defines the initial (default) value of the input field
* For "checkbox", "radio", "image" - it defines the value associated with the input (this is also the value that is sent on submit)

**type**

The type attribute specifies the type of <input> element to display.The default type is: text.

**FORM EXAMPLE:**

<form action="demo\_form.php" method="post">

First name: <input type="text" name="fname"><br>

Last name: <input type="text" name="lname"><br>

<input type="submit" value="Submit">

</form>

**<div>**

Used to group different html tags together.All properties applied to a div are applied to all the members of that div group.

**<script>**

Html is NOT a programming language.It cannot even compute sum of two numbers.So we use Vbscript,Javascript,etc to achieve such tasks.Script tag tells the browser to include this script in the webpage.

**<style>**

Html Web page elements don't look very appealing .But they can be styled as per our convinience.Stlye tag is used to write style blocks into html pages which can be applied to differernt elements.

**SQL**

SQL stands for structured query language.A webpage cannot store data for future use.

This is where Databases come into play.They can store any type of data for future use.

A Database stores data in tables just like a Microsoft Excel Book has Excel Sheets to store data.To manipulate and access this data we need sql queries.

**SQL QUERIES**

**SELECT**

*SELECT column\_name(s) FROM table\_name*

Selects the whole column(s) from the specified table

*SELECT \* FROM table\_name*

Selects all the columns from the specified table

**INSERT**

*INSERT INTO table\_name (column1,column2,...) VALUES (value1,value2,..)*

inserts the values-value1,value2,... into column1,column2,... respectively in the specified table.

**WHERE**

*SELECT column\_name(s) FROM table\_name WHERE column\_name = some\_value*

selects only those rows from specified column(s) where the value specified by some\_value is equal to value of specified column\_name of that row

**UPDATE**

*UPDATE table\_name SET column1=value1,column2=value2,...*

update the table column values with the given values

**PHP**

PHP is a server side programming language.Variables in PHP are accessed by putting a $ before the variable name.PHP variables have no data type.

e.g.

$a=3;

A php block looks like:

**<?php**

**some\_php\_code\_here**

**?>**

Everything outside php blocks is ignored by php intepretor.

PHP blocks can appear many times in a html file just like any other html tags.

**echo**

echo is used to print to console or to a file

e.g. echo “Hello”;

**extract(array)**

Extract creates a symbol variables from the array passed to it.

It uses array keys as variable names and array values as variable values.

e.g. extract($\_POST)

where $\_POST is an array of form data submitted to the webpage

|  |  |
| --- | --- |
| Array keys | Array values |
| username | “abcd” |
| password | “efgh” |

The extract() command will create $username=”abcd” and $password=”efgh”

**define()**

The define() function defines a constant.

**define(name,value,case\_insensitive)**

**real\_escape\_string(string)**

This function is used to create a legal SQL string that you can use in an SQL statement. The given string is encoded to an escaped SQL string, taking into account the current character set of the connection.As the SQL string generated will be legal,malicious sql injections are avoided.

**session\_start()**

Starts a new session or resumes a ongoing session for a user.When users login, each user should be assigned a session.

We can store certain values for a session(example- login time) using $\_SESSION array.This are stored as cookies in web browser which can used further.

**session\_destroy()**

Destroys an ongoing session.

**Connecting to a Mysql database, executing sql queries and using the results**

**--Connection**

**$connection\_var = new mysqli(server\_address,username,password,database\_name);**

The above statement connects to a mysql database whose name is speciefied by **database\_name** which is hosted on server\_address.The username and password are passed into **username** and **password** parameters.**$connection\_var** acts a pointer to the connection and all sql operations are done with it.

**--Queries**

**$result=$connection\_var->query(string)**

query() function carries out the sql query specified in string parameter and stores the pointer to sql query result in $result.

string parameter should have a valid SQL query.

returns false if query fails to execute.

**$result->data\_seek(pos)**

data\_seek() makes the **$result** pointer obtained after a sql query to point to the row number, specified in pos parameter,in the results of sql query.

**$row = $result->fetch\_assoc()**

Stores the data of the entire row pointed by **$result** into an array **$row**.

**Example:**

$q = "SELECT \* from items WHERE userid = $id”; //sql query

$rs = $connection\_var**-**>query($q); //point to results of sql query

$rs->data\_seek(5); //point to 6th row in results

$row = $rs->fetch\_assoc() //store 6th row's all column data to row

If ROW 6 in sql query results have username=”abc” and password=”def”

then $row['username']=”abc” and $row['password']=”def”

**CSS**

CSS stands for cascading style sheets.Html elements dont look very appealing but with style sheets we can change properties of these elements like color,alignment,font-size,background-image,etc

CSS can be included in html document by writing it between <style></style> tag.

CSS written in an external file can be included in html document as

**<link rel="stylesheet" href="style.css">**

CSS selectors allow you to select and manipulate HTML element(s).

CSS selectors are used to "find" (or select) HTML elements based on their id, classes, types, attributes, values of attributes and much more.

Consider the HTML:

<html>

<body>

<h1 class='paraclass1' >Heading 1</h1>

<p class='paraclass1'>Paragraph 1

<p id='para1'>Paragraph 2

<p> Paragraph 3

<body>

</html>

|  |  |  |  |
| --- | --- | --- | --- |
| By tag name | By id | By class | By class of a tag |
| p {  text-align:center;  color: red;  } | #para1 {  text-align: center;  color: red;  } | .paraclass1 {  text-align: center;  color: red;  } | p.paraclass1{  text-align: center;  color: red;  } |
| Heading1  Paragraph 1  Paragraph 2  Paragraph 3 | Heading1  Paragraph 1  Paragraph 2  Paragraph 3 | Heading1  Paragraph 1  Paragraph 2  Paragraph 3 | Heading1  Paragraph 1  Paragraph 2  Paragraph 3 |

By tag code will apply to all <p> tags in our html document

By id code will apply to tags with id='para1'

By class code will apply to tags with class='paraclass1'

By tag.class will apply to all <p> with class='paraclass1'

We can even group different tags together to apply the same style

e.g.

p,h1,h2 {

text-align:center;

color: red;

}

For our WebPages we will be using bootstrap.css which has all css pre-defined under different class name.To apply the required css to a html element we just write the required class name in “class” attribute of html tag.

We use a grid system to arrange HTML elements on our page.

Suppose we have to place two text boxes size by side and three buttons side by side on next line,

we will need two rows -one with two columns, one for each textbox

and another with three columns, one for each button

Rows are placed into containers

So our design structure would look like

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| CONTAINER   |  |  |  | | --- | --- | --- | | ROW   |  |  | | --- | --- | | COLUMN1 TEXTBOX | COLUMN2 TEXTBOX | | | ROW   |  |  |  | | --- | --- | --- | | COLUMN1 BUTTON | COLUMN2 BUTTON | COLUMN3 BUTTON | | |