UI TECHNOLOGY TRAINING.

For any application , there is a front end and back end components.

While the back end mainly comprises of data base management and server management , the front end which is the user accessible component is mostly where we can see the functionality and effectiveness of the application.

UI technology is a front end technology which is used to make the application user friendly and provide knowledge about the application in an understandable way.

Every UI admin/developer must be aware of the following technologies.

1.GIT

2.HTML – Hyper Text Markup language- it is used to lay the foundation of any web application

3.CSS – Cascading style sheets- it is mainly used for the presentation such as including styles to the web page .

4.JAVASCRIPT- this is used to provide the functionality to the web page.

1.GIT - It is a revision control system, which is used to manage the source code history and **Git-hub** is the site which plays as a host for Git . Git-hub is comprised of directories known as **Repositories,** that are used to store the code and project files.

2. HTML

It is made up of various elements and there are mainly two sections in a html code , a head and a body.

In the head section, we usually write the text which we want to display on the web page.

In the body section, we include functionality such as links and navigation to other page and etc..

The html elements are represented by tags.

There are many tags such as heading,table, list ,paragraph, etc..

The syntax is

<tagname > content </tagname>

HTML has 2 types of elements.

1.Inline Elements.

2. Block level Elements.

1. Inline elements are used to display the text or data in a single line without starting a next line .

Some of the important inline elements are

<a>- it is used to display any links

<span> it is used to display text in one line without moving to the next line.

2. Block- level elements : these display content in a new line.

The most important block-level element is <div>.

<div>- it is used to define a section as a block level in the document.

3.CSS

There are different tags in CSS, but the most relevant one for this instance is BOX model.

It has four different layers starting from inner to outer layer ,

1.content- the text inside the element

2. Padding- area around the text

3.Border- border around the padding and content

4.Margin- a transparent line clearing area outside the border.

To display a box model for content on a web page and make it flexible according to the usage of the window , we use a property called flexbox.

* Using Flexbox, we can display content tabs in a horizontal/vertical direction.
* Wrap the flex container’s according to the window’s real estate.
* Display the containers with space and padding

To make changes in a specific container, we use class and id .

* Class is used for like containers
* Id is used to display content in a different way for a specific container.

Media queries- to adjust the application’s interface window in contrast with the device dimensions.

Position property

* This property is used to align the elements in a proper position.
* Changes that occur to the elements are linked with the position of the element.

There are mainly three properties that are widely used

* Fixed- the position is fixed relative to the web page
* Relative – the element is in a relative position to its normal position and any changes made are effective to its relative position.
* Absolute – The element is placed in a position relative to its parent element.

4.JAVASCRIPT.

Among most of the methods to assign a functionality using javascript, getelementbyid() is the widely used one.

In this method, when invoked , the id which is displayed initially is changed by calling the inner html element stated in that method.

* Using java script, we can change the content to display using property like button.
* Hide the html elements.
* Show the html elements
* Change the style of the content.

Scope: the scope of a variable/function is defined as the access given to it to be called whenever and wherever necessary. There are three kinds of scopes.

* Block-level scope – anything that is declared within the braces {}, is only accessible within that {}.
* Functional Scope – the scope of any variable is only limited within that function.
* Global Scope – the scope of all the global variables i.e. window. variables that are declared within a javascript.

\*\*Currently, Block level scope is outdated

Hoisting

The process of debugging a javascript code by a browser compilation engine .It has two phases.

1.Compilation phase :

In this phase, the browser scans each line of code for any keywords or function declarations.

At the same time, it also checks for the scope within which the variables are declared.

2. Execution Phase :

In this phase , each line of the code is checked from top to bottom for any values assigned to the variables and also any functions are to be called.

Example :

Var a = 1;

Var b = 1;

Function.add (a,b){

return.a\*b;

}

add();

In the above example,

1. It finds out that var keywords are declared and they are in global scope
2. It jumps to the function and finds that there are no keywords declared and are in the right scope.
3. As there are no more declarations, the execution begins by assigning the values 1 to the variables a & b.
4. As there are no more values assigned ,it jumps to the add() that has to be called and displays the output.

In case, if a keyword is declared within a function and not in global scope, at the time of execution , it creates a variable by default in the global scope . This type of scenario is a bad practice in writing a javascript. To avoid this , we declare a context “use strict” which gives a reference error in case of a bad declaration or a wrong function definition.

var a = b();

var c = d():

a;

c;

function b(){

return c ;

}

var d = function(){

return b();

}

Ans :

a = undefined

c = ?

d = undefined.

foo();

var foo = 2;

function foo(){

console.log("bar");

}

function foo() {

console.log("foo");

}

Ans :

When foo() is called first time , it displays bar in the console window

Again when foo() is called , it displays a reference error.