JAVASCRIPT

SC 335 - Introduction to Web development
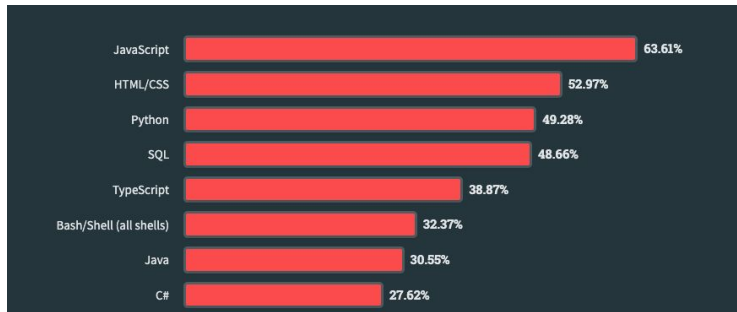
What is JAVASCRIPT ?

INTRO. | What is javascript?

Language	Percentage
JavaScript	63.61%
HTML/CSS	52.97%
Python	49.28%
SQL	48.66%
TypeScript	38.87%
Bash/Shell (all shells)	32.37%
Java	30.55%
C#	27.62%

Credits : Popular technologies survey 2023 (stack overflow)

$72,000/year

INTRO. | JAVASCRIPT Developer

Front-end developer

Back-end developer

JAVASCRIPT - BASICS

BASICS | Variables

1. console.log()

   Object that writes a message to the browser's console,

2. alert

   A method of the window object that displays a message box with a specified message and an OK button.

BASICS | Variables

1. var

   Declarations are globally scoped or function scoped.
   It can be updated or re-declared.

2. let

   Declarations are block scoped.
   Can be updated but not redeclared

3. const

   It cannot be updated or redeclared

BASICS | Data Types

1. ## Undefined

   A variable that has not been assigned a value.

2. ## Null

   A variable that has been explicitly assigned the value null.

3. ## String

   A sequence of characters enclosed in quotes.

4. ## Number

   A numeric value.

5. ## Boolean

   A logical value that can be either true or false.

BASICS | Arrays and Objects

Arrays

Syntax   `const array_name = [item1, item2, ...];`

1. Add/Remove

push - Add element to the end
pop - Remove element at the end
shift -Add element to the start
unshift - Remove element at the start

2. Sort,Reverse

A numeric value.

Objects

Objects are variables too. But objects can contain many values.

Syntax   `const object_name = {label1: value, label2, value, label3: function()};`

BASICS | String Methods

1. ## Case

   toUppercase - *Change the case to upper*

   toLowercase - *Change the case to lower*

2. ## Substring

   slice(*start,end****)*** -

   split(*delimiter****)*** -

3. ## Find and Replace

   find(*char****)*** -

   replace(*searchValue,replaceValue****)*** -

   replaceAll(*searchValue,replaceValue****)*** -

BASICS | Loops

1. for

Regular *for loop.*

```
for (expression 1; expression 2;
expression 3) {
  // code block to be executed
}
```

for of: *It loops through the values of an iterable object..*

```
for (variable of iterable) {
  // code block to be executed
}
```

2. while

The while loop loops through a block of code as long as a specified condition is true.

```
while (condition) {
  // code block to be executed
}
```

BASICS | Conditional Statements

1. if/else

if…else statement is used to execute a statement if a specified condition is truthy. If the condition is falsy, another statement in the optional else clause will be executed

```
if (condition) {
  //  block of code to be
executed if the condition is
true
} else {
  //  block of code to be
executed if the condition is
false
}
```

2. switch/case

Perform different actions based on different conditions

```
switch(expression) {
  case x:
    // code block
    break;
  case y:
    // code block
    break;
  default:
    // code block
}
```

BASICS | Functions

The function is a block of code designed to perform a particular task.

Syntax

```
function name(parameter1, parameter2, parameter3) {
  // code to be executed
}
```
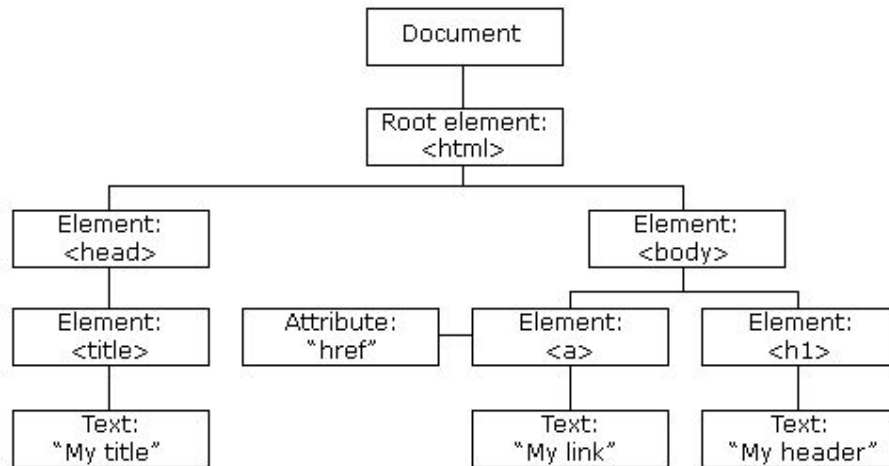
HTML Document Object Model (DOM)

DOM | HTML DOM Tree

Image credits : W3schools.com

*Use if to Manipulate **content,Style** and **Structure***

DOM | Finding Elements in DOM

Methods used access HTML elements in an HTML DOM.

getElementById() *Access by ID*

getElementsByTagName() *Access by Tag name*

getElementsByClassName() *Access by Class name*

querySelector() *Access by CSS selectors*

DOM | Changing HTML Elements

The HTML DOM allows JavaScript to change the content of HTML elements using following methods.

innerHTML
The easiest way to modify the content of an HTML element

attribute
change the value of an HTML attribute

style.property
change the style of an HTML element

setAttribute(attribute, value)

DOM | Events

Basics

HTML DOM

Finding Elements

Changing Elements

Events

Event Listeners

API

HTML DOM allows JavaScript to react to HTML events:

onmouseover

onmouseout

onmousedown

onload

DOM | Events Listeners

Method allows you to add event listeners on any HTML DOM object such as HTML elements

Syntax : `element.addEventListener(event, function, useCapture);`

events

It is a way to attach a function to an element or an object that will be executed when a certain event occurs.

DOM | Browser API

It is a way to attach a function to an element or an object that will be executed when a certain event occurs.

localstorage

setItem*(key,variable)*

getItem*(key)*

removeItem*(key)*

open/close

open*(url)*

close*()*

history

back(n)

go(n)