In JavaScript we have the following conditional statements:

* Use **if** to specify a block of code to be executed, if a specified condition is true
* Use **else** to specify a block of code to be executed, if the same condition is false
* Use **else** **if** to specify a new condition to test, if the first condition is false

In JavaScript we can use following logical operator in conditions:

**AND (&&)**: if all values are true then only returns.

**OR (||)**: if one of the values is true then only returns true.

**NOT (!)**: inverts true/false values based on the condition output.

They have lower precedence than comparison operators.

In JavaScript we can use following Conditional operators:

Ternary Conditional Operator:

* This operator assigns a value to a variable based on some condition.
* Syntax: variablename = (condition)? value1:value2;

Switch statement:

* It executes the code of block when a case matches the value.
* To stop the execution of inside the block we use break statement.

Execution:

1. The switch expression is evaluated once.
2. The value of the expression is compared with the values of each case.
3. If there is a match, the associated block of code is executed.
4. If there is no match, the default code block is executed.

Truthy and falsy values:

Falsy Values are values assigned to variable which return false when used in condition like undefined, null, empty string, 0, NaN.

Truthy values are values which are not falsy values.

Equality Operators:

==: Does type coercion

===: Doesn’t do type coercion

Functions:

If we need to run a code multiple times then we can write that into function to execute it whenever needed.

function function\_name(parameters){ statements; }

Function expression:

var variable\_name = function(parameters){ statements; }

and for calling the function use the function name (in this case is variable\_name).

[An expression always returns a value no matter how long they are but statement not always produces immediate results.]

Arrays:

A user defined type where values are of same type or in key value pair and are index based.

Ways to declare:

var variable\_name = ();

var variable\_name = new Array();

Some methods to add values to array

array\_name.push() : pushes the element at the last

array\_name.unshift(): pushes the element at the first

array\_name.pop(): removes the element from the last

array\_name.shift(): removes the element from the first

array\_name.indexOf(elementName): returns the position of the element in the array specified in the argument