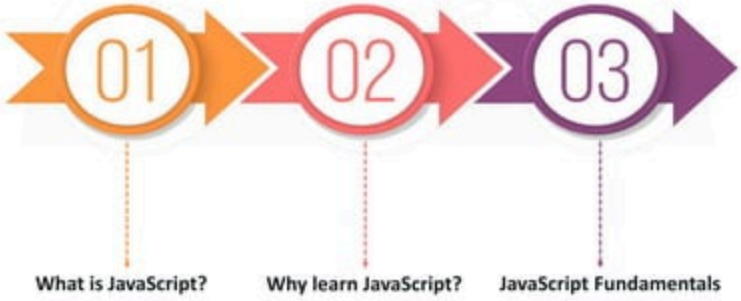
Javascript Tutorial

edureka!

Agenda


01
02
03
What is JavaScript?
Why learn JavaScript?
JavaScript Fundamentals

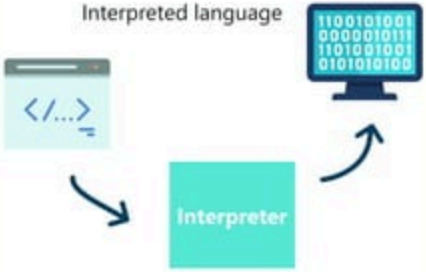
Introduction To JavaScript

What is JavaScript?

JavaScript is a scripting language that allows you to implement complex things on web pages.

Web pages more interactive

Interpreted language



Runs on the client's computer

JavaScript Stats

What Can JavaScript do?

Smart watches

Web application

Website

Games

JavaScript Framework

AngularJS

ReactJS

Meteor

METEOR

jQuery

jQuery
write less, do more.

The Big Picture – HTML, CSS & JavaScript

HTML

CSS

JavaScript

Top Websites Built Using JavaScript

Benefits Of JavaScript

Easy to learn

Speed

Benefits of JavaScript

Provides rich framework

Makes web pages more interactive

Benefits Of JavaScript

No compilation needed

Platform Independent

JavaScript Fundamentals

Variables

Variable is a name given to a memory location which acts as a container for storing data.

Syntax:

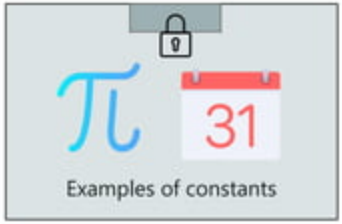
```
1  let age;
2  age = 22;
```

Constants

Constants are fixed values that do not change during execution time.

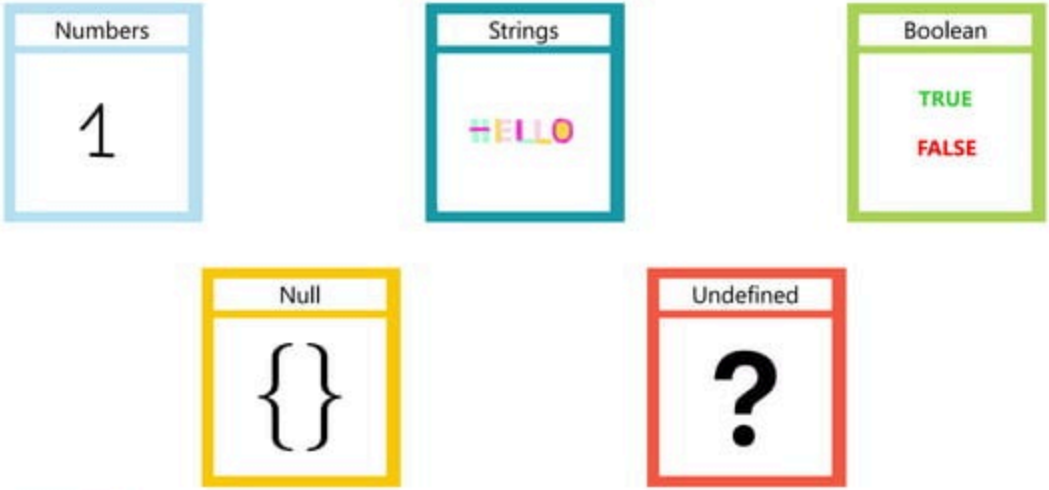
Syntax:

```
1   const mybirthday;
2   mybirthday = '03.08.1996' ;
```



Examples of constants

Primitive Data Types

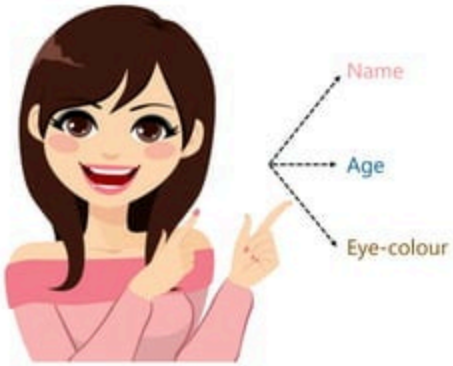


Reference Data Types - Objects

An object is a standalone entity, with properties and type.

Syntax:

```
1 | let object1 = { };
```

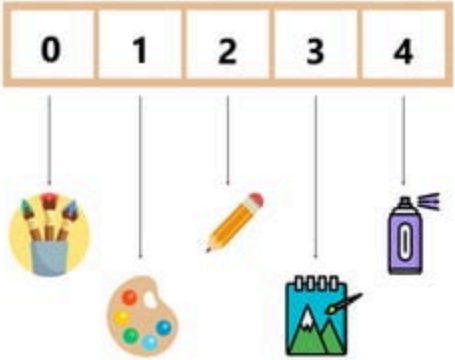


Reference Data Types - Arrays

An array is a data structure that contains a list of elements. These elements are all of the same data type, such as an integer or string.

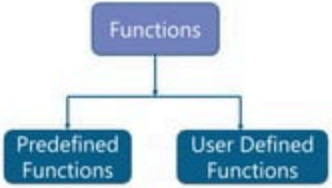
Syntax:

```
1   let arr[ ];
2   let arr = new Array( ) ;
```



Reference Data Types - Functions

A **function** is a block of organized, reusable code that is used to perform a single, related action.



name of the function

enter parameters here

Syntax:

```
1   function greetings( ) {
2       alert(' Hello everyone! ');
3   }
```

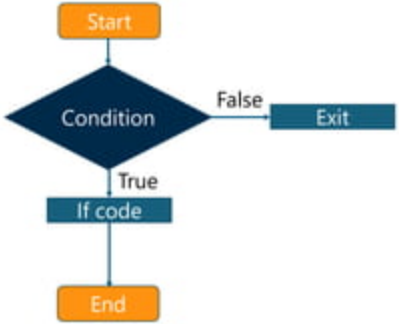
Conditional Statements

Conditional statement is a set of rules performed if a certain condition is met. It is like an 'If-Then' statement. (IF a condition is met, THEN an action is performed)

If Statement

Syntax:

```
1   if(condition) {
2   statement;
3   }
```



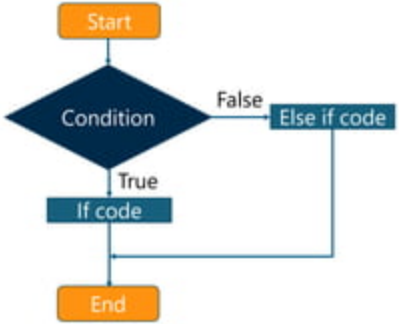
Conditional Statements

Conditional statement is a set of rules performed if a certain condition is met. It is like an 'If-Then' statement. (IF a condition is met, THEN an action is performed)

Else If Statement

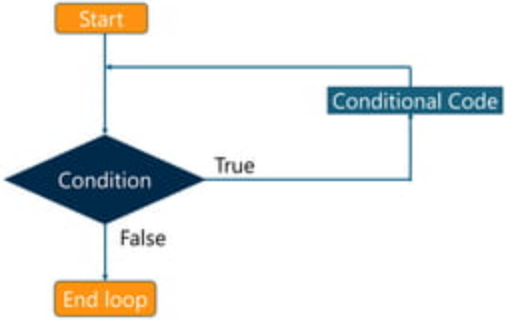
Syntax:

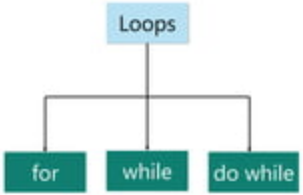
```
1   if(condition) {
2   statement a;
3   }
4   else (condition) {
5   statement b;
6   }
```



Loops

Loops are used to repeat a specific block until some end condition is met.





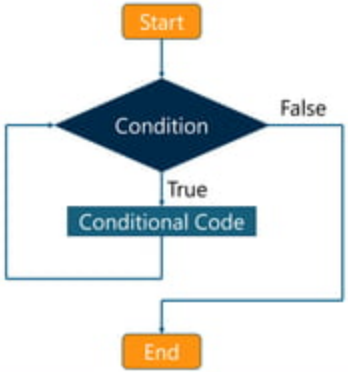
While Loop

While the **condition is true**, the code within the loop is executed.

While loop

Syntax:

```
1   while(condition) {
2   loop code;
3   }
```



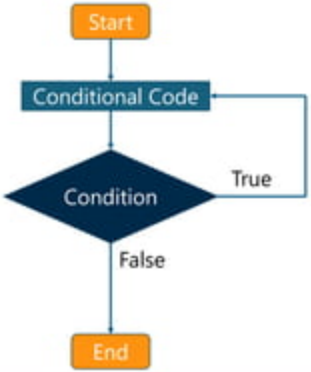
Do While Loop

This loop will **first execute the code**, then check the condition and while the condition holds true, execute repeatedly.

Do while loop

Syntax:

```
1    do {
2    loop code;
3    }
4    while(condition);
```



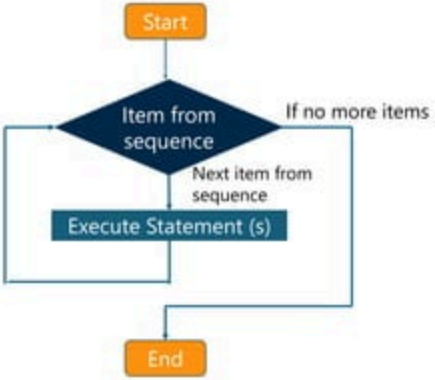
For Loop

Repeatedly executes the loop code while a given condition is TRUE. It tests the condition before executing the loop body.

For loop

Syntax:

```
1   for(begin; condition; step) {
2   loop code;
3   }
```



Switch Case

The switch statement is used to perform different actions based on different conditions.

Syntax:

```
1   switch(expression) {
2       case 1 :
3           code block 1
4           break;
5       case 2 :
6           code block 2
7           break;
8       default :
9           code block 3
10  }
```