Table of Contents

[1. Java script fundamentals 2](#_Toc67391619)

[1.Data Types 3](#_Toc67391620)

[2. let, const and var 4](#_Toc67391621)

[3. Basic Operators 4](#_Toc67391622)

[4. Strings and Template Literals 4](#_Toc67391623)

[5. Type Conversion and Coercion 5](#_Toc67391624)

[5. Truthy and Falsy Values 5](#_Toc67391625)

[6. Equality Operators: == vs. === 5](#_Toc67391626)

[7. Switch case 6](#_Toc67391627)

[8. History of JAVASCRIPT 7](#_Toc67391628)

[9. Strict Mode 7](#_Toc67391629)

[10. Function declaration vs expressions. 7](#_Toc67391630)

[11. Array 8](#_Toc67391631)

[12. Array Operations 9](#_Toc67391632)

[13 Objects dot vs bracket notation 9](#_Toc67391633)

[2. Development Guide 10](#_Toc67391634)

[1 Software lists and setup 10](#_Toc67391635)

[3. Java script in browsers 11](#_Toc67391636)

[1. DOM 11](#_Toc67391637)

[2. Guess My Number! Game 11](#_Toc67391638)

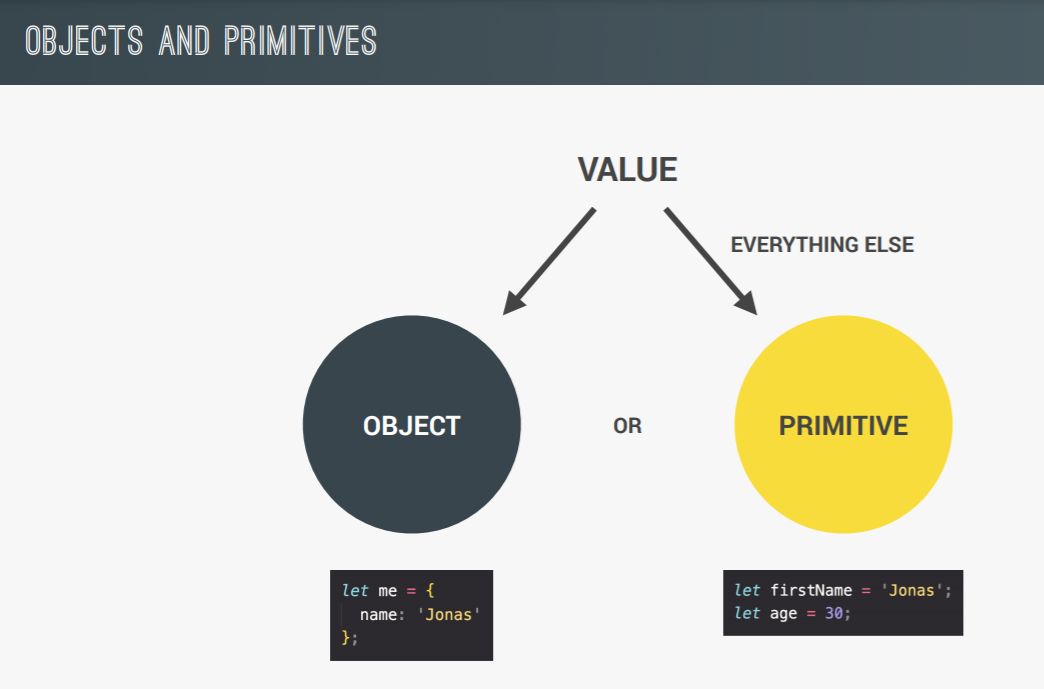
# Java script fundamentals

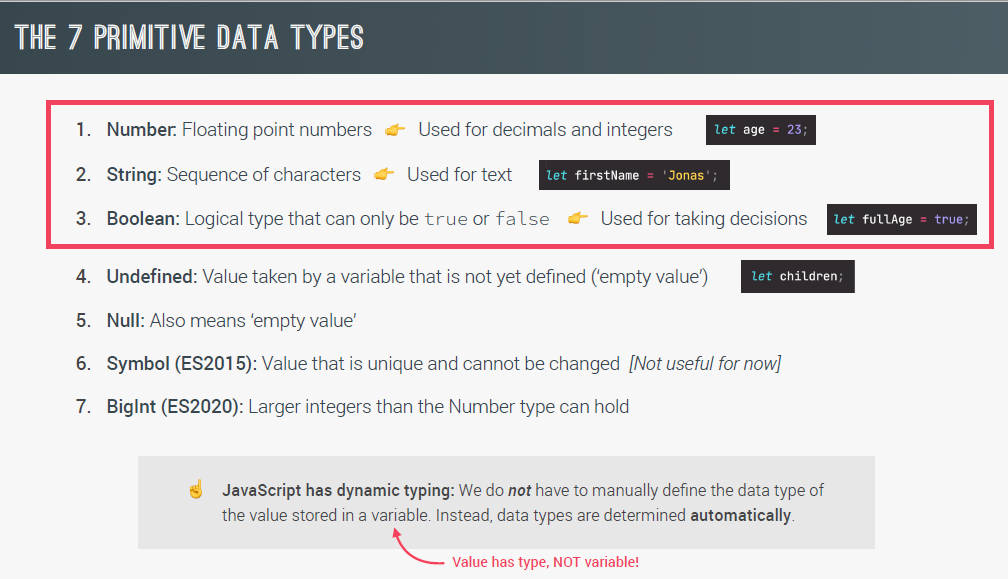
1. Java script is a high level, object oriented, multi-paradigm programming language.



1. JavaScript rule and convention for variable names is same as java.

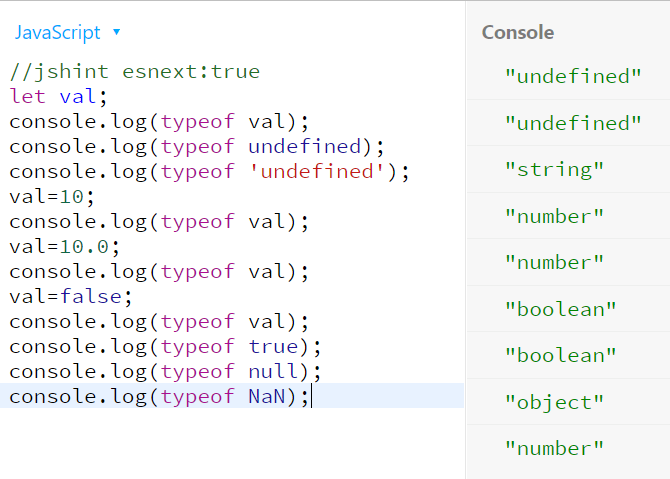
## 1.Data Types





1. In java script all the numeric are floating points. i.e. if(10===10.0) is true.
2. typeof will give the type of value particular variable hold.

Let val;



## 2. let, const and var

1. let and const are introduced in es6.



2. In javascript we can also create variable like this ->

newVar = 10;

console.log(newVar);

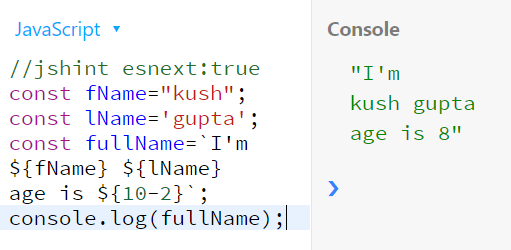
// it will create newVar variable and attach it to global object. Where as var is attached to functions scope only.

## 3. Basic Operators

1. 2\*3= 6, 2\*\*3=8 (it is just like pow), 10-8=2

2. plus operator can also be used to concat string.

## 4. Strings and Template Literals



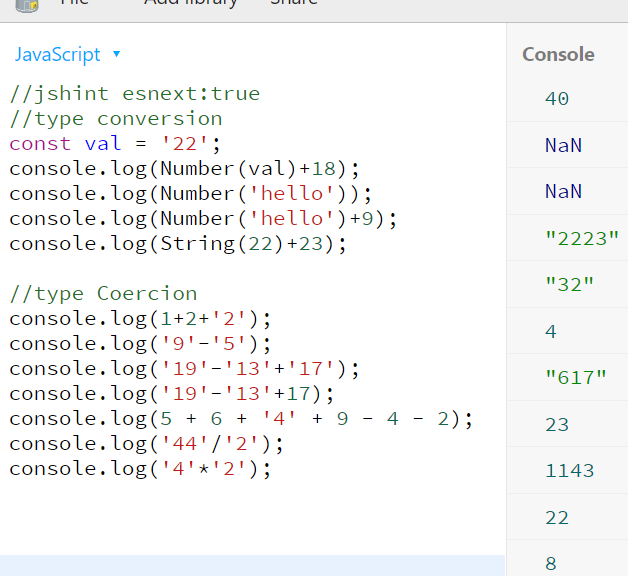
1. Template literals are introduced in es6 and provide below extra features –

1.1 can use multiline strings without need of writing \n\ for newline.

1.2 can use placeholders inside string ${} for actual value and can calculate expression also.

1.3 can use ‘ and “ also inside string without escape character.

## 5. Type Conversion and Coercion



1. type conversion is a way by which we can convert from one data type to other.

2. type coercion is way by which java script convert implicitly to some data type according to below rules ->

2.1 ‘+’ operator will perform mathematically add operation till numbers going from left to right. once string is found, number will be converted to string implicitly and concatenated.

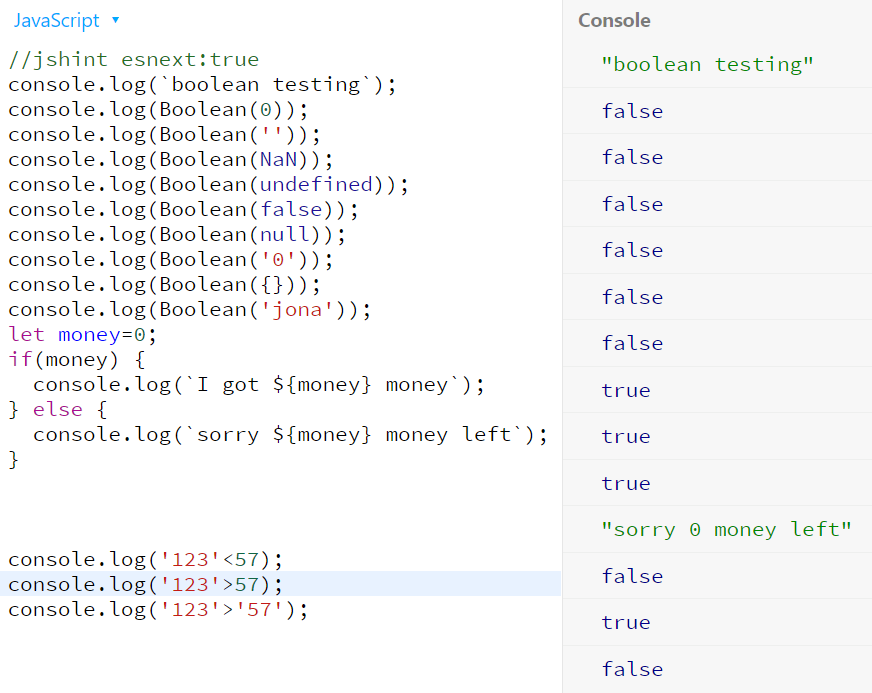
2.2 for ‘-‘ operator it will subtract normally. But for string like ‘22’-‘1’ it will convert string to number implicitly and give result in number i.e. 21.

2.3 same like minus operator, /,\* also works.

## 5. Truthy and Falsy Values

1. By default 0,’’,null,NaN,undefined and false are considered false values when used inside if block or logical operators.

2. if(undefined) or if(0) or if(null) will make code inside this if block not to be executed.



## 6. Equality Operators: == vs. ===

1. === check is strict checking which check data type and value both. Where as == is without strict check. == perform type coercion when needed to perform comparison.

2. !== and != are just not equal representation of not equals.

Example 🡪

let val='18';

if(val==18) {

console.log('== check');

}

if(val===18) {

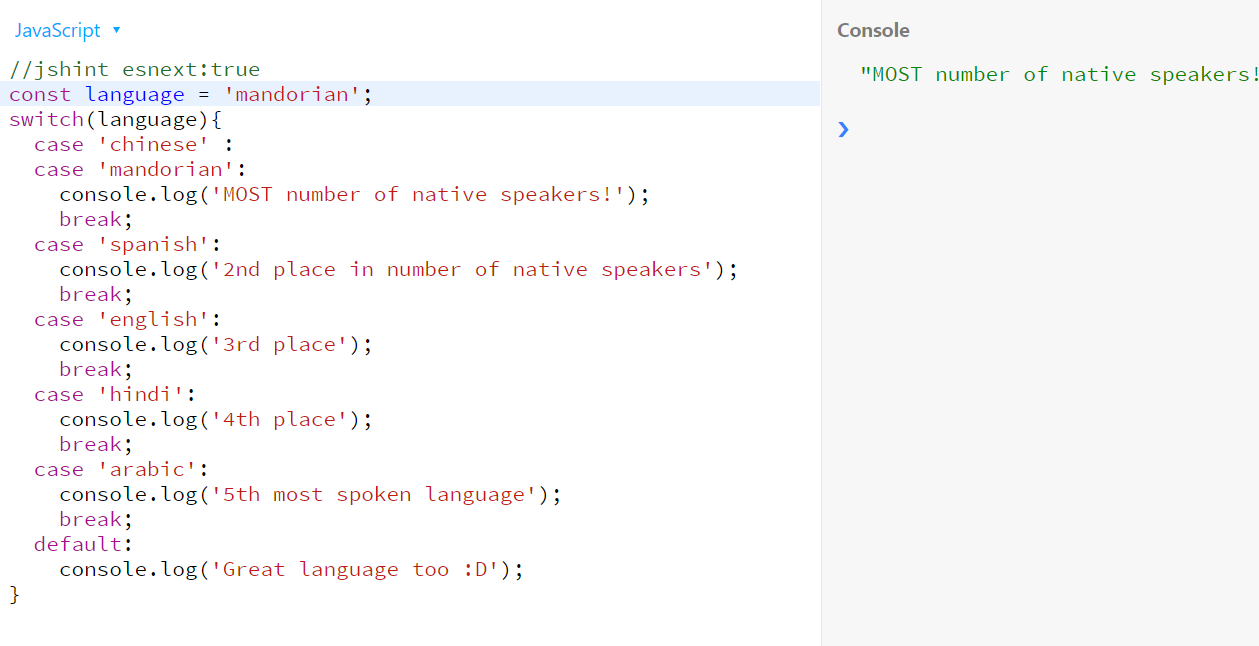
console.log('===check');

}

**Output ->**

== check

## 7. Switch case



## 8. History of JAVASCRIPT



## 9. Strict Mode

1. strict mode can be enabled by adding it at first line of java script file. Once enabled it will protect developer from some mistakes which javascript not complaint earlier.

Example 1->

const firstName = ‘kush’;

let value = 10;

if(firstName===’kush’) {

values=29;

}

Console.log(value); //output 10.

Above code works because js will create new variable values with 29. But with script mode enabled js will give error that values is not defined.

2. strict mode is enabled using below line

‘use strict’;

Example 2->

Strict mode restrict programmers to use reserved words to be used in code as a variable. Like interface, private, if etc. even though some of these reserved words are still not there in js and will be introduced in future.

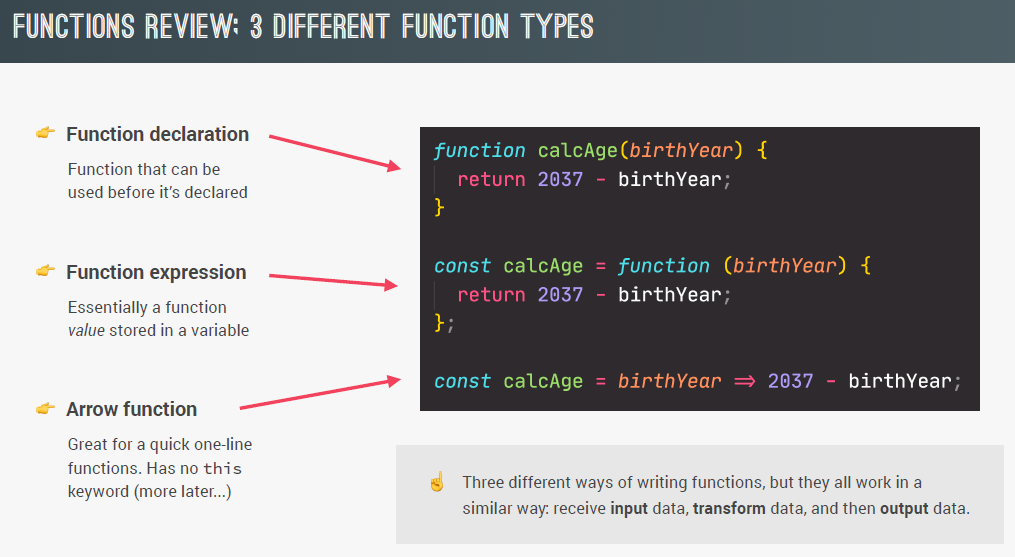
let private =’ss’; //give CT error if strict mode is on

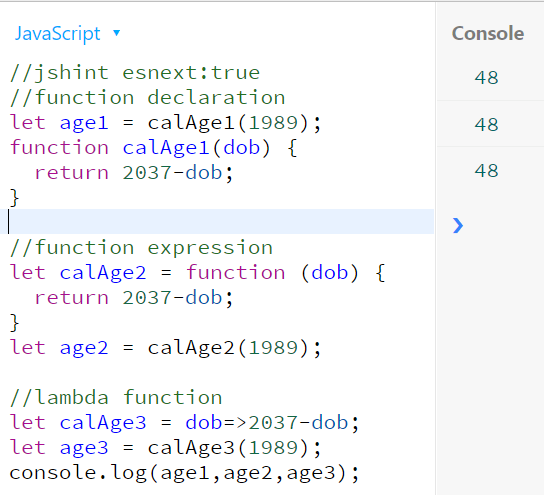
## 10. Function declaration vs expressions.

1. function expressions also called anonymous function and they are assigned to a variable like any other values.

2. anonymous function can be invoked only after it is assigned to a variable otherwise js will give CT error. But normal function can be called in line 1 and in later line will have definition.

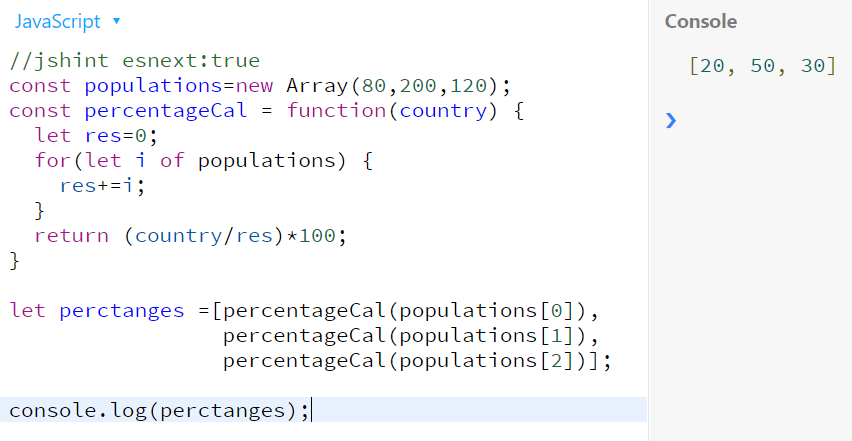
3. from es6 we can write lambda expression to represent function. It is just a simplified way of writing anonymous function. The difference b/w anonymous function and lamba is that lamba does not have this reference.

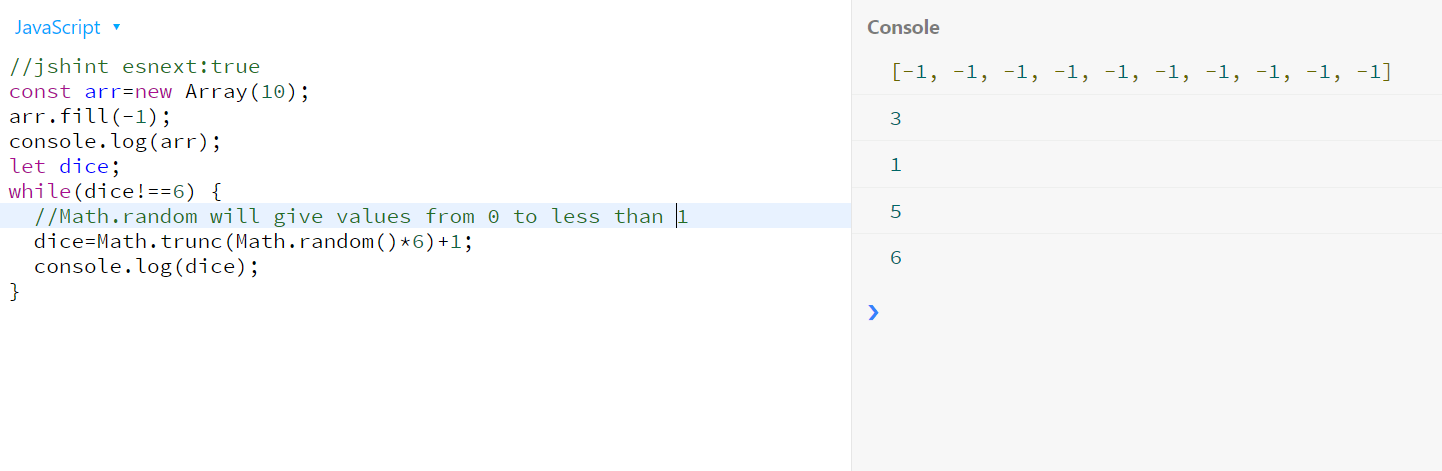




## 11. Array

1.there are two ways to create array. One using new operator and other using [] bracket. Always prefer [] bracket way as it will lead less confusion. As new Array(10,20) will create array with 2 elements. But new Array(10) will create array of size 10 with all 10 undefined values.





## 12. Array Operations

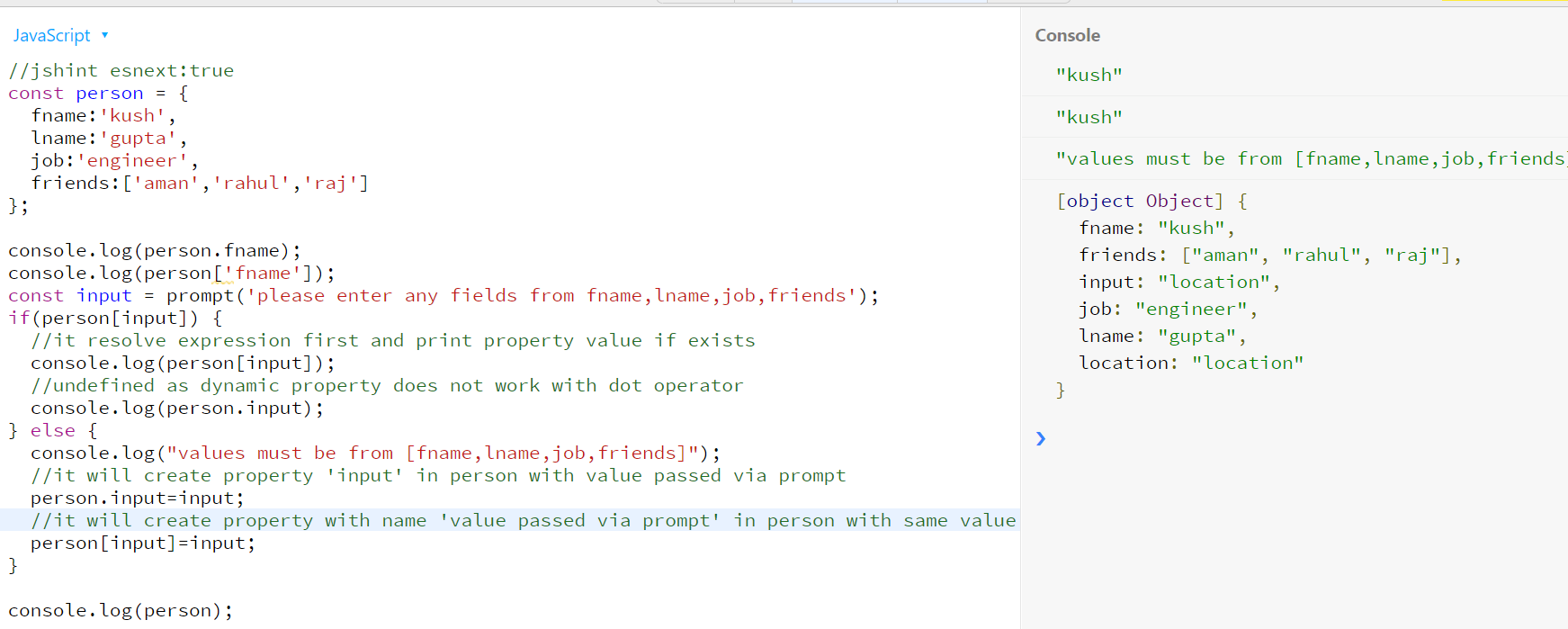


## 13 Objects dot vs bracket notation

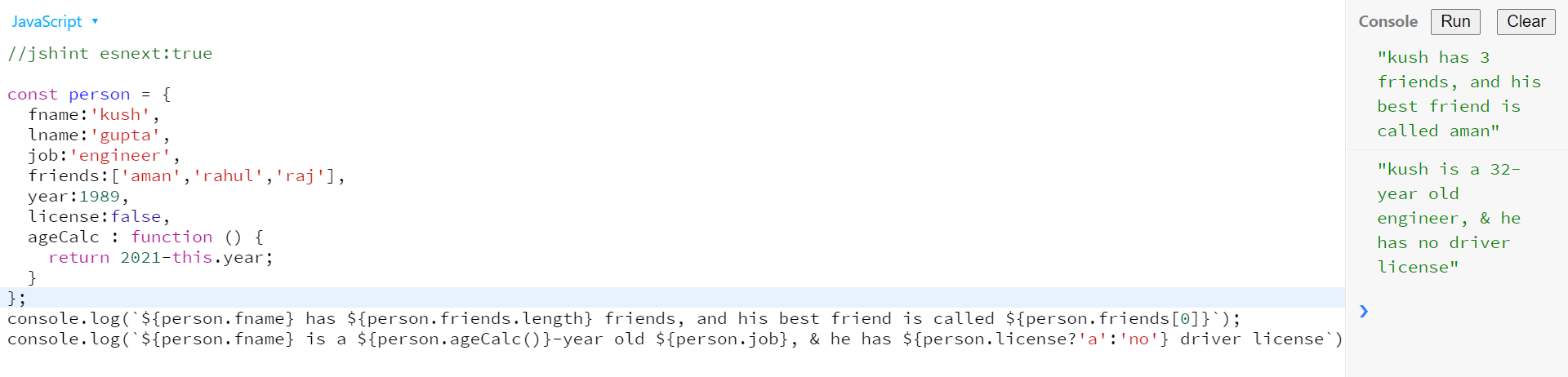
1. in java script objects contains key value pairs where key is always string and value can be anything.

2.every property value can be accessed either by dot or by bracket notation. Only difference is dot operator take static field. If we use field which does not exists both way give undefined value.

In below example input from prompt passed is location.



Example 2 with function as a value of a property.



Example 3 –



# Development Guide

## 1 Software lists and setup

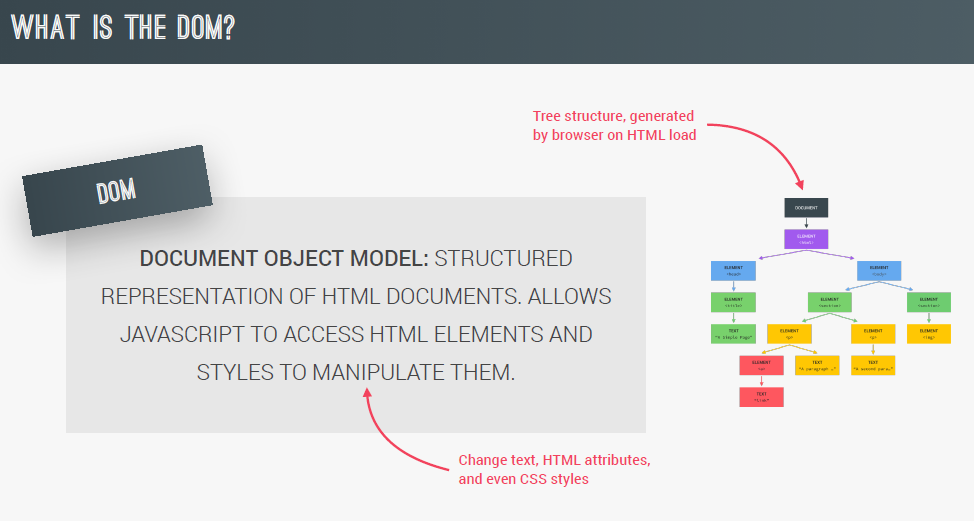
* Download and install vs code and node js.
* Go to terminal and check node -v for version.
* Type – npm install live-server -g
* Now in vs code create any html file and script file.
* Now goto path of above file via terminal and Type –> live-server
* It will launch application in browser with url -> <http://127.0.0.1:58442/>
* Further Changes will be hot deployed.

2 Debug

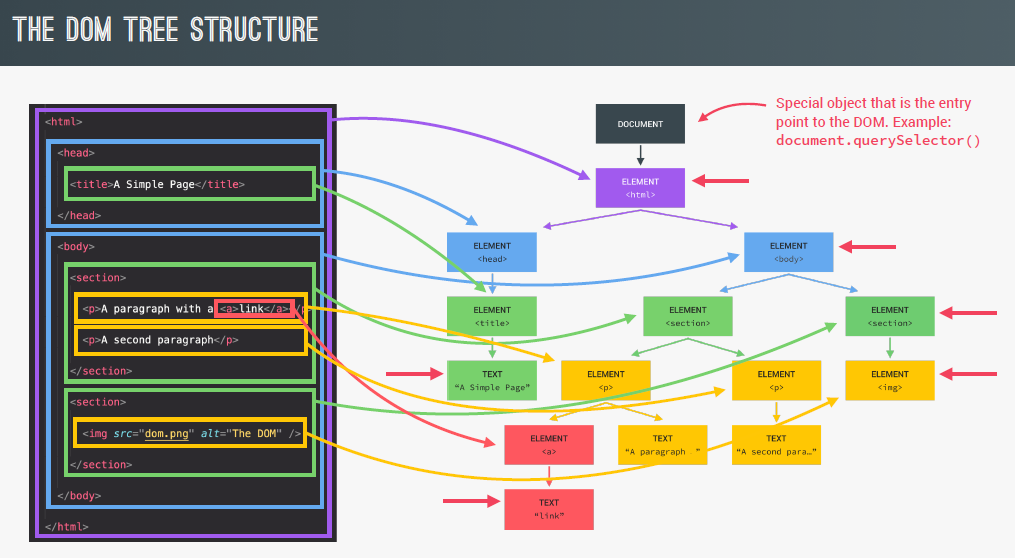
* console.log -> log with normal text
* console.warn -> log with warn text
* console.error -> log with error text
* console.table -> if logged with object. It will display in console in tabular form key,value fields.

# Java script in browsers

## 1. DOM



1. Tree structure root node is document. And by using document we can extract all other child nodes. example by document.querySelector(“#id”);
2. Html is the first child of document. As usually html page starts with html tag. And rest of the tree resembles the content inside the html page.
3. By using dom we can access html content inside js and can modify content according to need.



## 2. Guess My Number! Game

1. Computer will randomly select a Number from 1 to 21 and then ask the user to guess it in maximum 20 attempts.

2. once user enter the number which exceeds the random number, message will be displayed that It’s too high else too low.

3. On tight guess it will current score. And if it is higher then prev max it will be the max score.

4. user can play again by clicking again button.

Git utl –

https://github.com/kushguptacse/JavaScript/tree/main/guessMyNumber