1. Switch Statements :

A switch statement tests a value and can have many *case* statements which define various possible values. Statements are executed from the first matched case value until a break is encountered.

A : switch(number) {

case "1":

console.log("one");

break;

case "2":

console.log("two");

break;

}

## B : Adding a default option in switch statements

switch (num) {

case value1:

statement1;

break;

case value2:

statement2;

break;…

default: defaultStatement; break; }

## 2 : Replacing If Else Chains with Switch

3 : Returning boolean values from functions

## 4 : JavaScript Objects :

Objects are similar to arrays, except that instead of using indexes to access and modify their data, you access the data in objects through what are called properties.

Example :

var ME = { "name": "Nikita", "Age": 19, "branch": "C.S", "hobbies": ["dancing", "travelling"]};

* + 1. Accessing Object Properties with Dot Notation:

var val1 = ME.name; // val1

var val2 = ME.age; // val2

## B. Accessing Object Properties with Bracket N Notation:

var myObj = { "Space Name": "Kirk"}

myObj["Space Name"]; // Kirk

## C. Accessing Object Properties with Variables:

var dogs = { Fido: "Mutt", Hunter: "Doberman", Snoopie: "Beagle"};

var myDog = "Hunter";

var myBreed = dogs[myDog];

console.log(myBreed); // "Doberman"

## D. Updating Object Properties:

var myDog = {

"name": "Coder",

"legs": 4,

"tails": 1,

"friends": ["freeCodeCamp Campers"]

};

myDog.name="Happy Coder";

COMPLETED 75% OF BASICS OF JAVA-SCRIPT AND ALSO STARTED “10 DAYS OF JAVASCRIPT ON HACKERRANK AND DONE WITH 1 STAR TODAY.