# Points ToDo:

**Array ||| For (n) ||| For (iteration) ||| ForEach ||| Map ||| Filter ||| IF ||| Arrow Function**

MAP >> Returns Array/Object && ObjMembers**;**

FOREACH >> No Returns (write in body with Obj/ObjMembers);

FILTER >> Returns Array/Object && No ObjMembers;

FIND >> Returns Object && ObjMembers;

FINDINDEX >> Returns int;

# Console:

console.log("Hi")

console.warn("Hi")

console.error("Hi")

console.clear()

# Variables [var, let, const]:

var is global level and let & const has block level scope.

Can initialize “let” but can’t initialize “const”.

let name; const (xx) age;

name = “raghunadh”; age (xx) = 30;

# Data Types: [String, Numbers, Boolean, null, undefined]

const name = “Raghu”;

console.log(typeof name);

console.log(`My name is ${name}`);

console.log(name.split(‘’)); || .split(‘, ’);

console.log(name.length);

console.log(name.substring(0,5) .toUpperCase());

# Arrays:

Arrays are zero ‘0’ based and within ‘[Square Brackets]’. Also supports multiple data types. We can access the array through index based.

let fruits = ["apple", "mango", "goa", 1, true, null, {name: "raghunadh"}];

let single = fruits[6];

fruits.push("Banana"); >> (Push for insert at last index);

fruits.unshift("Grapes"); >> (Unshift for insert at first index);

fruits.pop(); >> (Removes last indexed value);

console.log(Array.isArray(fruits)); >> (Checks and returns the Boolean value);

console.log(fruits.indexOf("mango")) >> (Returns the index of value);

*//  for*

*//  for(let s of ss)*

*//  foreach(c => console.log(c.text));*

*//  map(c => c);*

*//  filter(f => f.id == 2);*

*//  find(c => c.id == 1);*

*//  findIndex(c => c.id == 3);*

*//  join(',');*

*//  indexOf('feb');*

# Object Literals:

let person = {

firstname: "raghu",

lastname: "nadh",

age: 30,

interests: ["chess", "driving", "coding"],

addres: {

city: "Hyderabad",

colony: "Kothapet"

}

}

console.log(person.addres.colony);

console.log(person.interests[person.interests.indexOf("chess")]);

let {firstname, lastname} = person; >> (Pulling values from object);

let {age, interests} = person; >> (Pulling values from object);

console.log(firstname, lastname); >> (Accessing pulled values from object);

console.log(interests[1]); >> (Accessing pulled values from object);

let {age, addres: {city}} = person;

console.log(city);

let todos = [

{id: 1, text: 'node.express', iscompleted: false},

{id: 2, text: 'angular', iscompleted: true},

{id: 3, text: 'mongoDB', iscompleted: false},

{id: 4, text: 'javascript', iscompleted: false},

{id: 5, text: 'node', iscompleted: false}

];

# \*General For Loop:

for(let i =0; i < 10; i++){ console.log(`Looped number % ${i}`);}

# \*Incremental For Loop:

for(let i = 0; i < todos.length; i++){ console.log(todos[i].text) }

# \*Iteration For Loop:

for(let todo of todos){ console.log(todo.text, ": ", todo.iscompleted);}

# \*ForEach Loop:

todos.forEach(function(todo){ console.log(todo.text);})

# \*Map Loop (Returns the Object Array):

const todoText = todos.map(function(todo){ return todo.text;})

# \*Filter (Returns the Object Array):

const results = todos.filter(function(todo){ return todo.iscompleted == true;})

Example:

const results = todos.filter(function(t){ return t.iscompleted == true;}).map(function(t){ return t.text;})

# \*Ternary Operator:

const color = x ? 'red': 'blue';

# \*Functional Arrows:

const todo = (n) => console.log("This is: " + n);

Note: todo =()/todo=(n) >> function todo(); | remain part is body | if multi lines need ‘{ }’;

# Sort:

Const sortedCompanies = companies.sort((a,b) => (a.age > b.age ? 1: -1));

# Reduce:

Const ageSug = ages.reduce((total, age) => total + age, 0);

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

**String Functions**:

\*//toUpperCase; subString; length; split; indexOf; trim; typeof; replace; charAt; toString; search;

**Array Functions**:

\*//push; unshift; pop; Array.isArray; toString; length; join; replace, reverse; indexof;

**Iteration Functions**:

\* //for i; for a of as; foreach; filter; find; join; map; sort;

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

const nameString = 'raghunadh';

const ageNumber  = 38;

const brosArray = ['ravi', 'raghu', 'raju'];

const studentObj = {id: 28, name: 'raghu', location: 'hayatnagar'};

const {name, location} = studentObj;

const numberFloat = 33.3;

@@@@@@@@@@@@@@@@@@@@@@@@