**ES6- Session-2**

**Q1. Filter unique array members using Set.**

**Ans.**

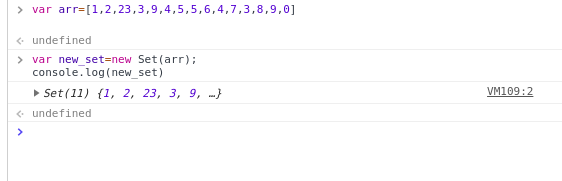
Code:

var arr=[1,2,23,3,9,4,5,5,6,4,7,3,8,9,0]

var new\_set=new Set(arr);

console.log(new\_set)

Output:

****

**Q2. Find the possible combinations of a string and store them in a MAP?**

**Q3. Write a program to implement inheritance upto 3 classes.The Class must public variables and static functions.**

**Ans.**

Code:

class Animal {

constructor(name) {

this.name = name; }

static speak() {

console.log(this.name + ' makes a noise.'); }

}

class Dog extends Animal {

constructor(name) {

super(name);

}

speak() {

console.log(this.name + ' barks.');

}

}

class Lion extends Animal {

constructor(name) {

super(name);

}

speak() {

console.log(this.name + ' roar.');

}

}

let a=new Animal("Elephants");

Animal.speak();

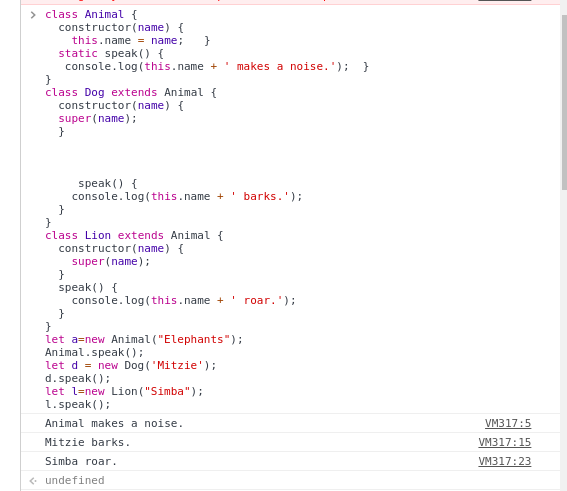
let d = new Dog('Mitzie');

d.speak();

let l=new Lion("Simba");

l.speak();

Output:



**Q4.Write a program to implement a class having static functions**

**Ans.**

**Code:**

class Animal {

constructor(name) {

this.name = name;

}

static speak() {

console.log(this.name + ' makes a noise.');

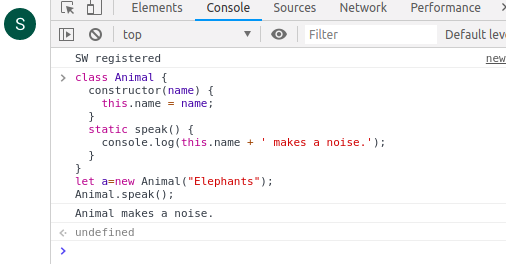
}

}

let a=new Animal("Elephants");

Animal.speak();

**Output:**

****

**Q5.Import a module containing the constants and method for calculating area of circle, rectangle, cylinder.**

**Ans.**

//module1.js

export function AreaOfCircle(r){

return 3.14\*(r\*r);

}

export function AreaOfRect(l,b){

return l\*b;

}

export function AreaOfCylinder(h,r){

return (2\*3.14\*r\*h+2\*3.14\*r\*r);

}

//Exec.js

import {AreaOfRect,AreaOfCircle,AreaOfCylinder} from './module1';

document.getElementById("alpha").innerHTML=AreaOfCircle(5);

**Q6.Import a module for filtering unique elements in an array**

**Ans.**

//module2.js

export function onlyUnique(value, index,self) {

return self.indexOf(value) === index;

}

console.log(unique)

//Exec2.js

import {filterModule} from './module2';

var a = ['a', 1, 'a', 2, '1'];

var unique = a.filter( onlyUnique );

Q7.**Write a program to flatten a nested array to single level using arrow functions**

**Ans.**

var Array1 = [[1, 2],[3, 4, 5], [6, 7, 8, 9]];

var NewArray2 = [];

for (var i = 0; i < Array1.length; ++i) {

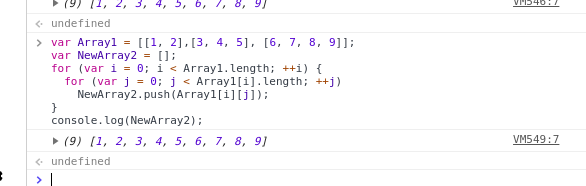
for (var j = 0; j < Array1[i].length; ++j)

NewArray2.push(Array1[i][j]);

}

console.log(NewArray2);

Output:

**Q9Implement Map and Set using Es6?**

**Ans.**

**Code:**

var myMap = new Map();

var keyString = 'afgj ';

var keyObj = {};

myMap.set(keyString, "value with 'a string'");

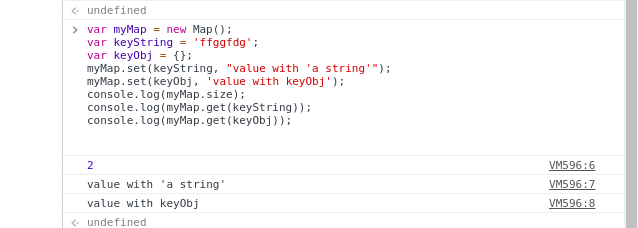
myMap.set(keyObj, 'value with keyObj');

console.log(myMap.size);

console.log(myMap.get(keyString));

console.log(myMap.get(keyObj));

Output:



-- code:

var mySet = new Set();

mySet.add(1);

mySet.add(5);

mySet.add(5); //this wont get added

mySet.add('hbbjf');

var o = {a: 1, b: 2};

console.log(mySet.add(o));

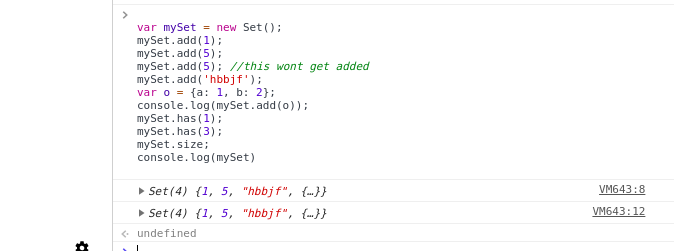
mySet.has(1);

mySet.has(3);

mySet.size;

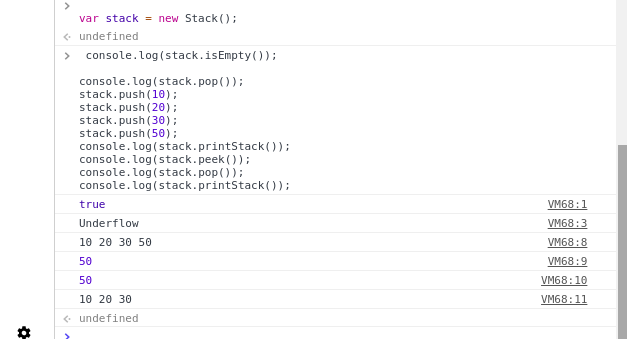
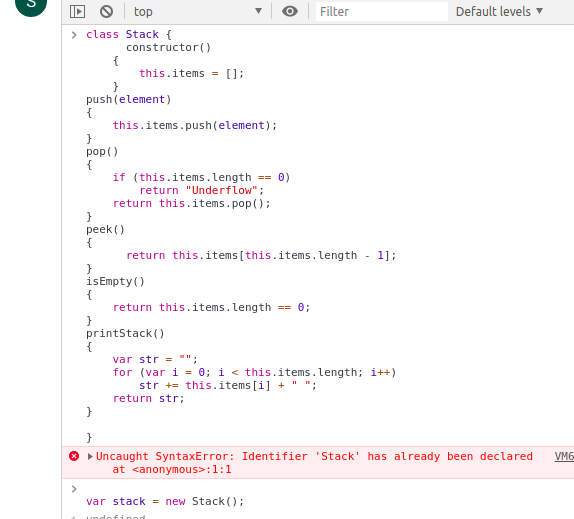
console.log(mySet)

Output:



**Q10.Implementation of stack (using linked list)**

Ans.



class Stack {

constructor()

{

this.items = [];

}

push(element)

{

this.items.push(element);

}

pop()

{

if (this.items.length == 0)

return "Underflow";

return this.items.pop();

}

peek()

{

return this.items[this.items.length - 1];

}

isEmpty()

{

return this.items.length == 0;

}

printStack()

{

var str = "";

for (var i = 0; i < this.items.length; i++)

str += this.items[i] + " ";

return str;

}

}

var stack = new Stack();

console.log(stack.isEmpty());

console.log(stack.pop());

stack.push(10);

stack.push(20);

stack.push(30);

stack.push(50);

console.log(stack.printStack());

console.log(stack.peek());

console.log(stack.pop());

console.log(stack.printStack());

**Find the possible combinations of a string and store them in a MAP?**

**Ans:-** I tried but couldn’t create a logical program for this.

**Q8.Implement a singly linked list in es6 and implement addFirst() addLast(), length(), getFirst(), getLast(). (without using array)**

**Ans:-** I tried but couldn’t create a logical program for this.