**Assignment1\_ONE.ts**

**Program**

function Maximum(num:number[]):number

{

var max:number = num[0];

var iCount:number = 0;

for(iCount = 1;iCount < num.length;iCount++)

{

if(max < num[iCount])

max = num[iCount];

}

return max;

}

var lstNumbers:number[] = [23,89,6];

var largestNumber:number = Maximum(lstNumbers);

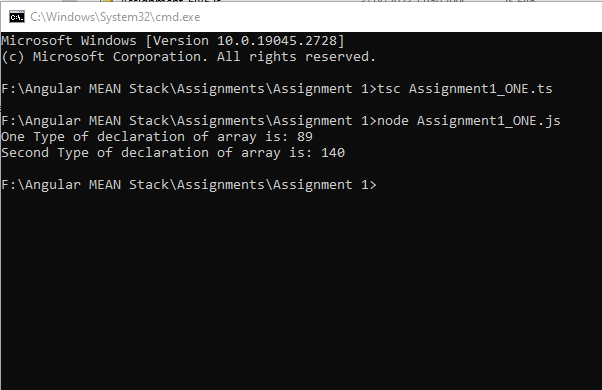
console.log("One Type of declaration of array is: "+largestNumber);

var lstNumbersType2:number[] = new Array(10,20,30,140,50,60,90,);

var largestNumber:number = Maximum(lstNumbersType2);

console.log("Second Type of declaration of array is: "+largestNumber);

**Output:**



**Assignment1\_TWO.ts**

**Program**

function Area(radius:number):number

{

var area:number = 0;

const pi:number = 3.14;

area = pi\*radius\*radius;

return area;

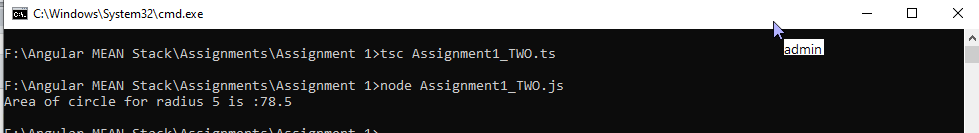
}

var radius = 5;

var area:number = Area(radius);

console.log("Area of circle for radius "+radius+" is :"+area);

**Output:**



**Assignment1\_THREE**

**Program:**

function DisplayFactors(InputNumber:number):number[]

{

var iCount = 1;

var lstFactors:number[] = new Array();

for(iCount = 1;iCount < InputNumber;iCount++)

{

if(InputNumber % iCount == 0)

lstFactors.push(iCount);

}

return lstFactors;

}

var InputNumber:number = 90;

var lstFactors:number[] = DisplayFactors(InputNumber);

var iCount:number = 0;

console.log("Factors of " +InputNumber+"are as shown below:");

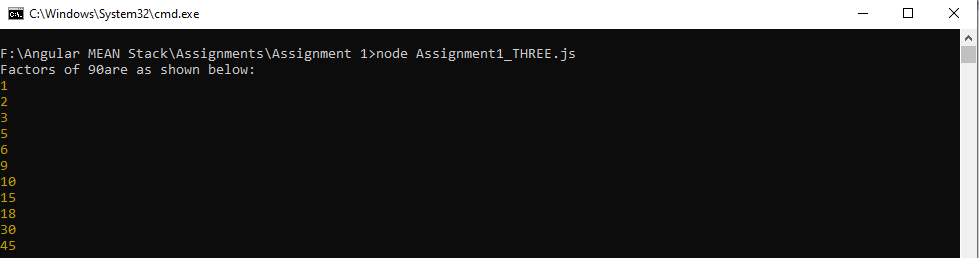
for(iCount = 0;iCount<lstFactors?.length;iCount++)

{

console.log(lstFactors[iCount]);

}

**Output:**



**Assignment1\_FOUR.ts**

**Problem:**

function Fibonaci(p\_iNumber:number):void

{

var iNumber1:number = 0;

var iNumber2:number = 1;

var Sum:number = 0;

while(Sum < p\_iNumber)

{

console.log(iNumber2);

Sum = iNumber1+iNumber2;

iNumber1 = iNumber2;

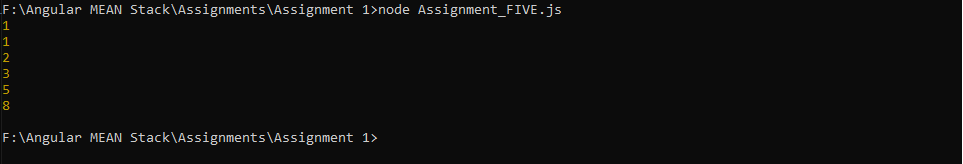
iNumber2 = Sum;

}

}

Fibonaci(12);

**Output:**



**Assignment1\_FIVE.ts**

**Program:**

function IsItPrimeNumber(p\_iNumber:number):boolean

{

var f\_boolStatus:boolean = false;

var iCount:number = 1;

if(p\_iNumber == 2)

{

f\_boolStatus = true;

return f\_boolStatus;

}

else if(IsEvenNumber(p\_iNumber))

{

f\_boolStatus = false;

return f\_boolStatus;

}

var iDivisibleCount:number = 0;

for(iCount=1;iCount<=p\_iNumber;iCount++)

{

if(p\_iNumber % iCount == 0)

iDivisibleCount++;

}

if(iDivisibleCount == 2)

{

f\_boolStatus = true;

return f\_boolStatus;

}

else if(iDivisibleCount > 2)

{

f\_boolStatus = false;

return f\_boolStatus;

}

return f\_boolStatus;

}

function IsEvenNumber(p\_iNumber:number):boolean

{

if(p\_iNumber % 2 == 0)

return true;

else

return false;

}

var InputNumber:number = 12;

if(InputNumber == 1)

console.log("1 is a non-prime number");

else

{

if(IsItPrimeNumber(InputNumber))

console.log(InputNumber+ " is Prime Number");

else

console.log(InputNumber + " is not Prime Number");

}

**Output:**