

## Assignment No 1

Write a typescript program which contains one function named as Maximum. That function accepts three parameters and it should returns largest value from three input parameters.

Input :23 89 6

Output: Maximum number is 89.

```
function
Maximum(value1:number,value2:number,value3:number):void
{
    if(value1 < value2 && value2 > value3)
    {
        console.log("Maximum is :"+value2);
    }
    else
    if(value2 < value1 && value1 > value2)
    {
        console.log("Maximum is :"+value1);
    }
    else
    if(value1 < value3 && value3 > value2)
    {
        console.log("Maximum is :"+value3);
    }
}
var no1 : number = 23;
var no2 : number = 89;
var no3 : number = 6;
Maximum(no1,no2,no3);
```

```
harshal@asus-vivobook: ~/Marvellous-Infosystem/Angular/Assignment$ tsc lassign1.t
s
harshal@asus-vivobook: ~/Marvellous-Infosystem/Angular/Assignment$ node lassign1.
js
Maximum is :89
harshal@asus-vivobook: ~/Marvellous-Infosystem/Angular/Assignment$
```

2. Write a typescript program which contains one function named as Area. That function should calculate area of circle. Accept value of radius from user and return its area. Default value of PI should be 3.14 if it is not provided by the caller.  
Input : 5  
Output: Area of circle is 78.5

```
function Area(value1:number):number
{
var Ans:number = 3.14 * value1 * value1;
return Ans;
}
var radius : number = 5;
var iRet:number = 0;
iRet = Area(radius);
console.log("Area of Circle : "+iRet);
```

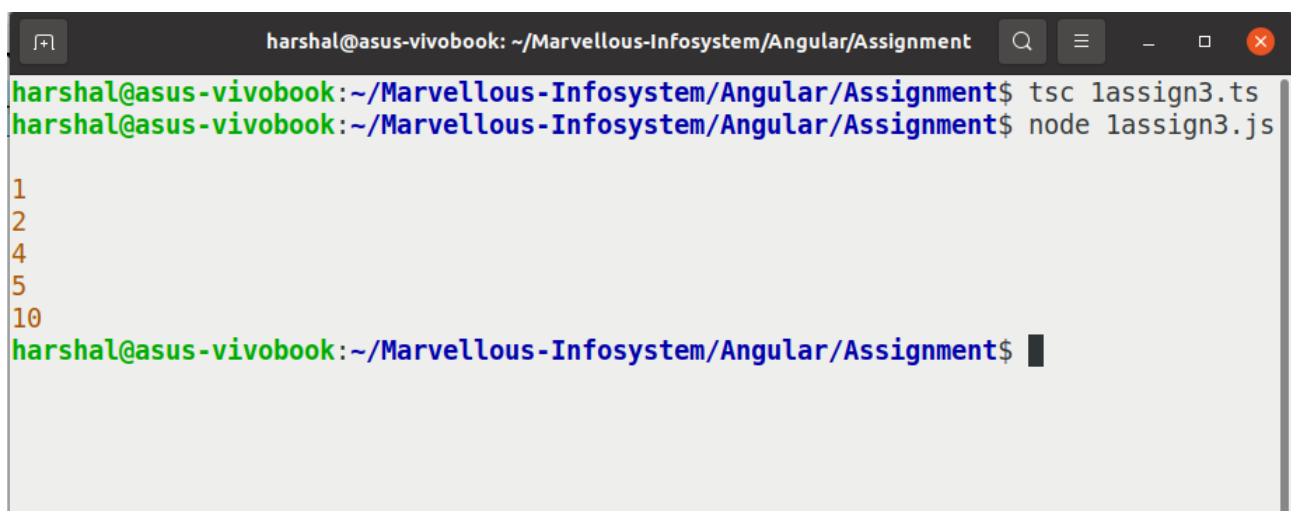
```
harshal@asus-vivobook: ~/Marvellous-Infosystem/Angular/Assignment$ tsc lassign2.ts
harshal@asus-vivobook: ~/Marvellous-Infosystem/Angular/Assignment$ node lassign2.js
Area of Circle : 78.5
harshal@asus-vivobook: ~/Marvellous-Infosystem/Angular/Assignment$
```

3. Write a typescript program which contains one function named as DisplayFactors. That function should accept one number and display factors of that number.

Input : 20

Output: 1 2 4 5 10

```
function Factor(value1:number):void
{
var iCnt:number = 0;
for(iCnt = 1;iCnt < value1;iCnt++)
{
if(value1 % iCnt == 0)
{
console.log(iCnt);
}
}
}
var fact : number = 20;
Factor(fact);
```

A terminal window titled 'harshal@asus-vivobook: ~/Marvellous-Infosystem/Angular/Assignment'. It shows the command 'tsc 1assign3.ts' being executed, followed by 'node 1assign3.js'. The output of the program is displayed as '1', '2', '4', '5', and '10' on separate lines. The terminal prompt is 'harshal@asus-vivobook:~/Marvellous-Infosystem/Angular/Assignment\$'.

4. Write a typescript program which contains one function named as ChkPrime. That function should accept one number and it should return true if the given number is prime and otherwise return false.

Input : 11

Output: It is prime number

```

function Prime(value1:number):boolean
{
var iCnt:number = 0;
for(iCnt = 2;iCnt < value1;iCnt++)
{
if(value1 % iCnt != 0)
{
return true
}
else
{
return false
}
}
}
var no : number = 11;
var bRet : boolean;
bRet = Prime(no);
if(bRet == true)
{
console.log("Number is Prime")
}
else
{
console.log("Number is not Prime")
}

```

```

harshal@asus-vivobook: ~/Marvellous-Infosystem/Angular/Assignment
harshal@asus-vivobook:~/Marvellous-Infosystem/Angular/Assignment$ tsc lassign4.ts
harshal@asus-vivobook:~/Marvellous-Infosystem/Angular/Assignment$ node lassign4.js

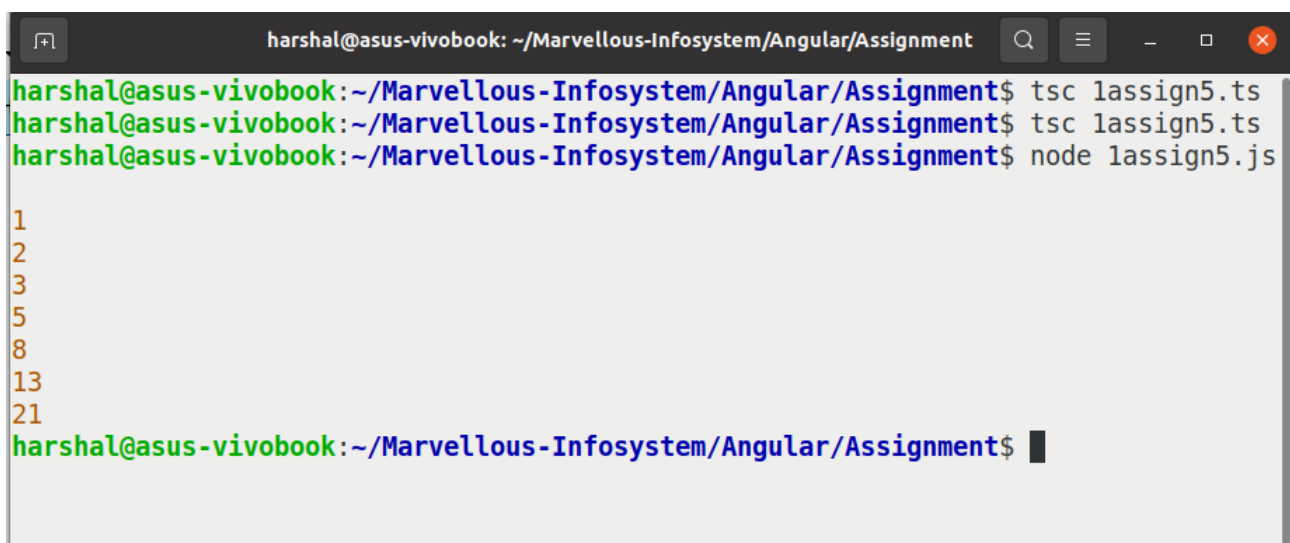
Number is Prime
harshal@asus-vivobook:~/Marvellous-Infosystem/Angular/Assignment$

```

5. Write a typescript program which contains one function named as Fibonacci. That function accept one number from user and print Fibonacci series till that number.  
Input : 21

Output: 1 1 2 3 5 8 13 21

```
function Fibonacci(value1:number):void
{
var iCnt:number = 0;
var ans:number = 0;
var no1 : number = 0;
var no2:number = 1;
for(iCnt = 1;iCnt < value1;iCnt++)
{
ans = no1 + no2;
if(value1 < ans)
{
break;
}
console.log(ans);
no1 = no2;
no2 = ans;
}
}
var no : number = 21;
var bRet : boolean;
Fibonacci(no);
```

A terminal window titled 'harshal@asus-vivobook: ~/Marvellous-Infosystem/Angular/Assignment' shows the execution of a Fibonacci program. The user runs 'tsc lassign5.ts' twice and 'node lassign5.js' once. The output of the program is displayed as a list of numbers: 1, 2, 3, 5, 8, 13, and 21.

```
harshal@asus-vivobook: ~/Marvellous-Infosystem/Angular/Assignment$ tsc lassign5.ts
harshal@asus-vivobook: ~/Marvellous-Infosystem/Angular/Assignment$ tsc lassign5.ts
harshal@asus-vivobook: ~/Marvellous-Infosystem/Angular/Assignment$ node lassign5.js
1
2
3
5
8
13
21
harshal@asus-vivobook: ~/Marvellous-Infosystem/Angular/Assignment$
```