Task

1. **What will be the output of this code?**
   Console.log(x);
   Var x=5;
   o/p: undefined
   Explanation : concept of hoisting a variable in javascript variable moved to the top of their scope before code execution .so we assign number does not take .

2. **What will be the output of this code ?**
   Console.log(a);
   Var a;
   o/p: undefined
   Explanation : concept of hoisting a variable in javascript variable moved to the top of their scope before code execution .

3. **What will be the output of this code?**
   console.log(b);
   b=10;
   var b
   o/p: undefined
   Explanation : concept of hoisting a variable in javascript variable moved to the top of their scope before code execution .so we assign number does not take .

4. **What will happen here?**
   console.log(c);
   o/p: Reference error c is not defined.
   Explanation : Here try to accessing a variable in the code we are not declare any variable so without declaration we cannot access any variable

5. **What will be the output of this code?**

   console.log(e);

   var e = 10;

   console.log(e);

   e= 20;

   console.log(e);

   o/p: undefined

   10

   20

Explanation: step:1-variable is declared but not assigned a value then output will be undefined.

Step2: variable e has been assigned value 10. then output will be 10

Step3: : variable e has been assigned value 20. then output will be 20

6. **What will be the output of this code?**

Console.log(f);

var

f=100;

var f;

console.log(f);

o/p:

undefined 100

Explanation: - step1: variable is declared but not assigned a value then output will be undefined.

Step2: variable f has been assigned value 100.then output will be 100

Step3: there is not effect on f because variable already declared

7. **What will be output of this code?**

console.log(g);

var g=g+1;

console.log(g);

o/p: undefined

NaN

Explanation: trying to add two different data types that is number and string we can't add by using + operator so then output will be NaN.

8. **What will be the output of this code?**

var h;

console.log(h);

```
 h=58;

 console.log(h);
```

o/p: undefined

     58

Explanation: step1: concept of hoisting a variable in javascript variable moved to the top of their scope before code execution .so we assign number does not take .

Step2: variable h has been assigned value 58.

9. What will be the output of this code?

```
 console.log(t);

 t = 16;

 var t = 5;

 console.log(t);
```

o/p:undefined

    5

Explanation: step1: concept of hoisting a variable in javascript variable moved to the top of their scope before code execution .so we assign number does not take .

Step2: In java script excecution done one -by-one so we assign a value same variable it takes recent assigned value this occurs in this program