JAVA SCRIPT Question Solving

---

1)INPUT-1: **OUTPUT:**

```
console.log(x);
var x = 5;
```

```
Undefined
```

Explanation:
We know that,in **JS** we use **var** to declare a variable, that declaration will be done(**Hoisted**) at the top of the code(**scope**),It doesn't matter that the declaration is done After or Before the **console.log()**.In this code we see that ,The **console.log(x)**. Is performed first and then goes to 2nd line which is **var x=5;** which means Declaration.Thats why the code prints output as "**Undefined**".

2)INPUT-2: **OUTPUT:**

```
console.log(a);
var a;
```

```
Undefined
```

Explanation:
We know that,in **JS** we use **var** to declare a variable, that declaration will be done(**Hoisted**) at the top of the code(**scope**),It doesn't matter that the declaration is done After or Before the **console.log()**.In this code we see that ,The **console.log(a)**. Is performed first and then goes to 2nd line which is **var a;** which means Declaration.Thats why the code prints output as "**Undefined**".

3)INPUT-3: **OUTPUT:**

```
console.log(b);
b = 10;
var b;
```

```
Undefined
```

Explanation:
Here the variable **b** is declared using **var**. This declaration is hoisted to the top of the scope.so when **JS interpreter** compiles the code, it hoists the declaration of **b** `so that this` will be considered as declaration first **var b;** and the **console.log(b);** in next step and it executes as the "variable is declared but value is not assigned" so the output is "**Undefined**".

4)INPUT-4:

```
console.log(c);
```

OUTPUT:

```
ReferenceError: c is not defined
```

Explanation:

We use **var** for declaration,But here the variable is not declared or assigned in entire code.In the code **JS Interpreter** will execute the 1st line which is **console.log(c);** ,But here there is nor variable declaration of "**c**".Hence it gives out as "**ReferenceError**" and gives details statement as" **c is not defined**".

6)INPUT-6:

```
console.log(e);
var e = 10;
console.log(e);
e = 20;
console.log(e);
```

OUTPUT:

```
Undefined
10
20
```

Explanation:

Here the **JS Interpreter** executes the 1st line which is **console.log(e)**,here the declaration of **var** & **assigning** done after the **console.log(e)** so here it prints output as "**Undefined**".Coming to 3rd line which is **console.log(e)**,here the declaration and assigning both are done at 2nd line of assigning value **10** to **var e** , So it prints output as "**10**'.Coming to 5th line which is also **console.log(e)**, here the re-assigning the vale **20** to the **var e** is done before this line execution , So it prints output as "**20**"**.**

7)INPUT-7:

```
console.log(f);
var f = 100;
var f;
console.log(f);
```

OUTPUT:

```
Undefined
100
```

Explanation:

Here the **console.log(f)** given above, declaration and the variable **`f`** is declared with **`var`**, so the declaration is hoisted to the top of the scope. And no value is assigned to the declaration , So it prints the output as"**Undefined**".Coming to 4th line which is **console.log(f)**,here the variable "**var**"declaration and value assigning of "**100**" to the "**var f**" both are done in previous line i.e.,in 3rd line , So it prints output as"**100**".

8)INPUT-8:

```
console.log(g);
var g = g + 1;
console.log(g);
```

OUTPUT:

```
Undefined
 NaN
```

Explanation:

Here the **JS Interpreter** executes the 1st line which is **console.log(g)**,here the variable “**g**” is declared in the next line,By hoisting property the declaration part of **var** moves to the top of the scope.Hence it prints output as “**Undefined**”.And coming to 3rd line which is ”**console.log(g);**”. Here for the variable “**g**” the declaration is done in the previous line,But the value of “**g**” is assigned as “**g+1**” this means “**Undefined + 1**” .Hence it prints output as “**NaN**” which means Not a Number.

9)INPUT-9:

```
var h;
console.log(h);
h = 50;
console.log(h);
```

OUTPUT:

```
Undefined
 50
```

Explanation:

Here the **JS Interpreter** executes the 2nd line which is **console.log(h)**,here the variable “**h**” is declared in the previous line as **var h**,Hence it prints output as “**Undefined**”.And coming to 4th line which is **console.log(h)** ,here the value “**50**” is assigned to variable”**h**” ,So it prints the output as”**50**”.

10)INPUT-10:

```
console.log(i);
i = 10;
var i = 5;
console.log(i);
```

OUTPUT:

```
Undefined
5
```

Explanation:

Here the **JS Interpreter** executes the 1th line which is **console.log(i)** ,here the variable “**i**” is declared in next line as “**i** = **10**” ,So it prints output as”**Undefined**” due to the hoisting property will place the variable “**i**” to the top of scope.And coming to 4th line which is “**console.log(i)**”,Here the variable “**i**” is reassigned with value “**5**” in previous line as”**var i = 5**”.So it prints output as “**5**”.