

Assignment

```
/* Q1. write a javaScript program to find the area of
a rectangle using a function. where users can enter
the value run time and see the result in the browser
*/

//A1:-

//declare variables
/*
let width = 4;
let length = 6;

//create a function
function Area(width, length){
    return (width * length);
}

//call
console.log("area of rectangle is" + " " + Area(width,
length) );
*/

/* Q2. write a program to print the below pattern of
stars.
*
**
***
****
***** */

//A2:-

/*
for (var i=1; i<=5; i++){
```

Assignment

```
    console.log("*".repeat(i));
  }
  */

//using repeat() method to construct and return
specified copies, concatenated together.
```

```
/* Q3. write a program to print the pyramid of stars.
*/

//A3.

/*
function pyramid(n){
  for(let i=1; i<=n; i++){
    let str = " ".repeat(n-i);
    let str2 = "*".repeat(i*2 -1)

    console.log(str + str2 + str);
  }
}
pyramid(5);
*/

/* Q4. write a program named displayname() which will
print your name. but this function will pass as a
callback to another function named main() pass your
name from main() and argument and print in the console
via displayname(). */

//A4.
```

Assignment

```
/*
function main(firstname, lastname){
    console.log("hello, my name is " + firstname + " " +
lastname);
}
//callback function
function displayname(){
    main("vinay", "choudhary");
}
displayname();
*/
```

/* Q5. what is the output of below code and why? */

//A5.

```
/*
let obj1 = { name: "Monday" };
let obj2 = { name: "Monday" };
console.log(obj1 == obj2);
*/
```

//Output = False

//1. obj1 and obj2 are in global frame.

/* Q6. What is the output of below code and why? */

//A6.

```
/*
function display(MyName)
{
let userName = MyName || "CK";
console.log(userName);
}
```

Assignment

```
return true
}
display('Yuvraj')
*/

//Output = yuvraj
//1. display() points console.log(username)(101)
//2. myname is assign to "yuvraj" and username is also
"yuvraj"(97,98)

/* Q7. Explain the output of the below code. */

//A7.

/*
for (let i = 0; i < 5; i++) {
  if (i == 2) {
    continue;
  }
  console.log(i);
  for (let i = 0; i <= 3; i++) {
    if (i == 2) {
      break;
    }
    console.log(`i:${i}`);
  }
}
*/
/*
Output = 0
        i:0
        i:1
        1
        i:0
```

Assignment

```
        i:1
        3
        i:0
        i:1
        4
        i:0
        i:1
    */

//1. it will print zero from the first for loop,
console.log(i).
//2. it will print i:0, i:1 from the second for loop,
console.log(`i:${i}`) and continue till if
condition(i==2)(SECOND IF CONDITION) and breaks the
loop and will move to the first loop again.
//3. same process for each until it reaches the
(i==2)(FIRST IF CONDITION) condition and will CONTINUE
THE LOOP AGAIN FROM START and after the execution of
continue statement the loop will print same for 3 and
4 and the execution will be terminated as i<5.

/* Q8. - Explain the output of the below code. */

//A8.

/*
let value = 0;
if (-2) {
var value = 10;
}
console.log(value);
*/
```

Assignment

```
//Output = cannot execute(SyntaxError)
//1. Identifier 'value'(153) has already been declared
at(151).
```

```
/* Q9. Explain the output of the below code. */
```

```
//A9.
```

```
/*
console.log(add);
var add = function (a, b) {
const results = a + b;
return results;
};
*/
```

```
//Output = Undefined
//1. at console.log add is undefined.
```

```
/* Q10. Explain the result of the below codes? */
```

```
//A10.
```

```
/*
let results = ((4 + 5) / 3) ** 2
console.log(results)
let result = NaN && null || '0' && 0
console.log(result)
*/
```

```
//console.log(0 === 0);
```

```
//Output = 9,0
```

Assignment

```
//1. Operator Precedence...  
//2. ** is is exponentiation  
//3. the output 9 is from the first two lines(183,184)  
i.e (4+5) is 9 and 9/3 is 3 and 3**2 is 9.  
//4. compare all one by one(184) i.e NaN == NaN is  
false and NaN === NaN is false, null == null is true  
and null === null is true, '0' == '0' is true and '0'  
=== '0' is true, 0 == 0 is true and 0 === 0 is true.  
//5. the &&(AND) operator returns is left side value  
and the ||(OR) operator returns the right side value.
```