

1. * In the following code snippets there are two variables initialized and printing console log statements

* The "num1" will be printed as undefined. It is declared with "var" which will make the initialization and declaration during the compilation stage and only undefined will be given as output ~~which~~ when we print the num1 value with console.log() before its assignment statement as the assignment will take place only during execution phase.

* ~~And~~

* And the let num2 = 20 is initialized after the console log statement so it gives Reference Error

2.

<!DOCTYPE html>

<html lang = "en">

<head>

<title> Equivalent Arrow Functions </title>

</head>

<body>

<script>

```
var multiply = (a, b) => {
    return a*b;
};
```

```
var toCelsius = (fahrenheit) => {
    return (5/9) * (fahrenheit - 32);
};
```

```
var greet = () => {
    return "welcome to javascript";
};
```

```
console.log (multiply (2,3))
```

```
console.log (to toCelsius (20))
```

```
console.log (greet) (greet ())
```

</script>

</body>

</html>

Output :

6
- 6.6667
welcome to javascript

3.

a) Sum of values in array :

~~let arr~~

< script >

let arr = [8, 5, 9, 6, 4]

let sum = 0;

arr.forEach (elements => { sum += elements })

console.log ("Sum of array is" + sum)

< /script >

< /body >

< /html >

Output :

Sum of Array is 32

b) Update the value with 10 times

<script>

```
let arr = [8, 5, 9, 6, 4];
```

```
updatedarr = arr.map (x => x*10)
```

```
console.log("Updated array is " + updatedarr)
```

</script>



Output :

Updated array is 80, 50, 90, 60, 40

4.

<script>

```
const name = "john";
```

```
const age = "29";
```

```
const phone = 12345;
```

```
var person = { name, age, phone };
```

</Script>

5.d) [4, 5, 6, 12, 3]

6 c) {foo : 1, bar : 2}