```
var x=5;
function getName(){
  console.log("React")
}
getName();
console.log(x)
```

>> React
5

```
getName();
console.log(x)
var x=5;
function getName(){
  console.log("React")
}
```

>> React
undefined

```
console.log(x)
Console.log(getName)
var x=7;
function getName(){
  Console.log("React")
}
```

>> undefined
[Function: getName]

" In the 1st phase of memory creation, js runs through the program and allocates memory to each variable and function. In case of variable it is undefined & in case of function it stores the whole code "

```
console.log(x)
Console.log(getNone)

function getNone(){
  Console.log("React")
  }
```

>> x is not defined

( reference x is not present and we are trying to initialize the value of x)

```
getNone();
Console.log(x);
Console.log(getNone)
 var x=7;
var getNone2 = function(){ }

var getNone=()=>{
   Console.log("React");
  }
```

Another way of function.

Arrow function

>> getNone is not a function

getNone and getNone2 behaves as a variable and it will allocate the memory with a placeholder undefined.