# Day 6

## Functions in JavaScript:

### What is a Function?

A function in JavaScript is a block of code designed to perform a specific task. It allows you to reuse code, make it organized, and reduce repetition.

### Syntax of a Function

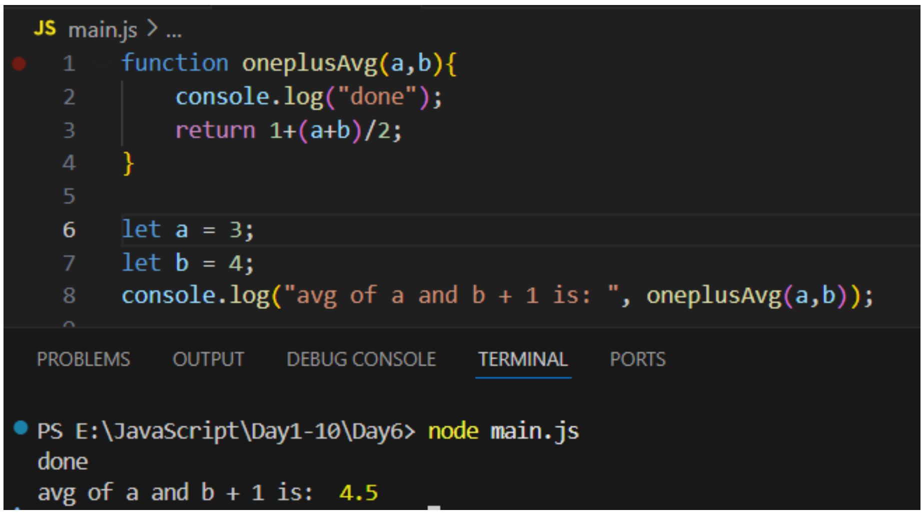
```
function functionName(parameters) {
  // code to execute
  return value;  // optional
}
```

| Part | Description |
|---|---|
| function | Keyword to declare a function |
| functionName | The name of the function |
| parameters | Input values the function can take (optional) |
| return | Sends a value back to the caller (optional) |

### Types of Functions

| Type | Example | Description |
|---|---|---|
| **Named Function** | function add(a,b){} | Regular declared function |
| **Anonymous Function** | function(a,b){} | No name, often used in variables or callbacks |
| **Function Expression** | let sum = function(a,b){} | Function stored in variable |
| **Arrow Function** | let sum = (a,b)=>a+b | Short and modern syntax |
| **IIFE** | (function(){})() | Immediately executed once |

Example: functions with parameters

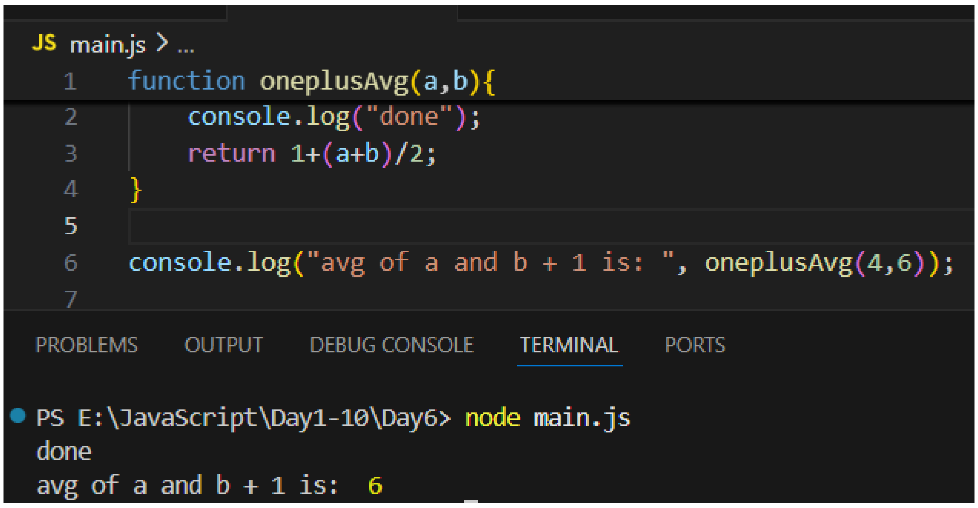


```javascript
function oneplusAvg(a,b){
    console.log("done");
    return 1+(a+b)/2;
}

let a = 3;
let b = 4;
console.log("avg of a and b + 1 is: ", oneplusAvg(a,b));
```

```
PS E:\JavaScript\Day1-10\Day6> node main.js
done
avg of a and b + 1 is:  4.5
```

Example: functions where we passed the exact values instead of variables.

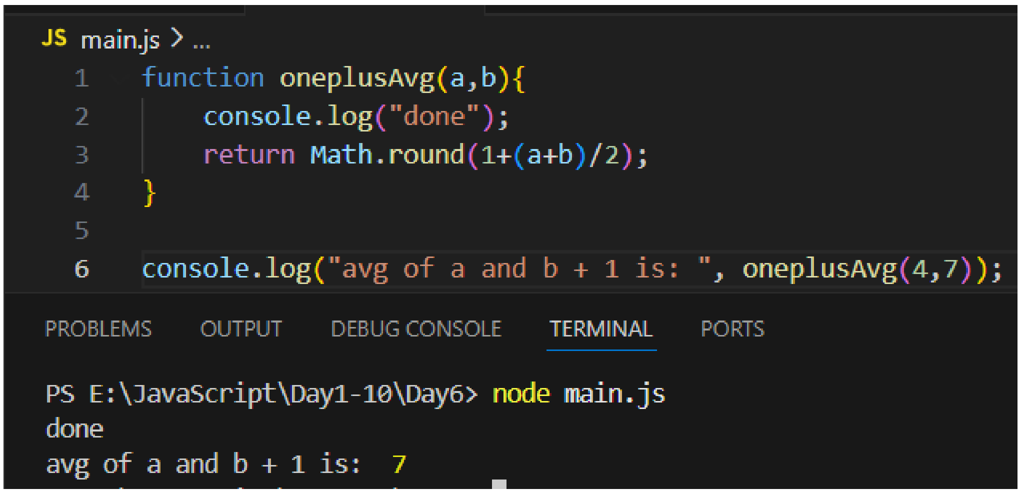


```javascript
function oneplusAvg(a,b){
    console.log("done");
    return 1+(a+b)/2;
}

console.log("avg of a and b + 1 is: ", oneplusAvg(4,6));
```

```
PS E:\JavaScript\Day1-10\Day6> node main.js
done
avg of a and b + 1 is:  6
```

Example: use functions to print the round off for the same above code, for that we will use Math.round() as shown below

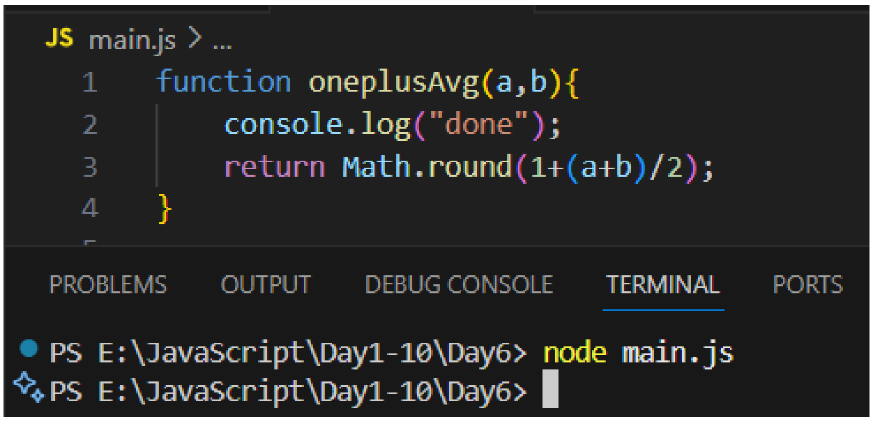


```javascript
function oneplusAvg(a,b){
    console.log("done");
    return Math.round(1+(a+b)/2);
}

console.log("avg of a and b + 1 is: ", oneplusAvg(4,7));
```

```
PS E:\JavaScript\Day1-10\Day6> node main.js
done
avg of a and b + 1 is:  7
```

Example: now in case we don't invoke the function, it will remain as it is, no output:

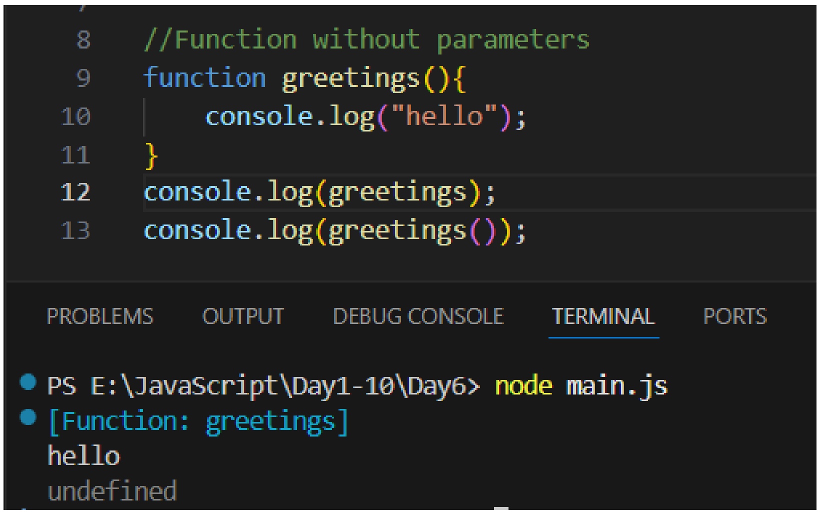


```
JS main.js > ...
function oneplusAvg(a,b){
    console.log("done");
    return Math.round(1+(a+b)/2);
}
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

```
PS E:\JavaScript\Day1-10\Day6> node main.js
PS E:\JavaScript\Day1-10\Day6>
```

Example: functions without parameters

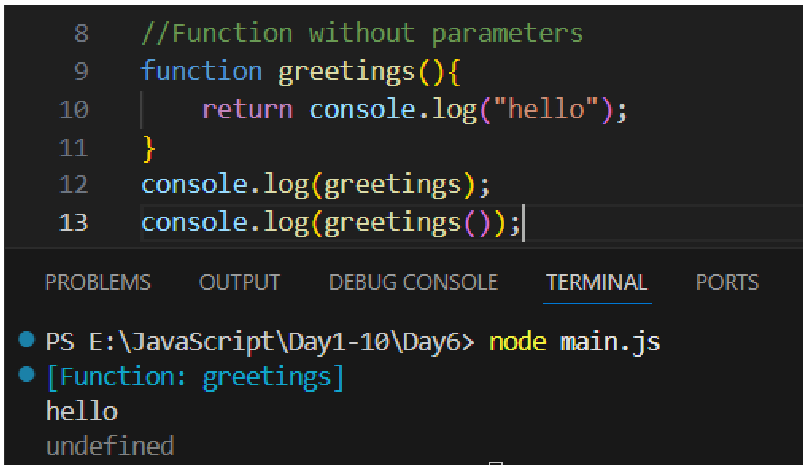


```
//Function without parameters
function greetings(){
    console.log("hello");
}
console.log(greetings);
console.log(greetings());
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

```
PS E:\JavaScript\Day1-10\Day6> node main.js
[Function: greetings]
hello
undefined
```

Example: functions without parameters but with return

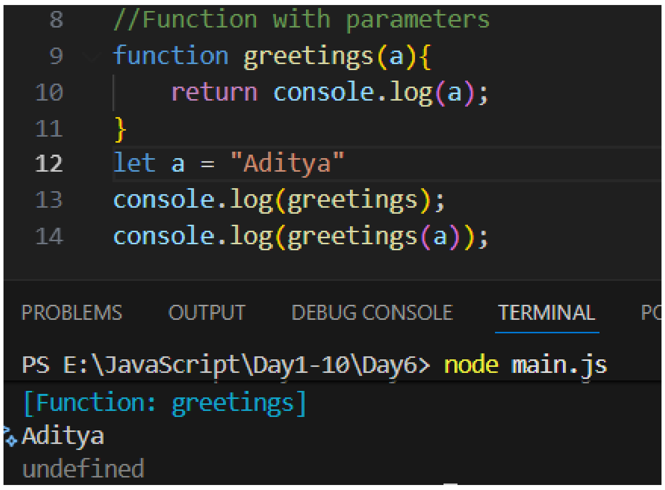


```
//Function without parameters
function greetings(){
    return console.log("hello");
}
console.log(greetings);
console.log(greetings());
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

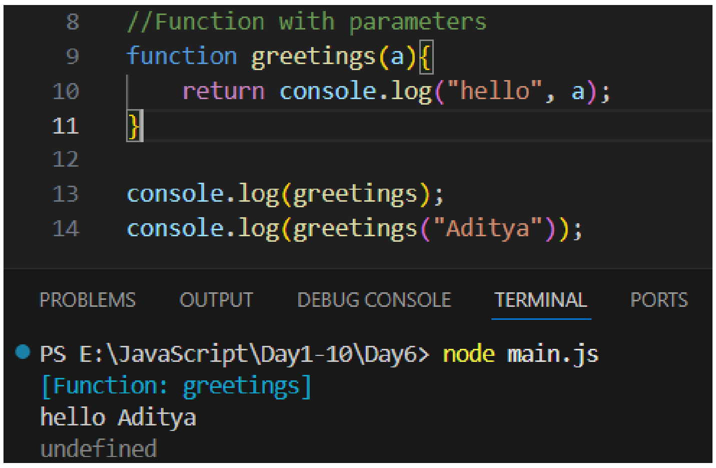
```
PS E:\JavaScript\Day1-10\Day6> node main.js
[Function: greetings]
hello
undefined
```

Example: same function but with parameters


```javascript
//Function with parameters
function greetings(a){
    return console.log(a);
}
let a = "Aditya"
console.log(greetings);
console.log(greetings(a));
```

```
PS E:\JavaScript\Day1-10\Day6> node main.js
[Function: greetings]
Aditya
undefined
```

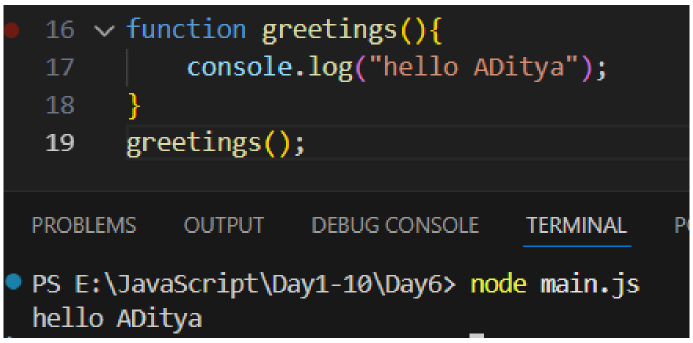
Example: same function with parameters


```javascript
//Function with parameters
function greetings(a){
    return console.log("hello", a);
}

console.log(greetings);
console.log(greetings("Aditya"));
```

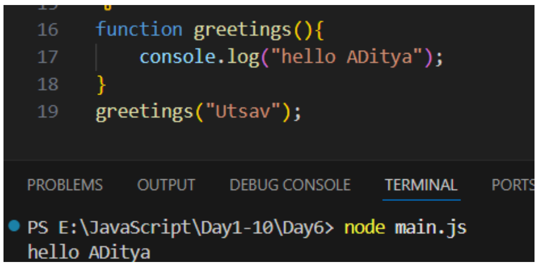
```
PS E:\JavaScript\Day1-10\Day6> node main.js
[Function: greetings]
hello Aditya
undefined
```

Example:


```javascript
function greetings(){
    console.log("hello ADitya");
}
greetings();
```

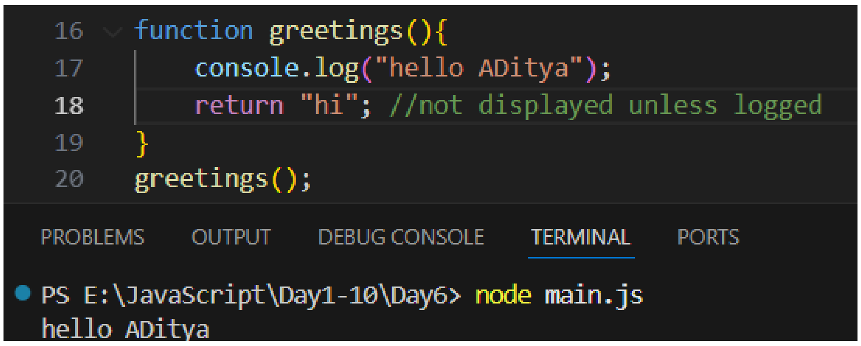
```
PS E:\JavaScript\Day1-10\Day6> node main.js
hello ADitya
```

Example:


```javascript
function greetings(){
    console.log("hello ADitya");
}
greetings("Utsav");
```

```
PS E:\JavaScript\Day1-10\Day6> node main.js
hello ADitya
```

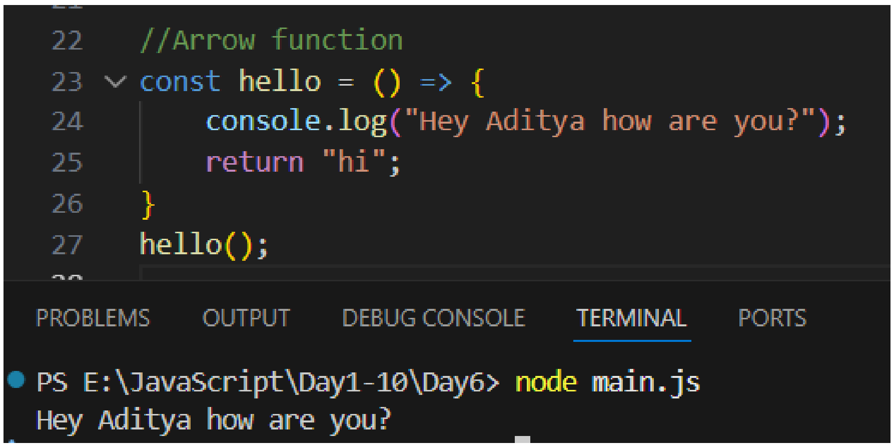
Example:



```
function greetings(){
    console.log("hello ADitya");
    return "hi"; //not displayed unless logged
}
greetings();
```

```
PS E:\JavaScript\Day1-10\Day6> node main.js
hello ADitya
```

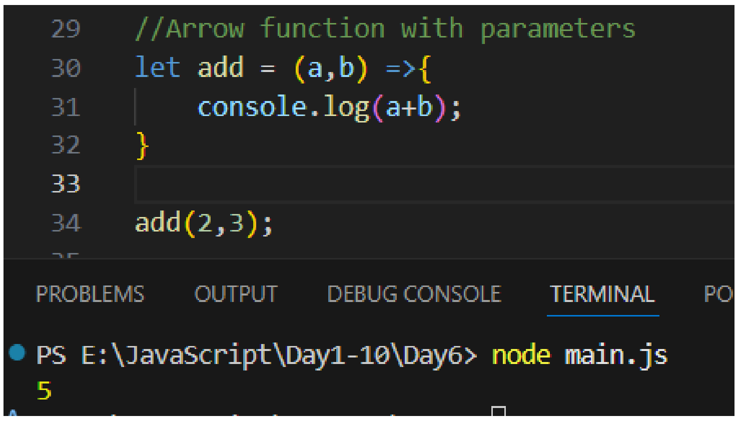
Example: arrow function in JS without any parameters



```
//Arrow function
const hello = () => {
    console.log("Hey Aditya how are you?");
    return "hi";
}
hello();
```

```
PS E:\JavaScript\Day1-10\Day6> node main.js
Hey Aditya how are you?
```

Example: arrow function with parameters

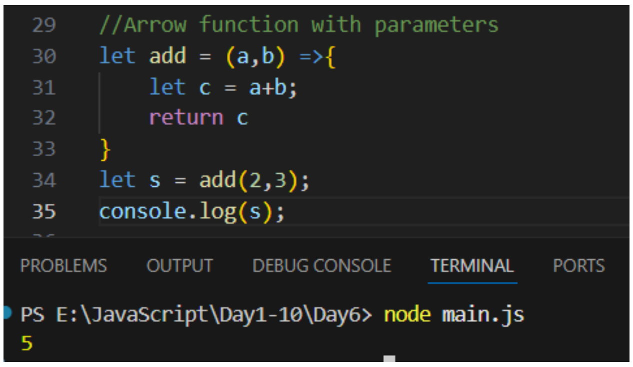


```
//Arrow function with parameters
let add = (a,b) =>{
    console.log(a+b);
}

add(2,3);
```

```
PS E:\JavaScript\Day1-10\Day6> node main.js
5
```

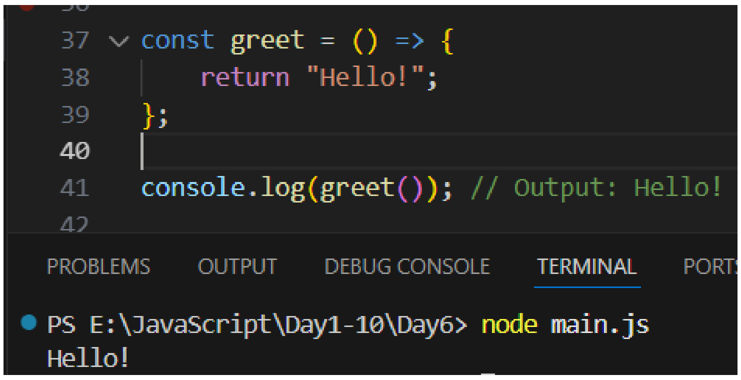
Example: arrow function with return and parameters



```
//Arrow function with parameters
let add = (a,b) =>{
    let c = a+b;
    return c
}
let s = add(2,3);
console.log(s);
```

```
PS E:\JavaScript\Day1-10\Day6> node main.js
5
```

Example: arrow function with return and no parameters



```
const greet = () => {
    return "Hello!";
};

console.log(greet()); // Output: Hello!
```

```
PS E:\JavaScript\Day1-10\Day6> node main.js
Hello!
```

--The End--