Lesson:

Function with a parameter (Single, and Multiple)

Topics Covered

1. Function with parameter.
2. Function with one parameter.
3. Function with default parameter.
4. Function with two parameters.
5. Array as an argument.

Before looking into the syntax and code let's understand what is a parameter and along with the parameter comes an argument.

A parameter is a variable in a function definition.

An argument is the actual value passed to the function when the function is called.

Inside a function, we can take one, two, multiple, or an unlimited number of parameters. In this lecture, we will be looking at how to take one parameter inside the function, and the rest will be seen in further lectures.

Let's look at an example where we will be writing a function that displays the message passed as a parameter onto the console.

```javascript
// Function with Parameter

// Function Declaration

function displayMessage(messageToBeDisplayed) {


    console.log(messageToBeDisplayed);
}
```

In the above example the "messageToBeDisplayed" is the parameter which is specified in the function declaration.

```javascript
// Function with Parameter

// Function Declaration

function displayMessage(messageToBeDisplayed) {
  console.log(messageToBeDisplayed);
}

// Calling in a function.

displayMessage("I am happy to learn Full Stack Web Development from
PW Skills");
displayMessage("I am enjoying the journey of javascript");
```

In the above examples the strings "I am happy to learn Full Stack Web Development from PW Skills" and "I am enjoying the journey of javascript" are passed into the function during the function call. These are the arguments.

When a function is called, the values passed in as arguments are assigned to the corresponding parameters in the function definition

Sometimes functions are written with default parameter values to avoid errors when the function is called without passing any arguments.

```javascript
// Function with Parameter

// Function Declaration

function displayMessage(
  messageToBeDisplayed = "I am a proud student of PW Skills"
) {
  console.log(messageToBeDisplayed);
}

// Calling in a function.

displayMessage("I am happy to learn Full Stack Web Development from PW
Skills");
// OUTPUT: I am happy to learn Full Stack Web Development from PW Skills

displayMessage();
// OUTPUT: I am a proud student of PW Skills
```

Here, the function "messageToBeDisplayed" has a default value of "I am a proud student of PW Skills".

When the argument "I am happy to learn Full Stack Web Development from PW Skills" is passed to the function and assigned to the parameter "messageToBeDisplayed". This results in the output "I am happy to learn Full Stack Web Development from PW Skills" being displayed in the console.

When the function is called without an argument or with undefined, the parameter "messageToBeDisplayed" is set to its default value "I am a proud student of PW Skills" and the output "I am a proud student of PW Skills" is displayed in the console.
To look at the function with two parameters, the best example would be of writing a function that gives us the sum of two numbers by taking the numbers as parameters.

```javascript
// Function with two parameters.

// Function Declaration

function sumOfTwoNumbers(num1, num2) {
  return num1 + num2;
}
```

The function "sumOfTwoNumbers" takes in two parameters, "num1" and "num2," and returns the sum of the two numbers.

```javascript
// Function with two parameters.

// Function Declaration

function sumOfTwoNumbers(num1, num2) {
  return num1 + num2;
}

// Calling a Function

let result = sumOfTwoNumbers(10, 40);
console.log(result);

// OUTPUT: 50
```

It is not necessary to always pass arguments as primitive values or variables containing primitive values like

```
// Function Declaration

function sumOfTwoNumbers(num1, num2) {
  return num1 + num2;
}

// Calling a Function

let number1 = 10;
let number2 = 40;

let result = sumOfTwoNumbers(number1, number2);
console.log(result);

// OUTPUT: 50
```

We can also pass the values as an array.

```
JavaScript
// function with two parameters.

// Function Declaration

function sumOfTwoNumbers([num1, num2]) {
  return num1 + num2;
}
```

The function takes an array as a parameter, this array should contain two values, and it returns the sum of the two numbers.

```
JavaScript
// Function with two parameters.

// Function Declaration

function sumOfTwoNumbers([num1, num2]) {
  return num1 + num2;
}

// Calling a Function

let numbers = [10, 40];

let result = sumOfTwoNumbers(numbers);
console.log(result);

// OUTPUT: 50
```

The function parameter is defined as an array, so the two values inside the argument array are accessed using their index, in this case, it is assumed that the first element of the array (index 0) is num1, and the second element of the array (index 1) is num2.

Ideally, developers pass arguments as primitive values, but it is important to note that we can pass an array as an argument too.