

Ada indikasi laporan sama dengan mahasiswa lain,  
untuk hari ini saya toleransi tapi tidak untuk lain hari

# Laporan Praktikum Pemrograman 1

## Modul III – Function pada JavaScript

NELLYANA SAFITRI/22103001011

Dosen : Achmad Arif Munaji, ST., M.Kom

Tanggal praktikum: 10 November 2022

nellyana2@gmail.com

Teknik Komputer

Institut Teknologi dan Sains Nahdlatul Ulama Kalimantan

5+5+7+5+25+5+3=

55

**5 Abstrak**— Pada praktikum ketiga ini mengenai Function pada javascript. Function diartikan sebagai fungsi. Untuk Pengertian Function atau fungsi adalah sebuah blok kode yang digunakan untuk membungkus suatu proses dengan tujuan agar penulisan kode atau proses yang sama tidak ditulis secara berulang kali. Function bisa memiliki parameter bisa juga tidak memiliki parameter, parameter ini digunakan untuk menentukan proses yang ingin dijalankan, sesuai dengan kebutuhan kode program kita. untuk membuat function, penulisannya harus diawali oleh syntax “function” dan kemudian dilanjutkan dengan nama function yang mau di buat. dan isi dari function di tuliskan di dalam tanda kurung kurawal pembuka “{” dan tanda kurung kurawal penutup “}”.

**Kata kunci**— bahasa pemrograman JavaScript

**5**

### I. PENDAHULUAN

Pada pembelajaran praktikum pemrograman kali ini, diberi materi mengenai function pada javascript yang diakses melalui website freecodecamp. Pada praktikum ini, sama dengan praktikum sebelumnya yaitu memecahkan masalah coding pada function di javascript. Soal yang diberikan sebanyak 8 soal, yaitu berkaitan dengan function. soal tersebut meliputi write reusable javascript with function, passing values to functions with arguments, return a value from a function with return, global scope and functions, local scope and functions, global vs local scope in functions, understanding undefined value returned from a function dan terakhir assingment with a returned value.

**7**

### II. TINJAUAN PUSTAKA

#### A. Function atau fungsi di JavaScript

Function atau fungsi JavaScript adalah blok kode yang digunakan untuk membungkus suatu proses dengan tujuan untuk tidak menulis kode atau proses yang sama berulang kali. dengan membuat function di javascript lebih memudahkan kita ketika ingin mengembangkan sebuah proses, apabila ada perubahan pada proses maka kita tidak perlu mengubah satu-per-satu.

Selain itu function adalah sub-program yang bisa digunakan kembali baik di dalam program itu sendiri, maupun di program yang lain. Fungsi di dalam Javascript adalah sebuah objek. Karena memiliki properti dan juga *method*. Bagi pemula konsep ini cukup membingungkan. Apalagi yang belum mengenal konsep [OOP](#).

Ada 4 cara yang bisa kita lakukan untuk membuat fungsi di Javascript:

1. Menggunakan cara biasa;
2. Menggunakan ekspresi;
3. Menggunakan tanda panah (=>);
4. dan menggunakan *Constructor*.

**5**

### III. LANGKAH PRAKTIKUM

Pertama-pertama pada praktikum ini diawali dengan soal **freetest** mengenai function. Setelah selesai mengerjakan soal **freetest** tersebut, lalu dilanjutkan dengan membuka google chrome tentang website freecodecamp. Setelah itu mulai menjawab soal-soal pada website freecodecamp. Soal tersebut sebanyak 8 soal dan setiap soal terdapat solusi (bantuan).

## 25 IV. HASIL DAN ANALISIS

sebelum dimulainya pratikum pertama-tama dimulai free test tentang function, sebagai berikut :

Tulis kode program sederhana menggunakan FUNCTION untuk melakukan proses Aritmatika 2 variabel atau nilai dan panggil Function tersebut.

setelah selesai dilanjutkan dengan mengerjakan kuis atau perintah pada website FreeCodeCamp tentang function. berikut rincinya:

### 1. Write Reusable JavaScript with Functions

Dalam JavaScript, kita dapat membagi kode kita menjadi bagian-bagian yang dapat digunakan kembali yang disebut **functions**.

```
function reusableFunction() {  
  console.log("Hi World");  
}
```

```
reusableFunction();
```

### 2. Passing Values to Functions with Arguments

Parameter adalah variabel yang bertindak sebagai placeholder untuk nilai yang akan dimasukkan ke fungsi saat dipanggil. Ketika suatu fungsi didefinisikan, biasanya didefinisikan bersama dengan satu atau lebih parameter. Nilai aktual yang dimasukkan (atau "dilewati") ke dalam fungsi saat dipanggil dikenal sebagai argumen.

```
function functionWithArgs(one, two) {  
  console.log(one + two);  
}  
functionWithArgs(7, 3);
```

### 3. Return a Value from a Function with Return

Kita dapat meneruskan nilai ke dalam suatu fungsi dengan argumen. Anda dapat menggunakan **returnpernyataan** untuk mengirim nilai kembali dari suatu fungsi.

```
function timesFive(num) {  
  return num * 5;  
}
```

### 4. Global Scope and Functions

Dalam JavaScript, ruang lingkup mengacu pada visibilitas variabel. Variabel yang didefinisikan di luar blok fungsi memiliki cakupan Global. Ini berarti, mereka dapat dilihat di mana-mana dalam kode JavaScript Anda.

```
const myGlobal = 10;  
  
function fun1() {
```

```
  oopsGlobal = 5;  
}
```

```
function fun2() {  
  var output = "";  
  if (typeof myGlobal !== "undefined") {  
    output += "myGlobal: " + myGlobal;  
  }  
  if (typeof oopsGlobal !== "undefined") {  
    output += "oopsGlobal: " + oopsGlobal;  
  }  
  console.log(output);  
}
```

### 5. Local Scope and Functions

Variabel yang dideklarasikan dalam suatu fungsi, serta parameter fungsi, memiliki cakupan lokal. Itu berarti mereka hanya terlihat di dalam fungsi itu.

```
function myLocalScope() {  
  
  var myVar;  
  console.log('inside myLocalScope', myVar);  
}  
myLocalScope();  
  
console.log('outside myLocalScope', myVar);
```

### 6. Global vs. Local Scope in Functions

Dimungkinkan untuk memiliki variabel lokal dan global dengan nama yang sama. Ketika Anda melakukan ini, variabel lokal lebih diutamakan daripada variabel global.

```
var outerWear = "T-Shirt";  
  
function myOutfit() {  
  var outerWear = "sweater";  
  return outerWear;  
}  
  
myOutfit();
```

### 7. Understanding Undefined Value returned from a Function

Suatu fungsi dapat menyertakan returnpernyataan tetapi tidak harus. Jika fungsi tidak memiliki returnpernyataan, saat Anda memanggilnya, fungsi memproses kode dalam tetapi nilai yang dikembalikan adalah undefined.

```
// Setup  
let sum = 3;
```

```
function addFive() {  
  sum += 5;  
}  
addFive();
```

## 5 V. KESIMPULAN

### 8. Assignment with a Returned Value

Jika Anda ingat dari diskusi kami tentang Menyimpan Nilai dengan Operator Penugasan, semua yang ada di sebelah kanan tanda sama dengan diselesaikan sebelum nilai ditetapkan. Ini berarti kita dapat mengambil nilai balik dari suatu fungsi dan menentukannya ke variabel.

```
// Setup
let processed = 0;

function processArg(num) {
  return (num + 3) / 5;
}

// Only change code below this line
processed = processArg(7);
```

#### - Analisis

pada praktikum ini untuk function pada JavaScript saya sedikit mulai memahami dikarenakan sebelumnya sudah mempelajari materi function pada pertemuan pemrograman dan dilanjutkan diajarkan oleh teman-teman Sehingga membantu saya untuk mengerjakan poin-poin pada freeCodeCamp.

Fungsi adalah satu blok (kumpulan) perintah coding (statements) yang mempunyai tujuan atau fungsi tertentu. Fungsi dipanggil atau dieksekusi apabila dipanggil oleh coding yang lain.

Selain itu Function pada JavaScript merupakan salah satu perkakas yang sangat powerful, paling fundamental, dan paling penting. Bisa bilang semua hal yang bisa dilakukan oleh JavaScript pasti didasari oleh function dan sifat-sifatnya. Jika Anda buka source code React, Vue, Redux, dan library JS lainnya, Anda bisa mengobservasi bahwa library-library tersebut 'hanyalah' kumpulan dari function-function.

## 3 DAFTAR PUSTAKA

[1] <https://www.petanikode.com/javascript-fungsi/>

[2] <https://www.slideshare.net/arryyoo/fungsi-function>

[3] <https://bahasaweb.com/membuat-function-di-javascript/>

[4] <https://medium.com/bahasa-ai/kenali-function-di-javascript-5427cd9128b2>