

Javascript Lanjutan



Ellya Rosanti
119140082
Pengembangan Aplikasi Mobile RB

Closure



Closure adalah sebuah inner function yang mempunyai hak akses ke dalam sebuah variabel di dalam function. Inner function ini juga memiliki akses di luar function. Closure dapat mengakses variabel dalam cakupan yang berbeda: 1. Variabel yang terdapat di dalam function sendiri. 2. Variabel yang terdapat di luar function. 3. Variabel yang terdapat dalam parent function.

https://www.w3schools.com/js/is_function_closures.asp

<https://medium.com/@rivayudha/closures-apaan-tuh-alasan-kamu-benci-javascript-8484a3437a40>

```
<!DOCTYPE html>
<html>
  <body>
    <h2>Closure</h2>
    <p>mencoba closure:</p>

    <p id="coba"></p>

    <script>
      let i = 4;
      closure();

      function closure() {
        document.getElementById("coba").innerHTML = i * i;
      }
    </script>
  </body>
</html>
```

Immediately Invoked Function Expression



Immediately Invoked Function Expression adalah sebuah fungsi javascript yang akan bekerja setelah didefinisikan. IIFE dapat membuat deklarasi fungsi lebih baik. IIFE dapat membuat variabel global terisolasi.

<https://developer.mozilla.org/en-US/docs/Glossary/IIFE>

```
const coba = async (url) => { /* implementation */};

(async () => {
  const stream = await coba ('https:developer.mozilla.org/en-US/docs/Glossary/IIFE');
  for await (const chunk of stream) {
  }
}) ();
```

First-class function

First-class function merupakan sebuah fungsi yang dapat diteruskan ke dalam fungsi lain.

Seperti contoh disamping, fungsi sayHello dapat diteruskan ke dalam fungsi greetin.

https://developer.mozilla.org/en-US/docs/Glossary/First-class_Function

```
function sayHello() {  
    return "Hello, ";  
}  
function greeting(helloMessage, name) {  
    console.log(helloMessage() + name);  
}  
// Pass sayHello as an argument to greeting function  
greeting(sayHello, "JavaScript!")  
// Hello, JavaScript!
```

Higher-order function

High-order function merupakan fungsi yang dapat bekerja pada fungsi lain. Fungsi tingkat tinggi dapat memungkinkan untuk membuat fungsi baru.

<https://eloquentjavascript.ne/>

```
function greaterThan(n){  
  return m => m > n;  
}  
let greaterThan10 = greaterThan(10);  
console.log(greaterThan10(11));  
// -> true
```

Execution Context



Execution Context merupakan variabel, objek, dan fungsi yang dapat di akses oleh kode javascript dalam waktu tertentu.

1. Global execution context
2. Funcional execution context
3. Eval

<https://www.javascripttutorial.net/javascript-execution-context/>

```
let x = 10;

function timeTen(a){
    return a * 10;
}

let y = timeTen(x);

console.log(y); //100
```

Execution Stack



Execution stack merupakan suatu fungsi yang memanggil fungsi lain. Ketika fungsi dipanggil, maka fungsi tersebut akan memanggil lagi fungsi yang ada sehingga fungsi yang terakhir dipanggil akan di eksekusi terlebih dahulu.

<https://www.javascripttutorial.net/javascript-call-stack/>

```
function add(a,b) {  
    return a+b;  
}  
  
function average(a,b){  
    return add(a,b)/2;  
}  
  
let x = average(10,20);
```

Event Loop

Event loop merupakan suatu fungsi yang dapat dieksekusi ketika tumpukan fungsi kosong.

<https://www.geeksforgeeks.org/what-is-an-event-loop-in-javascript/>

Callbacks

Callbacks merupakan sebuah fungsi yang diberikan ke fungsi lain sebagai argumen untuk menunggu dieksekusi.

<https://www.javascripttutorial.net/javascript-callback/>

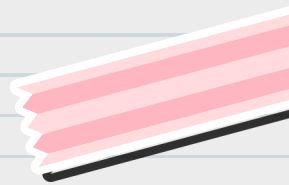
° Promises dan Async/Await



↻

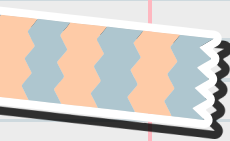
Promise adalah sebuah metode dalam javascript untuk mengembalikan sebuah nilai di masa mendatang. Promise terdiri dari pending, fulfilled, dan rejected.

https://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Statements/async_function



✿

Async merupakan sebuah fungsi yang menggunakan kata async dan kata kunci menunggu lainnya yang digunakan bersama dengan promise untuk membuat pengembalian nilai lebih baik.



```
async function name([param[, param[, ...param]]]) {  
  statements  
}
```





Link Github :

[EllyarRosanti/Tugas2-PAM \(github.com\)](https://github.com/EllyarRosanti/Tugas2-PAM)

Terimakasih

