

## TUGAS PENDAHULUAN

### MODUL 4

#### 1. KodePos.js

Inputan code

```
const readline = require('readline');

class KodePos {
  constructor() {
    this.dataKodePos = {
      Batununggal: 40266,
      Kujangsari: 40287,
      Mengger: 40267,
      Wates: 40256,
      Cijaura: 40287,
      Jatisari: 40286,
      Margasari: 40286,
      Sekejati: 40286,
      Kebonwaru: 40272,
      Maleer: 40274,
      Samoja: 40273,
    };
  }

  getKodePos(kelurahan) {
    const formattedKelurahan = kelurahan.charAt(0).toUpperCase() +
      kelurahan.toLowerCase().slice(1);
    return kodePos ? kodePos : "Kode pos tidak ditemukan";
  }

  listKelurahan() {
    console.log("Daftar Kelurahan");
    for(const kelurahan in this.dataKodePos) {
      console.log(`- ${kelurahan}`);
    }
  }
}

const kodePos = new KodePos();

const rl = readline.createInterface({
  input: process.stdin,
  output: process.stdout,
});

kodePos.listKelurahan();

rl.question('\nMasukkan nama kelurahan untuk mencari kode pos: ', (input) => {
  console.log(`Kode Pos ${input}:`, kodePos.getKodePos(input));
  rl.close();
});
```

Di dalam Class KodePos terdapat data kelurahan dan kode pos dalam bentuk objek. Memiliki metode getKodePos untuk mencari kode pos berdasarkan nama kelurahan (case-insensitive). Memiliki metode listKelurahan untuk mencetak daftar kelurahan yang tersedia. Menggunakan modul readline untuk membaca input pengguna dari terminal. Menampilkan daftar kelurahan, lalu meminta pengguna memasukkan nama kelurahan. Jika kelurahan ditemukan,

Jadi output code diatas sebagai berikut:

```

PS D:\Kuliah\KPL_MUHAMMAD MAHRUS ALI_2311104006_SE-07-01> node "d:\Kuliah\KPL_MUHAMMAD MAHRUS
ALI_2311104006_SE-07-01\04_00P\TP_00P_2311104006\KodePos.js"
Daftar Kelurahan
- Batununggal
- Kujangsari
- Mengger
- Wates
- Cijaurea
- Jatisari
- Margasari
- Sekejati
- Kebonwaru
- Maleer
- Samoja

Masukkan nama kelurahan untuk mencari kode pos: wates
Kode Pos wates: 40256
PS D:\Kuliah\KPL_MUHAMMAD MAHRUS ALI_2311104006_SE-07-01>

```

## 2. DoorMachine.js

Inputan code:

```

class DoorMachine {
  constructor() {
    this.state = 'Terkunci';
    this.showState();
  }

  lock() {
    this.state = 'Terkunci';
    this.showState();
  }

  unlock() {
    this.state = 'Terbuka';
    this.showState();
  }

  showState() {
    if (this.state === 'Terkunci') {
      console.log('Pintu terkunci');
    } else if (this.state === 'Terbuka') {
      console.log('Pintu tidak terkunci');
    }
  }
}

const door = new DoorMachine();

door.unlock();
door.lock();

```

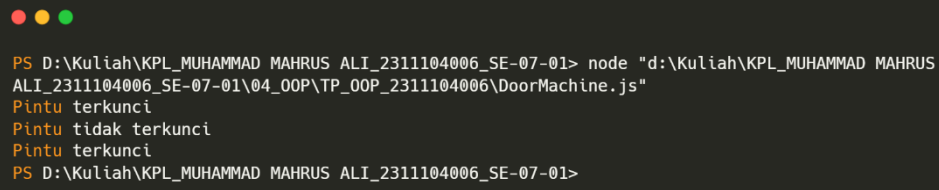
Class DoorMachine.js Merepresentasikan mesin pintu dengan dua state: Terkunci dan Terbuka. Saat objek DoorMachine dibuat, state default-nya adalah "Terkunci" dan langsung mencetak "Pintu terkunci". Method yang tersedia :

lock() → Mengubah state menjadi "Terkunci" dan mencetak "Pintu terkunci".

unlock() → Mengubah state menjadi "Terbuka" dan mencetak "Pintu tidak terkunci".

showState() → Menampilkan status pintu berdasarkan state saat ini.

Outputnya:

A terminal window with a dark background and three colored window control buttons (red, yellow, green) in the top-left corner. The terminal shows a command being executed in a PowerShell prompt, followed by three lines of output and the prompt returning.

```
PS D:\Kuliah\KPL_MUHAMMAD MAHRUS ALI_2311104006_SE-07-01> node "d:\Kuliah\KPL_MUHAMMAD MAHRUS
ALI_2311104006_SE-07-01\04_OOP\TP_OOP_2311104006\DoorMachine.js"
Pintu terkunci
Pintu tidak terkunci
Pintu terkunci
PS D:\Kuliah\KPL_MUHAMMAD MAHRUS ALI_2311104006_SE-07-01>
```