

Link GitHub repository

https://github.com/IsabelleArd/KPL_ISABELLE-PUTRI-ARDINI_2311104030_SE0701/tree/master/04_OOP

Class KodePos.js

```
// Class dengan teknik Table-Driven
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) {
    return this.dataKodePos[kelurahan] || "Kode pos tidak ditemukan";
  }
}
```

```
const kodePos = new KodePos();
console.log("Kode pos Batununggal:", kodePos.getKodePos("Batununggal"));
console.log("Kode pos Wates:", kodePos.getKodePos("Wates"));
```

Ouput

```
lenovo@Leniv MINGW64 /d/KPL_ISABELLE PUTRI ARDINI_2311104030_SE0701/04_OOP (master)
● $ node tp_KodePos.js
Kode pos Batununggal: 40266
Kode pos Wates: 40256
```

Penjelasan

- Class KodePos menggunakan teknik table-driven, yaitu mengambil hasil berdasarkan input.

- Table-Driven sendiri digunakan untuk mengelola inputan data secara efisien dengan cara lookup langsung dari tabel.
- 'dataKodePos' digunakan untuk menyimpan pasangan nama kelurahan dan kode pos.
- 'getKodePos' digunakan untuk memanggil kode pos berdasarkan input 'kelurahan'. Kode pos dikembalikan apabila kelurahan ditemukan dalam objek. Jika tidak ditemukan maka akan mengembalikan "Kode pos tidak ditemukan".

Class DoorMachine.js

```
// Class dengan teknik State-Based Construction
class DoorMachine {
  constructor() {
    this.state = "Terkunci";
  }

  lock() {
    this.state = "Terkunci";
    console.log("Pintu terkunci");
  }

  unlock() {
    this.state = "Terbuka";
    console.log("Pintu tidak terkunci");
  }
}
```

```
const pintu = new DoorMachine();
pintu.lock(); // Output: Pintu terkunci
pintu.unlock(); // Output: Pintu tidak terkunci
```

Output

```
lenovo@Leniv MINGW64 /d/KPL_ISABELLE PUTRI ARDINI_2311104030_SE0701/04_OOP (master)
● $ node DoorMachine_tp.js
Pintu terkunci
Pintu tidak terkunci
```

Penjelasan

- Class DoorMachine menggunakan teknik State_Based Construction, yang berarti objek berubah berdasarkan aksi.

- State-Based Construction sendiri digunakan untuk mengelola perubahan status suatu objek dengan aksi tertentu.
- Pada program tersebut, state awal DoorMachine adalah “Terkunci”, state tersebut dapat diubah menggunakan method ‘lock()’ untuk mengubah status awal menjadi “Pintu terkunci” dan ‘unlock()’ untuk mengubah status awal menjadi “Pintu tidak terkunci”.