

Nama : Elvaretta Anantya Velya

NIM : 2211104074

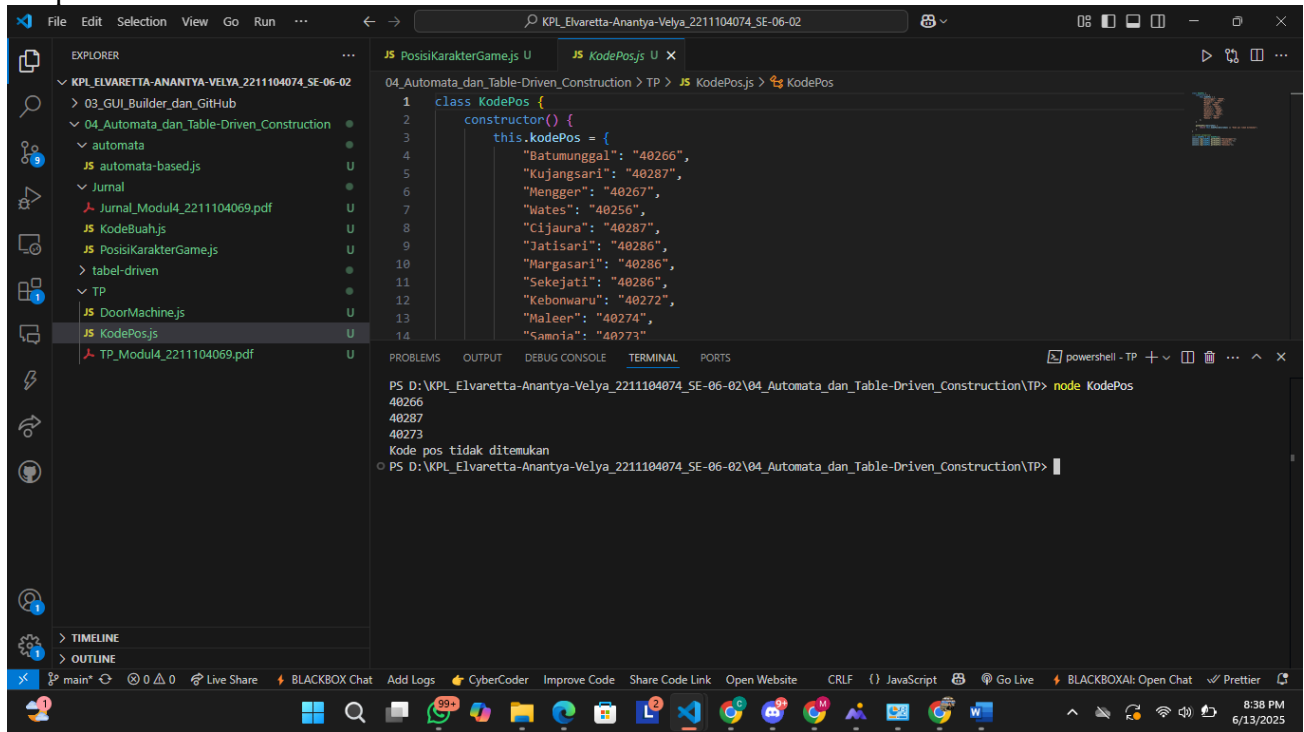
Kelas : SE06-02

Kode Pos

Source Code :

```
1  class KodePos {
2      constructor() {
3          this.kodePos = {
4              "Batumunggal": "40266",
5              "Kujangsari": "40287",
6              "Mengger": "40267",
7              "Wates": "40256",
8              "Cijaura": "40287",
9              "Jatisari": "40286",
10             "Margasari": "40286",
11             "Sekejati": "40286",
12             "Kebonwaru": "40272",
13             "Maleer": "40274",
14             "Samoja": "40273"
15         };
16     }
17
18     getKodePos(kelurahan) {
19         return this.kodePos[kelurahan] || "Kode pos tidak ditemukan";
20     }
21 }
22
23 // Contoh penggunaan
24 const kodePos = new KodePos();
25 console.log(kodePos.getKodePos("Batumunggal"));
26 console.log(kodePos.getKodePos("Kujangsari"));
27 console.log(kodePos.getKodePos("Samoja"));
28 console.log(kodePos.getKodePos("Banyumas"));
```

Output :



The screenshot shows a Visual Studio Code editor window with the following details:

- Explorer Panel:** Displays a file tree for the project 'KPL_ELVARETTA-ANANTYA-VELYA_2211104074_SE-06-02'. The file 'KodePos.js' is selected under the 'TP' folder.
- Editor Panel:** Shows the content of 'KodePos.js' with the following code:

```
1 class KodePos {
2   constructor() {
3     this.kodePos = {
4       "Batumunggal": "40266",
5       "Kujangsari": "40287",
6       "Menger": "40267",
7       "Wates": "40256",
8       "Cijaura": "40287",
9       "Jatisari": "40286",
10      "Margasari": "40286",
11      "Sekejati": "40286",
12      "Kebonwaru": "40272",
13      "Maleen": "40274",
14      "Samoia": "40273"
15    }
16  }
17 }
```
- Terminal Panel:** Shows the output of running the command 'node KodePos'. The output is:

```
PS D:\KPL_Elvareta-Anantya-Velya_2211104074_SE-06-02\04_Automata_dan_Table-Driven_Construction\TP> node KodePos
40266
40287
40273
Kode pos tidak ditemukan
PS D:\KPL_Elvareta-Anantya-Velya_2211104074_SE-06-02\04_Automata_dan_Table-Driven_Construction\TP>
```

Penjelasan :

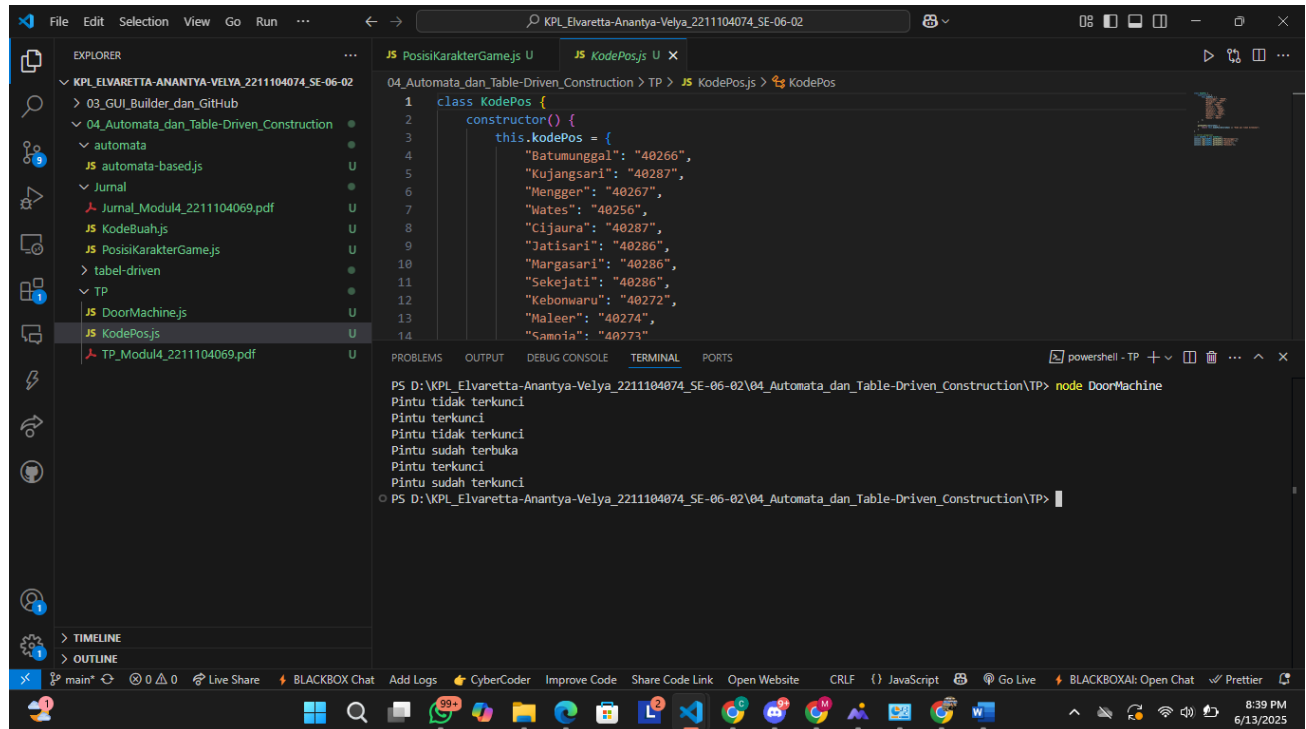
Program ini menggunakan 'KodePos' sebagai objek untuk menyimpan data dari kode pos. Method 'getKodePos' akan digunakan untuk mencari kode pos berdasarkan nama daerah yang diberikan. Apabila ditemukan kode pos akan dikembalikan, dan jika tidak ditemukan, maka akan mengembalikan pesan "Kode pos tidak ditemukan".

DoorMachine

Source Code :

```
1  class DoorMachine {
2      constructor() {
3          this.state = "Terkunci"; // State awal pintu adalah "Terkunci"
4      }
5
6      // Method untuk mengubah state pintu
7      changeState(action) {
8          if (action === "KunciPintu") {
9              if (this.state === "Terbuka") {
10                 this.state = "Terkunci";
11                 console.log("Pintu terkunci");
12             } else {
13                 console.log("Pintu sudah terkunci");
14             }
15         } else if (action === "BukaPintu") {
16             if (this.state === "Terkunci") {
17                 this.state = "Terbuka";
18                 console.log("Pintu tidak terkunci");
19             } else {
20                 console.log("Pintu sudah terbuka");
21             }
22         } else {
23             console.log("Aksi tidak valid");
24         }
25     }
26 }
27
28 // Contoh penggunaan
29 const door = new DoorMachine();
30
31 door.changeState("BukaPintu");
32 door.changeState("KunciPintu");
33 door.changeState("BukaPintu");
34 door.changeState("BukaPintu");
35 door.changeState("KunciPintu");
36 door.changeState("KunciPintu");
```

Output :



The screenshot shows the Visual Studio Code interface. The Explorer panel on the left shows a project structure with files like 'KodePos.js' and 'DoorMachine.js'. The main editor displays the code for 'KodePos.js', which defines a class with a constructor and a static property 'kodePos'. The terminal at the bottom shows the output of running 'node DoorMachine.js', which prints the state of the door (locked or unlocked) based on the actions performed.

```
class KodePos {
  constructor() {
    this.kodePos = {
      "Batumunggal": "40266",
      "Kujangsari": "40287",
      "Menger": "40267",
      "Mates": "40256",
      "Cijaura": "40287",
      "Jatisari": "40286",
      "Margasari": "40286",
      "Sekejati": "40286",
      "Kebonwaru": "40272",
      "Maleer": "40274",
      "Samnia": "40273"
    }
  }
}
```

```
PS D:\KPL_Elvareta-Anantya-Velya_2211104074_SE-06-02\04_Automata_dan_Table-Driven_Construction\TP> node DoorMachine
Pintu tidak terkunci
Pintu terkunci
Pintu tidak terkunci
Pintu sudah terbuka
Pintu terkunci
Pintu sudah terkunci
PS D:\KPL_Elvareta-Anantya-Velya_2211104074_SE-06-02\04_Automata_dan_Table-Driven_Construction\TP>
```

Penjelasan :

State awal pada pintu adalah “Terkunci”, yang mana diatur dalam constructor class ‘DoorMachine’. Method ‘changeState’ menerima parameter action yang bisa berupa "KunciPintu" atau "BukaPintu". Jika action adalah "KunciPintu", program akan memeriksa apakah state saat ini adalah "Terbuka". Jika ya, state akan diubah menjadi "Terkunci" dan pesan "Pintu terkunci" akan ditampilkan. Jika state sudah "Terkunci", program akan menampilkan pesan "Pintu sudah terkunci". Jika action adalah "BukaPintu", program akan memeriksa apakah state saat ini adalah "Terkunci". Jika ya, state akan diubah menjadi "Terbuka" dan pesan "Pintu tidak terkunci" akan ditampilkan. Jika state sudah "Terbuka", program akan menampilkan pesan "Pintu sudah terbuka".