

Kelvin Ferdinan_2311104009

https://github.com/mispipin/KPL_Kelvin-Ferdinan_2311104009_s1se0701

```
14_Clean_Code > JS Refactoring.js > KodePos > constructor
1 // Import module untuk input dari terminal
2 const readline = require("readline");
3
4 /**
5  * Class KodePos
6  * Menyediakan data dan fungsi untuk mengambil kode pos berdasarkan nama kelurahan.
7  */
8 class KodePos {
9     // Properti private untuk menyimpan data kelurahan dan kode pos
10    constructor() {
11        this._kelurahanKodePosMap = Object.fromEntries([
12            ["Batununggal", "40266"],
13            ["Kujangsari", "40287"],
14            ["Mengger", "40267"],
15            ["Wates", "40256"],
16            ["Cijaura", "40287"],
17            ["Jatisari", "40286"],
18            ["Margasari", "40286"],
19            ["Sekejati", "40286"],
20            ["Kebonwaru", "40272"],
21            ["Maleer", "40274"],
22            ["Samoja", "40273"]
23        ]);
24    }
25
26    /**
27     * Mengambil kode pos berdasarkan nama kelurahan.
28     * @param {string} kelurahan - Nama kelurahan
29     * @returns {string} Kode pos atau pesan error
30     */
31    getPostalCodeByKelurahan(kelurahan) {
32        if (kelurahan in this._kelurahanKodePosMap) {
33            return this._kelurahanKodePosMap[kelurahan];
34        }
35
36        return "Kode Pos tidak ditemukan";
37    }
38 }
39
40 // Inisialisasi antarmuka pembaca input dari terminal
41 const inputInterface = readline.createInterface({
42     input: process.stdin,
43     output: process.stdout
44 });
45
46 const postalCodeService = new KodePos();
47
48 /**
9_SE0701 9 main* 0 0 Git Graph
```

14_Clean_Code > Refactoring.js > KodePos > constructor

```
8   class KodePos {
9
10     constructor() {
11       this._kelurahanKodePosMap = [
12         ["Samoja", "40273"],
13         ["Jatisari", "40286"],
14         ["Margasari", "40286"],
15         ["Sekejati", "40286"],
16         ["Kebonwaru", "40272"],
17         ["Maleer", "40274"],
18         ["Samoja", "40273"]
19       ];
20     }
21
22     /**
23      * Mengambil kode pos berdasarkan nama kelurahan.
24      * @param {string} kelurahan - Nama kelurahan
25      * @returns {string} Kode pos atau pesan error
26      */
27     getPostalCodeByKelurahan(kelurahan) {
28       if (kelurahan in this._kelurahanKodePosMap) {
29         return this._kelurahanKodePosMap[kelurahan];
30       }
31
32       return "Kode Pos tidak ditemukan";
33     }
34   }
35
36   // Inisialisasi antarmuka pembaca input dari terminal
37   const inputInterface = readline.createInterface({
38     input: process.stdin,
39     output: process.stdout
40   });
41
42   const postalCodeService = new KodePos();
43
44   /**
45    * Fungsi callback untuk memproses input kelurahan dari user.
46    * @param {string} kelurahanInput - Input nama kelurahan
47    */
48   function handleUserInput(kelurahanInput) {
49     const postalCode = postalCodeService.getPostalCodeByKelurahan(kelurahanInput);
50     console.log(`Kode Pos ${kelurahanInput} adalah: ${postalCode}`);
51     inputInterface.close();
52   }
53
54   // Mulai menerima input dari pengguna
55   inputInterface.question("Masukkan nama kelurahan: ", handleUserInput);
56
57 }
```