

1) KodeBuah

- Source Code

Mendeklarasikan kelas KodeBuah. Class ini digunakan sebagai tempat untuk menyimpan data kode buah dan menyediakan fungsi untuk mengembalikan kode buah berdasarkan nama. Pada kelas tersebut terdapat objek yang menyimpan daftar nama buah dan kode buah. Kemudian terdapat export default KodeBuah yang digunakan mengekspor kelas KodeBuah agar bisa digunakan di kelas lain.

```
04_OOP > JURNAL_2311104013 > JS KodeBuah.js > KodeBuah > dataKodeBuah
1  class KodeBuah {
2      static dataKodeBuah = {
3          'Apel' : 'A00',
4          'Aprikot' : 'B00',
5          'Alpukat' : 'C00',
6          'Pisang' : 'D00',
7          'Paprika' : 'E00',
8          'Blackberry' : 'F00',
9          'Ceri' : 'H00',
10         'Kelapa' : 'I00',
11         'Jagung' : 'J00',
12         'Kurma' : 'K00',
13         'Durian' : 'L00',
14         'Anggur' : 'M00',
15         'Melon' : 'N00',
16         'Semangka' : 'O00'
17     };
18
19     static getKodeBuah(buah) {
20         return this.dataKodeBuah[buah] || 'Kode Buah tidak ditemukan'
21     }
22 }
23
24 export default KodeBuah;

1  /*----- */
2  import KodeBuah from "./KodeBuah.js";
3
4  console.log(KodeBuah.getKodeBuah('Apel'))
5  console.log(KodeBuah.getKodeBuah('Mangga'))
6  /*----- */
```

- Output

Memiliki output A00 sesuai dengan objek Apel yang memiliki kode tersebut dan Mangga memunculkan Kode Buah tidak ditemukan karena mangga tidak ada dalam kelas KodeBuah.

```
PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL  PORTS
PS D:\KULIAH\Semester 4\Konstruksi Perangkat Lunak\Praktikum\KPL_Hafizh Dwi Andhika Faruq_2311104013_SE-07-01> cd .\04_OOP\
PS D:\KULIAH\Semester 4\Konstruksi Perangkat Lunak\Praktikum\KPL_Hafizh Dwi Andhika Faruq_2311104013_SE-07-01\04_OOP> cd .\JURNAL_2311104013\
PS D:\KULIAH\Semester 4\Konstruksi Perangkat Lunak\Praktikum\KPL_Hafizh Dwi Andhika Faruq_2311104013_SE-07-01\04_OOP\JURNAL_2311104013> node Main.js
A00
A00
Kode Buah tidak ditemukan
PS D:\KULIAH\Semester 4\Konstruksi Perangkat Lunak\Praktikum\KPL_Hafizh Dwi Andhika Faruq_2311104013_SE-07-01\04_OOP\JURNAL_2311104013>
```

2) PosisiKarakterGame

- Source Code

Kelas PosisiKarakterGame dimulai dengan deklarasi state = "Berdiri"; untuk menentukan posisi awal karakter, lalu nim = 3; sebagai variabel tambahan untuk kondisi tertentu. Objek transitions menyimpan aturan perubahan posisi berdasarkan tombol yang ditekan, misalnya dari "Berdiri" ke "Jongkok" jika "TombolS" ditekan. Method ubahState(tombol) pertama-tama mencari transisi yang sesuai dengan let next = this.transitions[this.state] && this.transitions[this.state][tombol];, lalu memeriksa apakah transisi valid, mencetak "Transisi tidak valid" jika tidak ditemukan. Jika valid, program akan memeriksa nim % 3 untuk mencetak pesan tambahan, seperti saat tombol arah atas atau bawah ditekan. Setelah itu, this.state = next; memperbarui posisi karakter, lalu console.log(this.state); mencetak posisi terbaru ke konsol. Terakhir, export default PosisiKarakterGame; memungkinkan kelas ini digunakan di file lain.

```
04_OOP > JURNAL_2311104013 > JS PosisiKarakterGame.js > PosisiKarakterGame > ubahState
1  class PosisiKarakterGame {
2      static state = "Berdiri"
3      static nim = 3
4
5      static transitions = {
6          "Berdiri" : {"TombolS" : "Jongkok", "TombolW" : "Terbang"},
7          "Jongkok" : {"TombolW" : "Berdiri", "TombolS" : "Tengkurap" },
8          "Tengkurap" : {"TombolW" : "Jongkok"},
9          "Terbang" : {"TombolX" : "Jongkok", "TombolS" : "Berdiri"}
10     }
11
12     static ubahState(tombol) {
13         let next = this.transitions[this.state] && this.transitions[this.state][tombol];
14         if (!next) {
15             console.log("Transisi tidak valid");
16             return;
17         }
18
19         if (this.nim % 3 == 0) {
20             if (tombol == "TombolS") console.log("Tombol arah bawah ditekan");
21             if (tombol == "TombolW") console.log("Tombol arah atas ditekan");
22         }
23
24         if (this.nim % 3 == 1) {
25             if (tombol == "Berdiri") console.log("Posisi standby");
26             if (tombol == "Tengkurap") console.log("Posisi istirahat");
27         }
28
29         if (this.nim % 3 == 2) {
30             if (tombol == "Terbang" && next == "Jongkok") console.log("Posisi landing");
31             if (tombol == "Berdiri" && next == "Terbang") console.log("Posisi take off");
32         }
33
34         this.state == next
35         console.log(this.state)
36     }
37 }
38
39 export default PosisiKarakterGame;
```

```
import PosisiKarakterGame from "../PosisiKarakterGame.js";

const game = new PosisiKarakterGame();
const tombolList = ["TombolS", "TombolS", "TombolW", "TombolW", "TombolW", "TombolX", "TombolS"];

for (let i = 0; i < tombolList.length; i++) {
    game.ubahState(tombolList[i]);
}
```

- Output

Dengan nilai 3 yang ada pada variabel nim menghasilkan output sebagai berikut

```
PS D:\KULIAH\Semester 4\Konstruksi Perangkat Lunak\Praktikum\KPL_Hafizh Dwi Andhika Faruq_2311104013_SE-07-01\04_OOP\JURNAL_2311104013> node Main.js
Tombol arah bawah ditekan
Jongkok
Tombol arah bawah ditekan
Tengkurap
Tombol arah atas ditekan
Jongkok
Tombol arah atas ditekan
Berdiri
Tombol arah atas ditekan
Terbang
Jongkok
Tombol arah bawah ditekan
Tengkurap
```