

JURNAL MODUL 4

Nama : Prajna paramitha  
NIM : 2311104016  
Kelas : SE 07 01

A. MENAMBAHKAN KODE DENGAN TEKNIK TABLE DRIVEN

Dari project yang sudah dibuat sebelumnya:

- A. Buatlah sebuah class bernama “KodeBuah”.
- B. Pada class tersebut, tambahkan sebuah method dengan nama “getKodeBuah” yang mengembalikan kode buah dari tabel yang diberikan di bawah ini.
- C. Setelah method ditambahkan, panggil method tersebut pada class utama/main.

Nama Buah	Kode Buah
Apel	A00
Aprikot	B00
Alpukat	C00
Pisang	D00
Paprika	E00
Blackberry	F00
Ceri	H00
kelapa	I00
Jagung	J00

Nama Buah	Kode Buah
Kurma	K00
Durian	L00
Anggur	M00
Melon	N00
Semangka	O00

```

class KodeBuah {
  constructor() {
    // Table-driven data untuk mapping Kelurahan ke Kode buah
    this.data = {
      "Apel": 'A00',
      "Aprikot": 'B00',
      "Alpukat": 'C00',
      "Pisang": 'D00',
      "Paprika": 'E00',
      "Blackberry": 'F00',
      "Ceri": 'H00',
      "Kelapa": 'I00',
      "Jagung": 'J00',
      "kurma": 'K00',
      "Durian": 'L00',
      "Anggur": 'M00',
      "Melon": 'N00',
      "Semangka": 'O00',
    };
  }
  // B. Pada class tersebut, tambahkan sebuah method dengan nama "getKodeBuah" yang mengembalikan
  // kode buah
  // Method untuk mendapatkan kode pbuahos berdasarkan nama buah
  getKodeBuah(buah_segar) {
    return this.data[buah_segar] || "Kode buah gaada nih, salah deh masukinnya";
  }
}

// Contoh penggunaan dalam class utama/main
const kodeBuah = new KodeBuah();

console.log(kodeBuah.getKodeBuah("Apel"));
console.log(kodeBuah.getKodeBuah("Pisang"));
console.log(kodeBuah.getKodeBuah("Kiwi"));

```

paramitha\_2311104016\_SE0701\04\_OOP\Jurnal\  
A00  
D00  
Kode buah gaada nih, salah deh masukinnya

**SELAMAT MENGERJAKAN!**

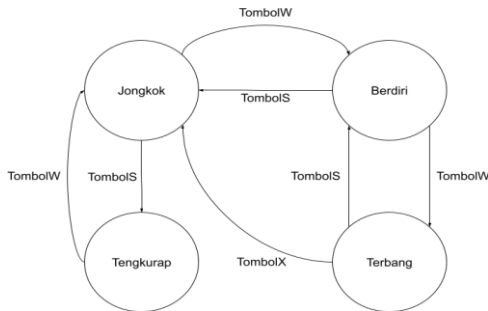
Penjelasan :

- Pertama kita membuat class KodeBuah yang didalamnya berisikan konstruktor yang kemudian kita memiliki table driven berisikan nama buah dan kode buahnya. Selanjutnya terdapat method GetKodeBuah untuk meng get kode buah yang kemudian kita panggil nama buah tersebut di console.log dimana outputnya adalah kode buah.

## B. MENAMBAHKAN KODE DENGAN TEKNIK STATE-BASED CONSTRUCTION

Pada folder project yang sama:

- A. Buatlah sebuah class bernama “PosisiKarakterGame”
- B. Tambahkan implementasi yang menerapkan kasus di bawah ini menggunakan teknik state-based construction (asumsikan state awal adalah berdiri):



C. Berikan implementasi tambahkan berdasarkan hasil mod dari NIM

D. Berikan implementasi tambahkan berdasarkan hasil mod dari NIM

i.  $NIM \% 3 == 0$ :

1. Pada saat TombolS ditekan, maka akan ada output “tombol arah bawah ditekan”
2. Pada saat TombolW ditekan, maka akan ada output “tombola rah atas ditekan”

**SELAMAT MENGERJAKAN!**

```
class PosisiKarakterGame {
  constructor() {
    // Disini Set State awal karakter adalah "Berdiri"
    this.state = "Berdiri";
  }

  // Method untuk mengubah posisi karakter
  ubahPosisi(tombol) {
    if (tombol === "S") {
      console.log("Tombol arah bawah ditekan"); // Output tambahan saat tombol S ditekan
    } else if (tombol === "W") {
      console.log("Tombol arah atas ditekan"); // Output tambahan saat tombol W ditekan
    }

    // Objek transisi menentukan perubahan posisi karakter
    const transisi = {
      "Berdiri": { "S": "Jongkok", "W": "Terbang" },
      "Jongkok": { "S": "Tengkurap", "W": "Berdiri" },
      "Tengkurap": { "W": "Jongkok" },
      "Terbang": { "S": "Berdiri", "X": "Berdiri" } // Jatuh langsung ke Berdiri
    };

    // cek ada ga transisi yang sesuai dengan tombol yang ditekan
    if (transisi[this.state] && transisi[this.state][tombol]) {
      this.state = transisi[this.state][tombol]; // Ubah posisi karakter
    }

    // Menampilkan posisi karakter setelah perubahan
    console.log(`Posisi sekarang: ${this.state}`);
  }
}

const karakter = new PosisiKarakterGame();
karakter.ubahPosisi("W"); // Output tambahan: "Tombol arah atas ditekan" + Posisi berubah: Berdiri ->
Terbang
karakter.ubahPosisi("S"); // Output tambahan: "Tombol arah bawah ditekan" + Posisi berubah: Terbang ->
Berdiri
karakter.ubahPosisi("S"); // Output tambahan: "Tombol arah bawah ditekan" + Posisi berubah: Berdiri ->
Jongkok
karakter.ubahPosisi("S"); // Output tambahan: "Tombol arah bawah ditekan" + Posisi berubah: Jongkok ->
Tengkurap
karakter.ubahPosisi("W"); // Output tambahan: "Tombol arah atas ditekan" + Posisi berubah: Tengkurap ->
Jongkok
```

```
param1: 2311104010_SLO701
Tombol arah atas ditekan
Posisi sekarang: Terbang
Tombol arah bawah ditekan
Posisi sekarang: Berdiri
Tombol arah bawah ditekan
Posisi sekarang: Jongkok
Tombol arah bawah ditekan
Posisi sekarang: Tengkurap
Tombol arah atas ditekan
Posisi sekarang: Jongkok
```

### Penjelasan

- Class posisi karakter : class nya, dimana didalamnya terdapat konstruktor yang kemudian terdapat method ubahPosisi untuk mengubah posisi dari karakternya.
- Karena ada penambahan sesuai dengan mod nim, maka ada output “Tombola rah bawah ditekan” dan “tombola rah atas ditekan”
- Kemudian menentukan transisi dari karakternya. Dari beriri -> jongkok kemudian terbang
- Lalu dibagian bawah menampilkan outputnya.