**LAPORAN PRAKTIKUM**

**Pemrograman Berbasis Objek**

**Jobsheet - 4: Relasi**



|  |  |
| --- | --- |
| Nama | : Ghoffar Abdul Ja’far |
| NIM | : 2341720035 |
| Kelas | : 2F |

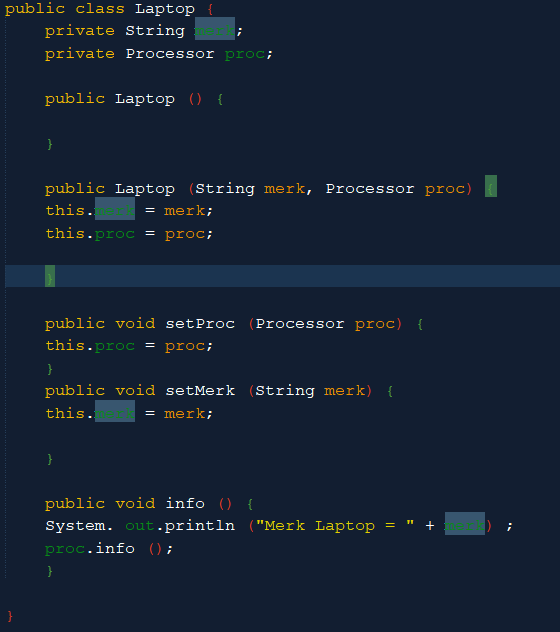
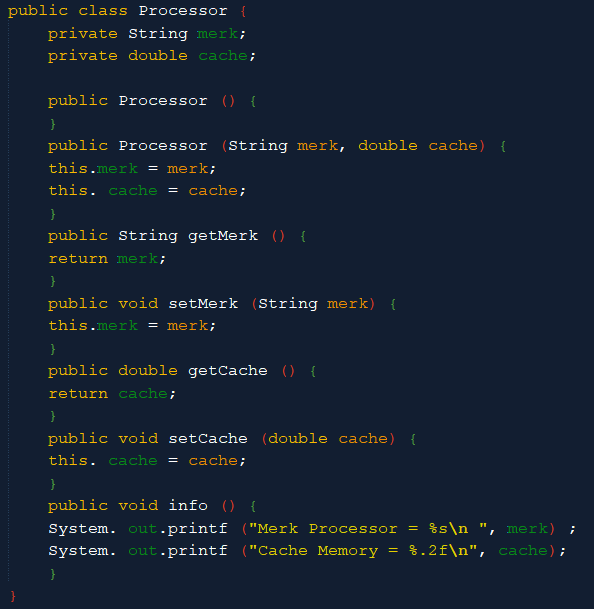
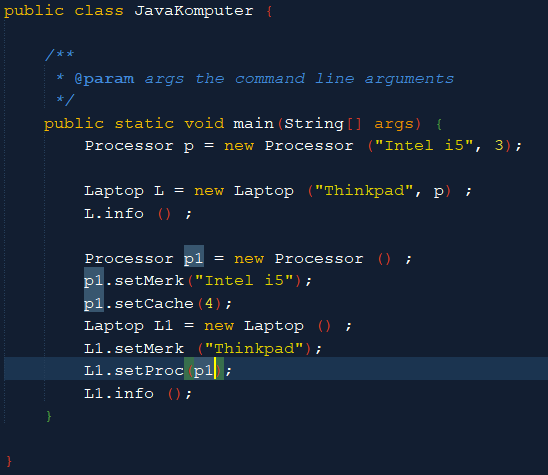
**JURUSAN TEKNOLOGI INFORMASI**

**POLITEKNIK NEGERI MALANG**

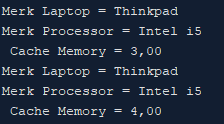
**2023/2024**

**Percobaan 1**

* **Code:**

* **Hasil:**



**Pertanyaan:**

1. Di dalam class Processor dan class Laptop , terdapat method setter dan getter untuk masing‑masing atributnya. Apakah gunanya method setter dan getter tersebut ?

**Jawab: setter digunakan untuk mengubah atau memasukkan data, sedangkan getter digunakan untuk mengambil data**

1. Di dalam class Processor dan class Laptop, masing‑masing terdapat konstruktor default dan konstruktor berparameter. Bagaimanakah beda penggunaan dari kedua jenis konstruktor tersebut ?

**Jawab: konstruktor default perlu getter dan setter untuk pengolahan data, sedangkan konstruktor berparameter sudah mencakup getter dan setter, bisa langsung diinputkan saat melakukan instansiasi object**

1. Perhatikan class Laptop, di antara 2 atribut yang dimiliki (merk dan proc), atribut manakah yang bertipe object ?

**Jawab: atribut proc**

1. Perhatikan class Laptop, pada baris manakah yang menunjukan bahwa class Laptop memiliki relasi dengan class Processor ?

**Jawab: pada saat deklarasi atribut proc pada class Laptop, karena atribut proc merupakan object**

1. Perhatikan pada class Laptop , Apakah guna dari sintaks proc.info() ?

**Jawab: digunakan untuk menampilkan informasi processor**

1. Pada class MainPercobaan1, terdapat baris kode:

Laptop l = new Laptop("Thinkpad", p);.

Apakah p tersebut ?

**Jawab: p sendiri adalah object dari class Processor**

Dan apakah yang terjadi jika baris kode tersebut diubah menjadi:

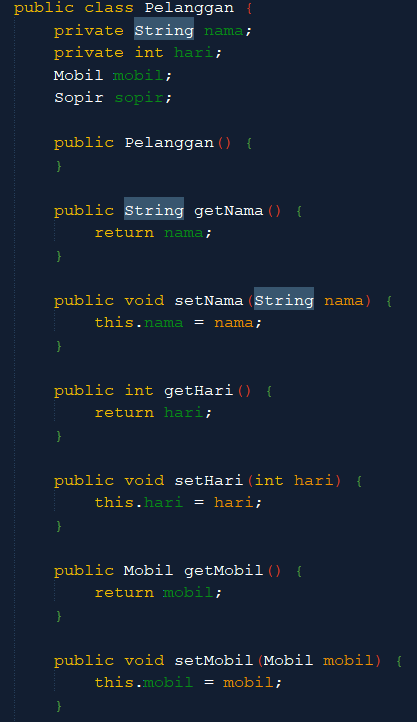
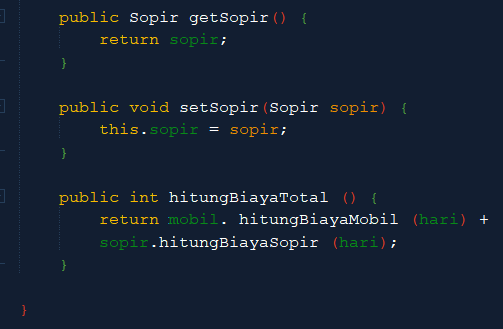
Laptop l = new Laptop("Thinkpad", new Processor("Intel i5", 3));

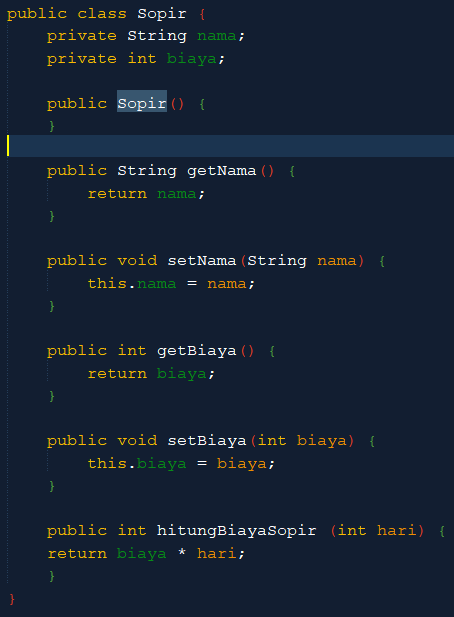
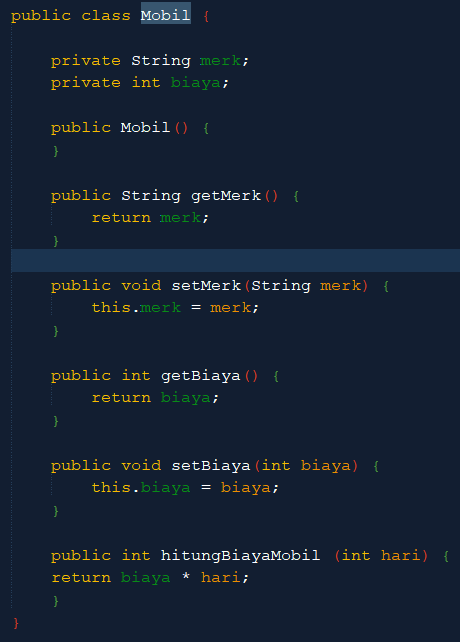
Bagaimanakah hasil program saat dijalankan, apakah ada perubahan ?

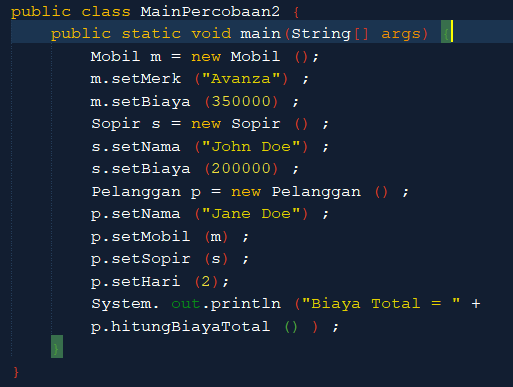
**Jawab: hasilnya akan tetap sama, hanya saja instansiasi object preocessor dilakukan langsung pada saat instansiasi object laptop**

**Percobaan 2**

* **Code:**



* **Hasil:**



**Pertanyaan:**

1. Perhatikan class Pelanggan. Pada baris program manakah yang menunjukan bahwa class Pelanggan memiliki relasi dengan class Mobil dan class Sopir ?

**Jawab: pada saat mendeklarasikan atribut mobil dan sopir, yang mana itu mengambil dari object class mobil dan sopir**

1. Perhatikan method hitungBiayaSopir pada class Sopir, serta method hitungBiayaMobil pada class Mobil. Mengapa menurut Anda method tersebut harus memiliki argument hari?

**Jawab: karena perentalan mobil juga menghitung seberapa lama mobil itu disewa**

1. Perhatikan kode dari class Pelanggan. Untuk apakah perintah mobil.hitungBiayaMobil(hari) dan sopir.hitungBiayaSopir(hari) ?

**Jawab: untuk menghitung total biaya mobil dan sopir perharinya**

1. Perhatikan class MainPercobaan2. Untuk apakah sintaks p.setMobil(m) dan p.setSopir(s)?

**Jawab: digunakan untuk mengisi value dari atribut mobil dan sopir dari object yang telah dibuat dengan menggunakan setter**

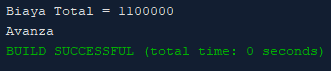
1. Perhatikan class MainPercobaan2. Untuk apakah proses p.hitungBiayaTotal() tersebut ?

**Jawab: digunakan untuk menghitung total biaya sewa mobil yang harus ditanggung pelanggan**

1. Perhatikan class MainPercobaan2, coba tambahkan pada baris terakhir dari method main dan amati perubahan saat di‑run!

System.out.println(p.getMobil().getMerk());

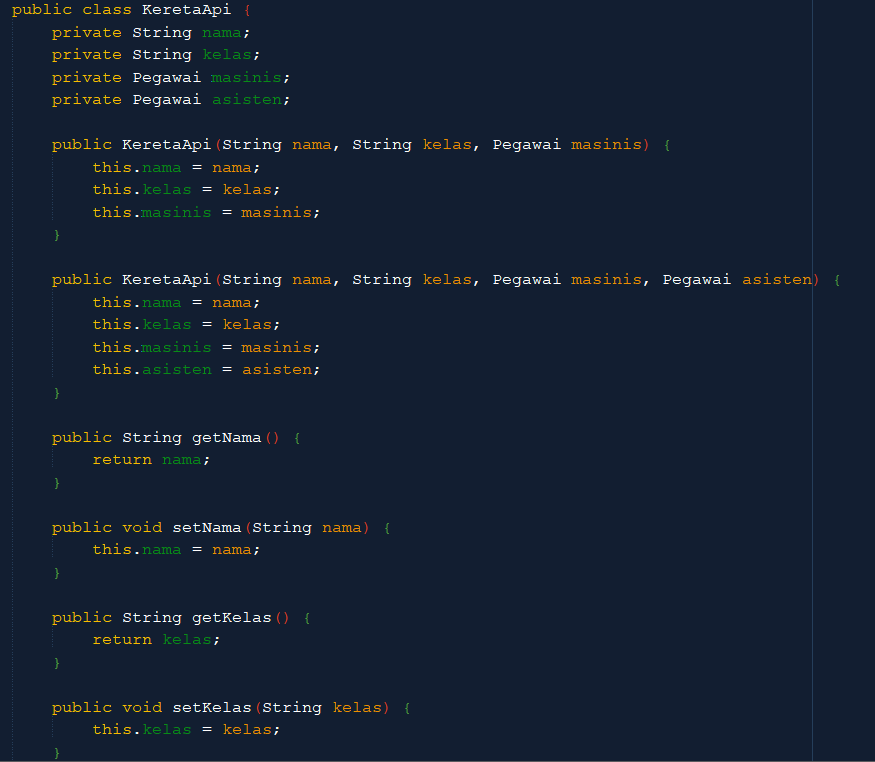
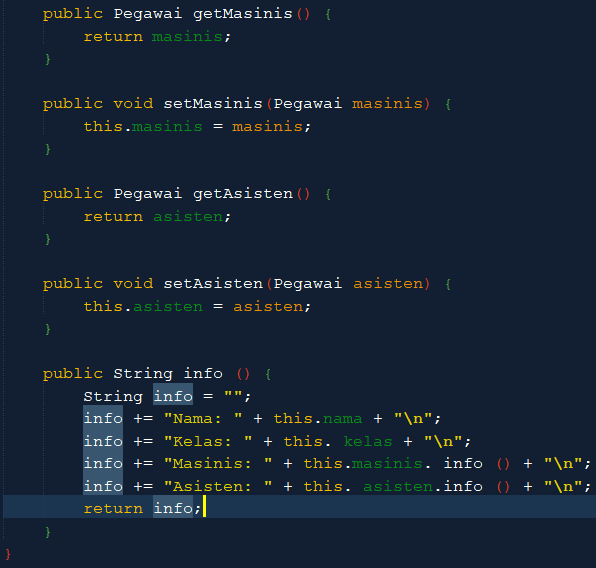
Jadi untuk apakah sintaks p.getMobil().getMerk() yang ada di dalam method main tersebut?

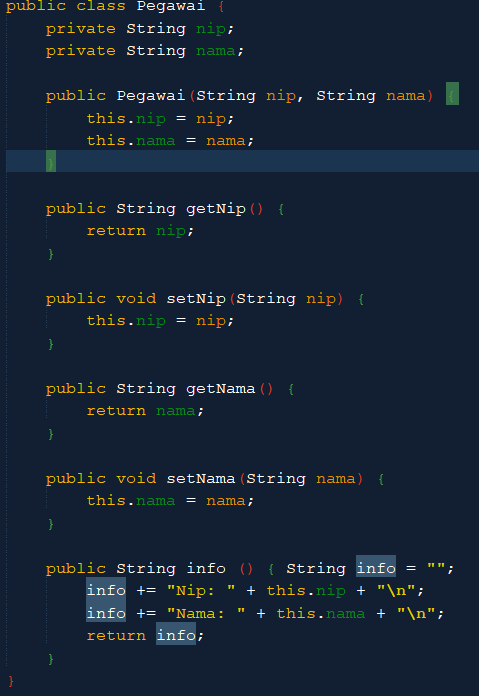
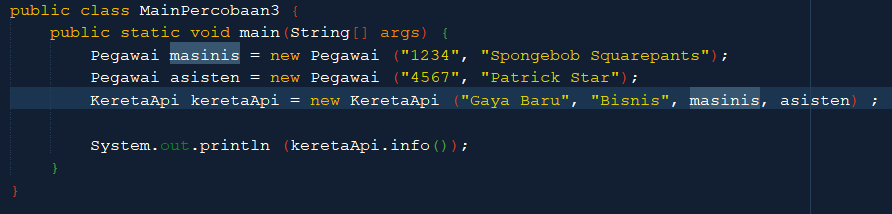
**Hasil: **

**Jawab: syntax tersebut digunakan untuk menampilkan merk dari sebuah mobil yang disewa oleh pelanggan**

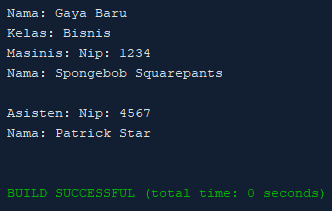
**Percobaan 3**

* **Code:**

* **Hasil:**



**Pertanyaan:**

1. Di dalam method info() pada class KeretaApi, baris this.masinis.info() dan this.asisten.info() digunakan untuk apa ?

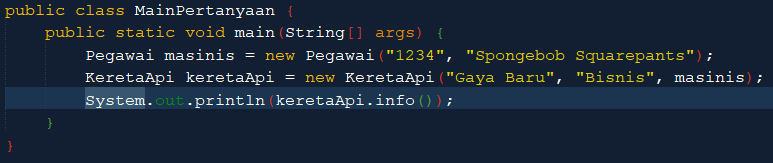
**Jawab: digunakan untuk menampilkan data object class pegawai yang tersimpan pada atribut object masinis dan asisten**

1. Buatlah main program baru dengan nama class MainPertanyaan pada package yang sama. Tambahkan kode berikut pada method main() !

Pegawai masinis = new Pegawai("1234", "Spongebob Squarepants");

KeretaApi keretaApi = new KeretaApi("Gaya Baru", "Bisnis", masinis);

System.out.println(keretaApi.info());

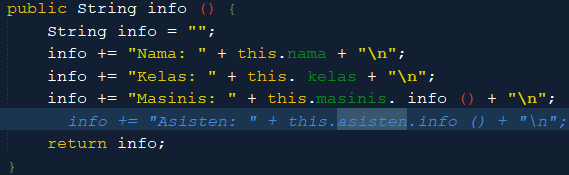


1. Apa hasil output dari main program tersebut ? Mengapa hal tersebut dapat terjadi ?

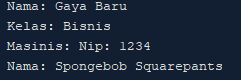
**Jawab: terdapat error yang mengarah ke object keretaApi, yang mana pada saat memanggil fungsi info() pada class KeretaApi output menjadi nullPointerException karena pada class KeretaApi terkhusus fungsi info() menampilkan atribut asisten juga sebagai output, sedangkan atribut asisten tidak terisi**

1. Perbaiki class KeretaApi sehingga program dapat berjalan !

**Code:**

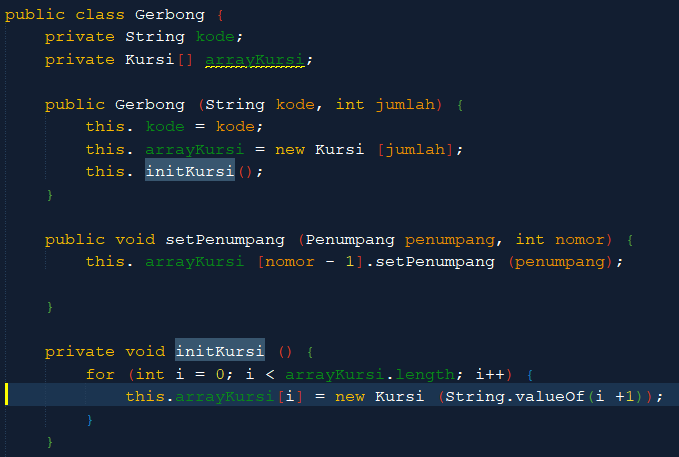
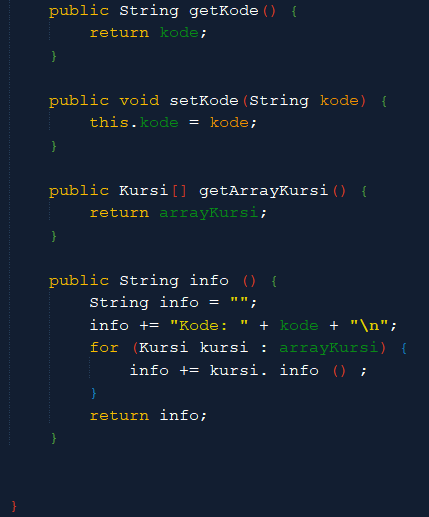
****

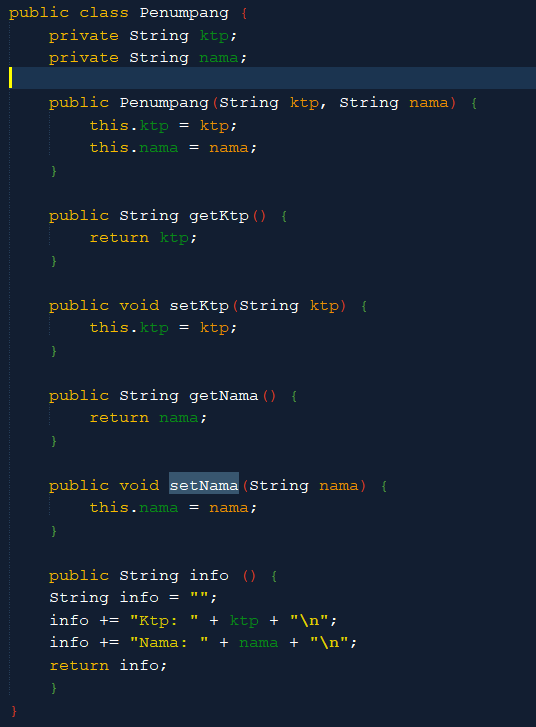
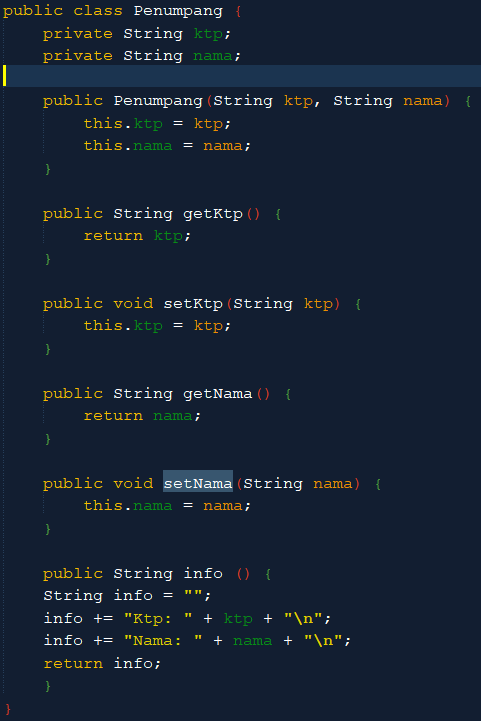
**Hasil:**

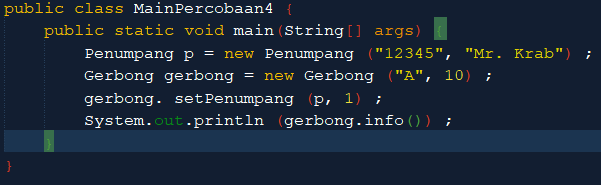
****

**Percobaan 4**

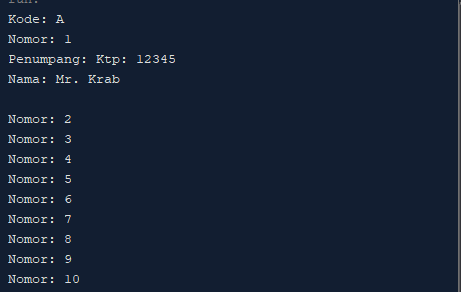
* **Code:**



* **Hasil:**



**Pertanyaan:**

1. Pada main program dalam class MainPercobaan4, berapakah jumlah kursi dalam Gerbong A ?

**Jawab: jumlah kursi pada Gerbong A adalah 10**

1. Perhatikan potongan kode pada method info() dalam class Kursi. Apa maksud kode tersebut ?

...

if (this.penumpang != null) {

info += "Penumpang: " + penumpang.info() + "\n";

}

...

**Jawab: menujukkan adanya kondisi jika value dari atribut penumpang tidak sama dengan null maka value dari penumpang akan ditampilkan**

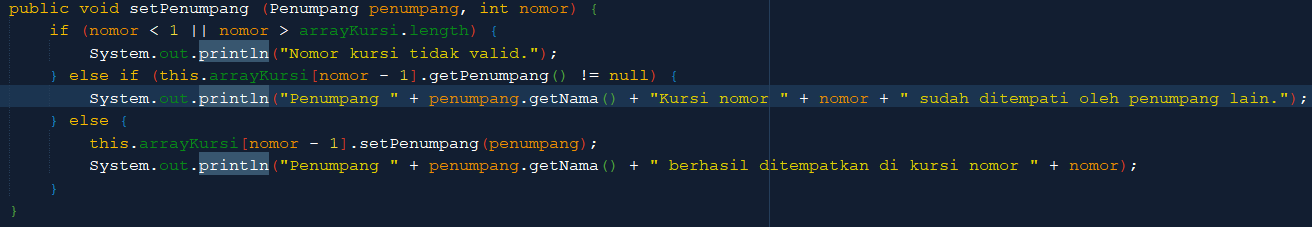
1. Mengapa pada method setPenumpang() dalam class Gerbong, nilai nomor dikurangi dengan angka 1 ?

**Jawab: karena array dimulai dari 0 dan jika memilih index dari value atribut maka perlu dikurangi 1 karna jika pada value atribut angka dimulai dari 1**

1. Instansiasi objek baru budi dengan tipe Penumpang, kemudian masukkan objek baru tersebut pada gerbong dengan gerbong.setPenumpang(budi, 1). Apakah yang terjadi ?

**Jawab: penumpang pada kursi nomor 1 tergantikan oleh budi karena value pada kursi nomor 1 ditumpuk oleh budi**

1. Modifikasi program sehingga tidak diperkenankan untuk menduduki kursi yang sudah ada penumpang lain !

* **Code:**
* **Hasil:**

