

1. Program tersebut adalah sebuah kelas Java bernama Elektronik yang berada di dalam package com.project.

2. Deklarasi Kelas dan Atribut:

java

Salin kode

```
public class Elektronik {  
    private final String merek, model;  
    private final int harga;  
}
```

- Elektronik: Nama kelas.
- merek, model, harga: Atribut dari kelas Elektronik yang bersifat private dan final. private berarti atribut ini hanya bisa diakses dari dalam kelas tersebut, dan final berarti nilainya tidak bisa diubah setelah diinisialisasi.

3. Konstruktor:

java

Salin kode

```
public Elektronik(String merek, String model, int harga) {  
    this.merek = merek;  
    this.model = model;  
    this.harga = harga;  
}
```

- Konstruktor ini digunakan untuk menginisialisasi objek Elektronik dengan nilai merek, model, dan harga yang diberikan sebagai parameter.

4. Metode informasi:

java

Salin kode

```
public void informasi() {  
    System.out.println("Merek      : " + merek);  
    System.out.println("Model      : " + model);  
    System.out.println("Harga      : " + harga);  
}
```

```
}
```

- Metode ini menampilkan informasi merek, model, dan harga dari objek Elektronik ke layar tanpa ada prefix tambahan.

5. Metode informasi Overload:

java

Salin kode

```
public void informasi(String prefix) {  
    System.out.println(prefix + "Merek      : " + merek);  
    System.out.println(prefix + "Model      : " + model);  
    System.out.println(prefix + "Harga      : " + harga);  
}
```

- Metode ini juga menampilkan informasi merek, model, dan harga dari objek Elektronik, tetapi dengan prefix tambahan yang diberikan sebagai parameter. Ini adalah contoh dari metode overloading di mana ada dua metode dengan nama yang sama tetapi parameter yang berbeda.

Secara keseluruhan, program ini memungkinkan pembuatan objek Elektronik dengan merek, model, dan harga tertentu, serta menyediakan dua cara untuk menampilkan informasi tersebut: satu tanpa prefix dan satu lagi dengan prefix.

2. Program tersebut adalah sebuah kelas Java bernama Handphone yang berada di dalam package com.project. Kelas Handphone ini adalah subclass dari kelas Elektronik, yang berarti Handphone mewarisi atribut dan metode dari Elektronik serta menambahkan atribut dan metode spesifik untuk Handphone.

1. Deklarasi Kelas dan Atribut:

java

Salin kode

```
class Handphone extends Elektronik {  
    private final double layar;  
    private final String CPU;  
    private final int baterai, RAM, storage;  
}
```

- Handphone: Nama kelas yang mewarisi kelas Elektronik.

- o layar, CPU, baterai, RAM, storage: Atribut tambahan dari kelas Handphone yang bersifat private dan final.

2. Konstruktor:

java

Salin kode

```
public Handphone(String merek, String model, int harga, double layar, String CPU,
int baterai, int RAM, int storage) {
    super(merek, model, harga);
    this.layar = layar;
    this.CPU = CPU;
    this.baterai = baterai;
    this.RAM = RAM;
    this.storage = storage;
}
```

- o Konstruktor ini digunakan untuk menginisialisasi objek Handphone dengan nilai merek, model, harga, layar, CPU, baterai, RAM, dan storage yang diberikan sebagai parameter.
- o super(merek, model, harga): Memanggil konstruktor dari kelas Elektronik untuk menginisialisasi atribut merek, model, dan harga.

3. Metode informasi Override:

java

Salin kode

@Override

```
public void informasi() {
    super.informasi();
    System.out.println("Ukuran Layar : " + layar + " Inci");
    System.out.println("CPU      : " + CPU);
    System.out.println("Baterai   : " + baterai + " mAh");
    System.out.println("RAM       : " + RAM + " GB");
    System.out.println("Storage  : " + storage + " GB\n");
}
```

- o Metode ini menimpa (override) metode informasi dari kelas Elektronik.

- `super.informasi()`: Memanggil metode informasi dari kelas `Elektronik` untuk menampilkan informasi merek, model, dan harga.
- Menampilkan informasi tambahan layar, CPU, baterai, RAM, dan storage spesifik untuk `Handphone`.

4. Metode informasi Overload dengan Prefix:

java

Salin kode

@Override

```
public void informasi(String prefix) {
    super.informasi(prefix);
    System.out.println(prefix + "Ukuran Layar : " + layar + " Inchi");
    System.out.println(prefix + "CPU      : " + CPU);
    System.out.println(prefix + "Baterai   : " + baterai + " mAh");
    System.out.println(prefix + "RAM      : " + RAM + " GB");
    System.out.println(prefix + "Storage   : " + storage + " GB\n");
}
```

- Metode ini juga menerima (override) metode informasi dengan prefix dari kelas `Elektronik`.
- `super.informasi(prefix)`: Memanggil metode informasi dengan prefix dari kelas `Elektronik`.
- Menampilkan informasi tambahan layar, CPU, baterai, RAM, dan storage dengan prefix.

Secara keseluruhan, kelas `Handphone` memungkinkan pembuatan objek `Handphone` yang memiliki atribut tambahan selain atribut dasar dari `Elektronik`. Metode informasi yang di-override digunakan untuk menampilkan informasi lengkap dari objek `Handphone`, baik dengan atau tanpa prefix tambahan.

3. Program tersebut adalah sebuah kelas Java bernama `Laptop` yang berada di dalam package `com.project`. Kelas `Laptop` ini adalah subclass dari kelas `Elektronik`, sehingga mewarisi atribut dan metode dari `Elektronik` dan menambahkan atribut serta metode spesifik untuk `Laptop`.

- **class Laptop extends Elektronik:** Baris ini mendefinisikan kelas bernama `Laptop` yang mewarisi dari kelas bernama `Elektronik`. Pewarisan memungkinkan kelas `Laptop` untuk menggunakan kembali properti dan metode dari kelas `Elektronik`.

Variabel Anggota:

- **private final double ukuranLayar;** Ini mendeklarasikan variabel privat final bernama ukuranLayar dengan tipe double untuk menyimpan ukuran layar laptop dalam inci. Kata kunci final membuat variabel ini konstan setelah inisialisasi.
- **private final String CPU;**, **private final String VGA;** Ini mendeklarasikan variabel privat final String bernama CPU dan VGA untuk menyimpan model CPU dan kartu VGA laptop.
- **private final int RAM, storage;** Ini mendeklarasikan variabel integer privat final bernama RAM dan storage untuk mewakili kapasitas RAM laptop dalam gigabyte (GB) dan kapasitas penyimpanan dalam GB.

Konstruktor:

- **public Laptop(String merek, String model, int harga, double ukuranLayar, String CPU, String VGA, int RAM, int storage):** Ini adalah konstruktor kelas. Ini adalah metode khusus yang dipanggil ketika objek Laptop baru dibuat. Ini membutuhkan argumen berikut:
 - merek: String yang mewakili merek laptop
 - model: String yang mewakili model laptop
 - harga: int yang mewakili harga laptop
 - ukuranLayar: double yang mewakili ukuran layar
 - CPU: String yang mewakili model CPU
 - VGA: String yang mewakili model kartu VGA
 - RAM: int yang mewakili kapasitas RAM
 - storage: int yang mewakili kapasitas penyimpanan
 - Di dalam konstruktor:
 - **super(merek, model, harga);** Baris ini memanggil konstruktor dari kelas induk (Elektronik) untuk menginisialisasi propertinya (merek, model, dan harga).
 - Baris sisanya menetapkan nilai yang diberikan ke variabel kelas Laptop yang sesuai.

Metode:

- **@Override public void informasi() { ... }:** Ini adalah metode yang ditimpa (overridden) bernama informasi. Anotasi @Override menunjukkan bahwa metode ini mendefinisikan ulang metode yang diwarisi dari kelas induk (Elektronik). Mungkin memiliki tanda tangan (nama dan parameter) yang sama tetapi memberikan implementasi khusus untuk objek Laptop.

- `super.informasi();`** : Baris ini memanggil metode informasi dari kelas induk untuk menampilkan informasinya (mungkin merek, model, dan harga).
- Baris berikutnya mencetak detail tentang ukuran layar, CPU, VGA, RAM, dan penyimpanan laptop.
- **@Override public void informasi(String prefix) { ... }** : Ini adalah metode lain yang ditimpa bernama `informasi`, tetapi ini mengambil argumen `String` bernama `prefix`. Ini kemungkinan menyediakan cara yang lebih dapat disesuaikan untuk menampilkan informasi.
 - `super.informasi(prefix);`** : Baris ini memanggil metode informasi dari kelas induk, meneruskan argumen `prefix` untuk berpotensi menyesuaikan outputnya (dengan asumsi kelas induk memiliki metode `informasi(String prefix)`).
 - Mirip dengan metode informasi sebelumnya, baris berikutnya mencetak detail spesifikasi laptop, menambahkan awalan pada setiap baris dengan `prefix` yang disediakan.

Kesimpulan:

Kelas `Laptop` mewakili objek laptop, mewarisi properti dari kelas `Elektronik` dan menyediakan atribut tambahan khusus untuk laptop (ukuran layar, CPU, VGA, RAM, penyimpanan). Konstruktor menginisialisasi properti ini, dan metode `informasi` menampilkan informasi tentang objek `Laptop` dengan cara yang berbeda (dengan atau tanpa awalan).

4. □ Paket dan Impor

java

Salin kode

```
package com.project;
```

```
import javax.swing.JOptionPane;
```

- `package com.project;` menyatakan bahwa kelas `Main` berada dalam paket `com.project`.
- `import javax.swing.JOptionPane;` mengimpor kelas `JOptionPane` dari pustaka `javax.swing` yang digunakan untuk menampilkan dialog.

□ Kelas Utama dan Metode main

java

Salin kode

```
public class Main {
```

```
    public static void main(String[] args) {
```

```
// Kode di sini...  
}  
}
```

- public class Main mendefinisikan kelas utama bernama Main.
- public static void main(String[] args) adalah metode utama yang akan dieksekusi saat program berjalan.

□ Mengambil Input untuk Laptop

java

Salin kode

```
String merekLaptop = JOptionPane.showInputDialog("Masukkan merek laptop:");  
String modelLaptop = JOptionPane.showInputDialog("Masukkan model laptop:");  
int hargaLaptop = Integer.parseInt(JOptionPane.showInputDialog("Masukkan harga laptop:"));  
double ukuranLayar =  
Double.parseDouble(JOptionPane.showInputDialog("Masukkan ukuran layar laptop:"));  
String CPULaptop = JOptionPane.showInputDialog("Masukkan CPU laptop:");  
String VGALaptop = JOptionPane.showInputDialog("Masukkan VGA laptop:");  
int RAMLaptop = Integer.parseInt(JOptionPane.showInputDialog("Masukkan RAM laptop (GB):"));  
int storageLaptop = Integer.parseInt(JOptionPane.showInputDialog("Masukkan storage laptop (GB):"));
```

- Menggunakan JOptionPane.showInputDialog untuk menampilkan dialog input dan mengambil nilai dari pengguna.
- Integer.parseInt dan Double.parseDouble digunakan untuk mengonversi input string menjadi tipe data int dan double.

□ Membuat Objek Laptop

java

Salin kode

```
Laptop laptop = new Laptop(merekLaptop, modelLaptop, hargaLaptop, ukuranLayar,  
CPU, VGALaptop, RAMLaptop, storageLaptop);
```

- Membuat objek Laptop dengan parameter yang diambil dari input pengguna.

□ Mengambil Input untuk Handphone

java

Salin kode

```
String merekHandphone = JOptionPane.showInputDialog("Masukkan merek  
handphone:");  
  
String modelHandphone = JOptionPane.showInputDialog("Masukkan model  
handphone:");  
  
int hargaHandphone = Integer.parseInt(JOptionPane.showInputDialog("Masukkan  
harga handphone:"));  
  
double layar = Double.parseDouble(JOptionPane.showInputDialog("Masukkan  
ukuran layar handphone:"));  
  
String CPUHandphone = JOptionPane.showInputDialog("Masukkan CPU  
handphone:");  
  
int baterai = Integer.parseInt(JOptionPane.showInputDialog("Masukkan kapasitas  
baterai handphone (mAh):"));  
  
int RAMHandphone = Integer.parseInt(JOptionPane.showInputDialog("Masukkan  
RAM handphone (GB):"));  
  
int storageHandphone = Integer.parseInt(JOptionPane.showInputDialog("Masukkan  
storage handphone (GB):"));
```

- Menggunakan JOptionPane.showInputDialog untuk mengambil input dari pengguna, serupa dengan bagian laptop.

□ **Membuat Objek Handphone**

java

Salin kode

```
Handphone handphone = new Handphone(merekHandphone, modelHandphone,  
hargaHandphone, layar, CPUHandphone, baterai, RAMHandphone,  
storageHandphone);
```

- Membuat objek Handphone dengan parameter yang diambil dari input pengguna.

□ **Menampilkan Informasi Laptop dan Handphone**

java

Salin kode

```
System.out.println("\n    Informasi Laptop :");  
  
System.out.println("=====");  
  
laptop.informasi("Laptop: ");  
  
  
  
System.out.println("    Informasi Handphone :");
```



```
System.out.println("=====");
```

```
handphone.informasi("Handphone: ");
```

- Menampilkan informasi tentang laptop dan handphone menggunakan metode informasi dari objek Laptop dan Handphone.