

# TUGAS MINGGUAN KE 2 PEMROGRAMAN BERORIENTASI OBJEK



DISUSUN OLEH

NAMA : ADHITYA WIRA YUDA

NIM : 2200018399

KELAS : I

MATKUL : PBO

DOSEN PENGAMPU : Herman Yuliansyah, S.T., M.Eng.

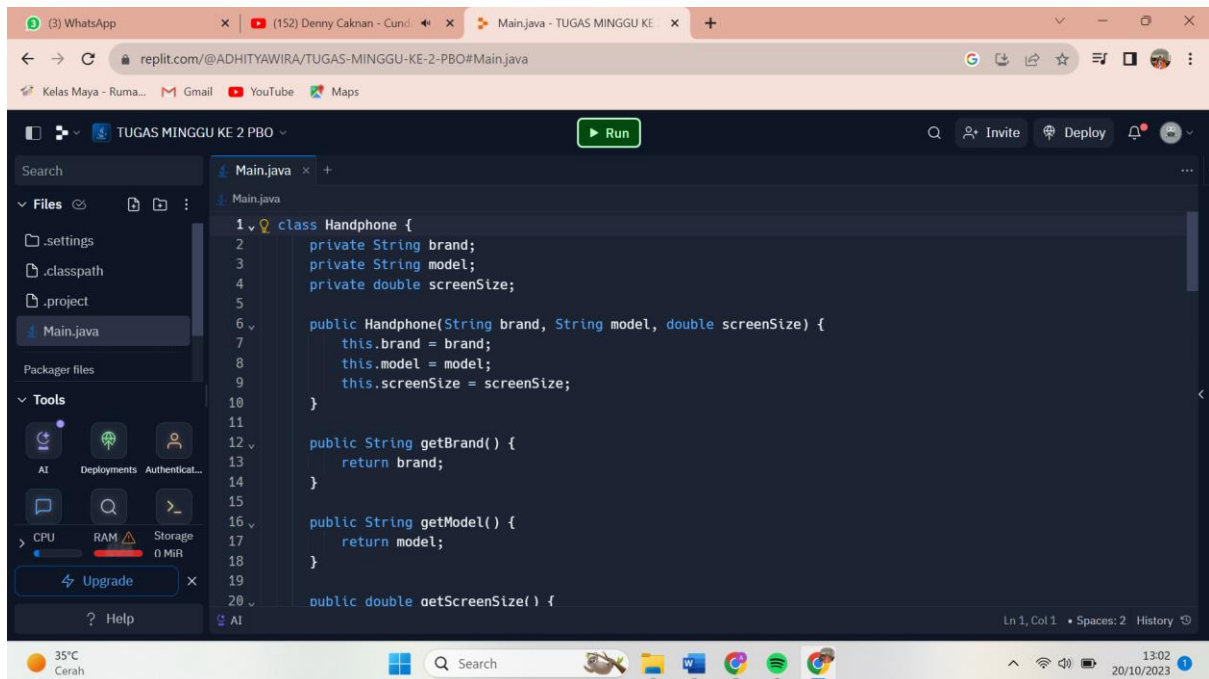
FAKULTAS TEKNOLOGI INDUSTRI

PROGRAM STUDI INFORMATIKA

UNIVERSITAS AHMAD DAHLAN

TAHUN AJARAN 2023/2024

## 1. DEFINISI CLASS HANDPHONE



```
1 class Handphone {
2     private String brand;
3     private String model;
4     private double screenSize;
5
6     public Handphone(String brand, String model, double screenSize) {
7         this.brand = brand;
8         this.model = model;
9         this.screenSize = screenSize;
10    }
11
12    public String getBrand() {
13        return brand;
14    }
15
16    public String getModel() {
17        return model;
18    }
19
20    public double getScreenSize() {
```

### ➤ Atribut Kelas:

- brand: Ini adalah atribut bertipe data String yang digunakan untuk menyimpan merek (brand) ponsel.
- model: Ini adalah atribut bertipe data String yang digunakan untuk menyimpan model ponsel.
- screenSize: Ini adalah atribut bertipe data double yang digunakan untuk menyimpan ukuran layar (screen size) ponsel.

### ➤ Konstruktor:

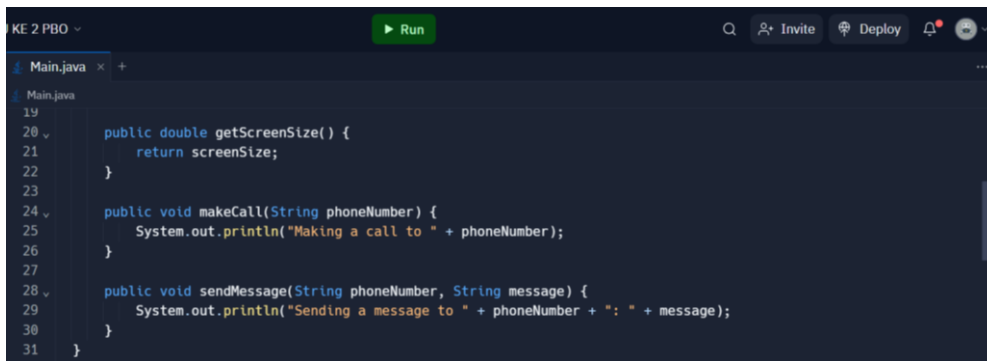
- Kelas Handphone memiliki satu konstruktor yang digunakan untuk membuat objek Handphone. Konstruktor ini menerima tiga parameter yaitu brand, model, dan screenSize, dan menginisialisasi atribut-atribut kelas dengan nilai-nilai yang diberikan melalui parameter.

### ➤ Metode getBrand():

- Metode ini adalah metode publik (public) yang digunakan untuk mengambil nilai dari atribut brand. Metode ini mengembalikan nilai atribut brand.

### ➤ Metode getModel():

- Metode ini juga adalah metode publik yang digunakan untuk mengambil nilai dari atribut model. Metode ini mengembalikan nilai atribut model.



```
19
20 public double getScreenSize() {
21     return screenSize;
22 }
23
24 public void makeCall(String phoneNumber) {
25     System.out.println("Making a call to " + phoneNumber);
26 }
27
28 public void sendMessage(String phoneNumber, String message) {
29     System.out.println("Sending a message to " + phoneNumber + ": " + message);
30 }
31 }
```

➤ Metode `getScreenSize()`:

- Ini adalah metode publik yang digunakan untuk mengambil nilai dari atribut `screenSize`, yang merupakan ukuran layar dari ponsel. Metode ini mengembalikan nilai atribut `screenSize` dalam bentuk `double`.

➤ Metode `makeCall(String phoneNumber)`:

- Ini adalah metode publik yang digunakan untuk mensimulasikan proses membuat panggilan telepon. Metode ini menerima satu parameter yaitu `phoneNumber`, yang merupakan nomor telepon yang akan dihubungi. Metode ini mencetak pesan "Making a call to " diikuti oleh nomor telepon yang ditentukan ke konsol.

➤ Metode `sendMessage(String phoneNumber, String message)`:

- Ini adalah metode publik yang digunakan untuk mensimulasikan proses mengirim pesan teks. Metode ini menerima dua parameter yaitu `phoneNumber`, yang merupakan nomor telepon tujuan, dan `message`, yang merupakan pesan yang akan dikirim. Metode ini mencetak pesan "Sending a message to " diikuti oleh nomor telepon tujuan dan pesan yang ditentukan ke konsol.

## 2. CLASS SMARTPHONE EXTENDS HANDPHONE

```
33 class Smartphone extends Handphone {
34     private String operatingSystem;
35     private boolean hasTouchScreen;
36
37     public Smartphone(String brand, String model, double screenSize, String operatingSystem, boolean
hasTouchScreen) {
38         super(brand, model, screenSize);
39         this.operatingSystem = operatingSystem;
40         this.hasTouchScreen = hasTouchScreen;
41     }
42
43     public String getOperatingSystem() {
44         return operatingSystem;
45     }
46
47     public boolean hasTouchScreen() {
48         return hasTouchScreen;
49     }
50
51     public void installApp(String appName) {
```

➤ Kelas Smartphone:

- Kelas Smartphone adalah subkelas dari kelas Handphone. Ini berarti bahwa kelas Smartphone mewarisi semua atribut dan metode dari kelas Handphone, dan juga memiliki atribut dan metode tambahan yang didefinisikan sendiri.

➤ Atribut Kelas Smartphone:

- operatingSystem: Ini adalah atribut bertipe data String yang digunakan untuk menyimpan informasi tentang sistem operasi (operating system) yang digunakan oleh smartphone.
- hasTouchScreen: Ini adalah atribut bertipe data boolean yang menunjukkan apakah smartphone memiliki layar sentuh (touchscreen) atau tidak.

➤ Konstruktor Smartphone:

- Kelas Smartphone memiliki satu konstruktor yang menerima lima parameter: brand, model, screenSize, operatingSystem, dan hasTouchScreen. Konstruktor ini memanggil konstruktor kelas induk Handphone menggunakan super(brand, model, screenSize) untuk menginisialisasi atribut-atribut warisan (dari kelas Handphone) dan kemudian menginisialisasi atribut khusus Smartphone yaitu operatingSystem dan hasTouchScreen dengan nilai-nilai yang diberikan melalui parameter.

➤ Metode getOperatingSystem():

- Ini adalah metode publik yang digunakan untuk mengambil informasi tentang sistem operasi (operating system) yang digunakan oleh smartphone. Metode ini mengembalikan nilai atribut operatingSystem.

➤ Metode hasTouchScreen():

- Ini adalah metode publik yang digunakan untuk mengecek apakah smartphone memiliki layar sentuh (touchscreen) atau tidak. Metode ini mengembalikan nilai atribut `hasTouchScreen`.

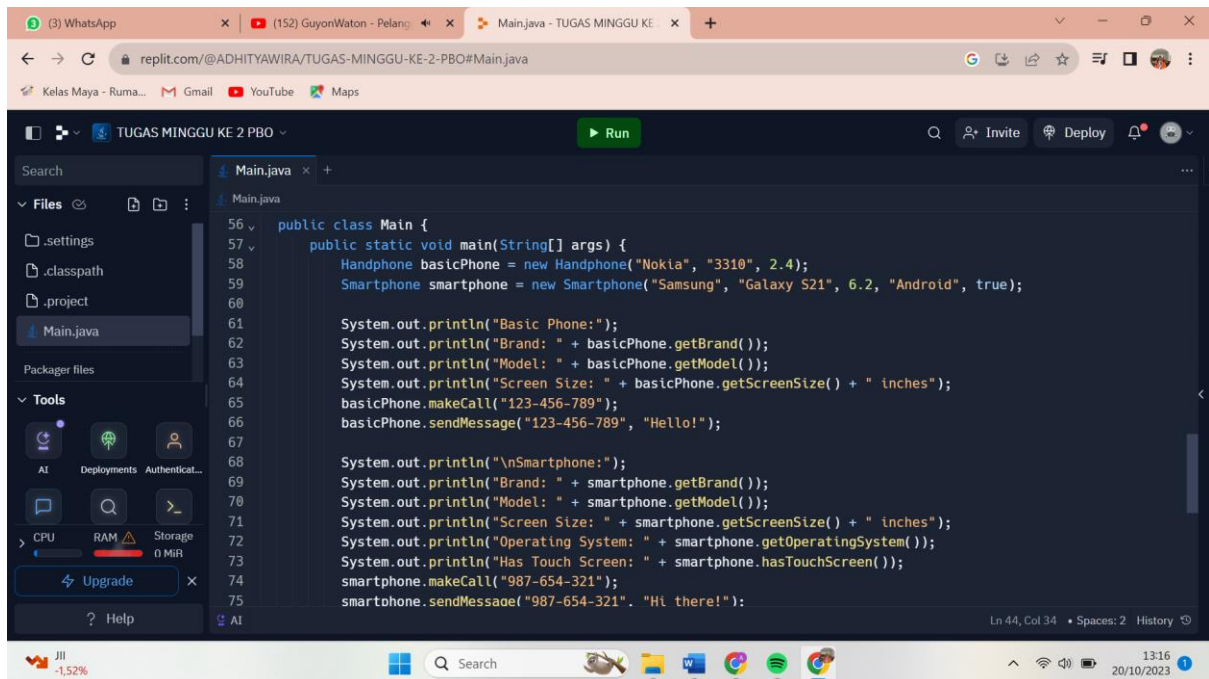
```
50  
51 ✓ public void installApp(String appName) {  
52     System.out.println("Installing " + appName + " on " + getBrand() + " " + getModel());  
53 }  
54 }
```

Ln 41, Col 8 • Spaces: 2 History

➤ Metode `installApp(String appName)`:

- Ini adalah metode publik yang digunakan untuk mensimulasikan proses menginstal sebuah aplikasi (app) pada smartphone. Metode ini menerima satu parameter, yaitu `appName`, yang merupakan nama aplikasi yang akan diinstal.
- Metode ini mencetak pesan ke konsol yang berisi informasi tentang aplikasi yang diinstal, merek dan model smartphone. Pesan ini dibuat dengan menggabungkan beberapa informasi sebagai berikut:
  - "Installing ": Pesan awal yang mengindikasikan bahwa sebuah aplikasi sedang diinstal.
  - `appName`: Nama aplikasi yang diteruskan sebagai parameter.
  - " on ": Pesan yang mengindikasikan bahwa aplikasi diinstal pada.
  - `getBrand()`: Metode yang mengambil merek (brand) smartphone dari kelas `Handphone` (kelas induk).
  - `getModel()`: Metode yang mengambil model smartphone dari kelas `Handphone` (kelas induk).

### 3. PUBLIC CLASS ATAU KELAS UTAMA



```
56 public class Main {
57     public static void main(String[] args) {
58         Handphone basicPhone = new Handphone("Nokia", "3310", 2.4);
59         Smartphone smartphone = new Smartphone("Samsung", "Galaxy S21", 6.2, "Android", true);
60
61         System.out.println("Basic Phone:");
62         System.out.println("Brand: " + basicPhone.getBrand());
63         System.out.println("Model: " + basicPhone.getModel());
64         System.out.println("Screen Size: " + basicPhone.getScreenSize() + " inches");
65         basicPhone.makeCall("123-456-789");
66         basicPhone.sendMessage("123-456-789", "Hello!");
67
68         System.out.println("\nSmartphone:");
69         System.out.println("Brand: " + smartphone.getBrand());
70         System.out.println("Model: " + smartphone.getModel());
71         System.out.println("Screen Size: " + smartphone.getScreenSize() + " inches");
72         System.out.println("Operating System: " + smartphone.getOperatingSystem());
73         System.out.println("Has Touch Screen: " + smartphone.hasTouchScreen());
74         smartphone.makeCall("987-654-321");
75         smartphone.sendMessage("987-654-321", "Hi there!");
76     }
77 }
```

#### ➤ Program Utama (Main):

- Ini adalah kelas program utama yang memiliki metode main sebagai titik masuk eksekusi program.

#### ➤ Membuat Objek Handphone dan Smartphone:

- Program ini membuat dua objek, yaitu basicPhone dari kelas Handphone dan smartphone dari kelas Smartphone. Ini dilakukan dengan menginisialisasi objek-objek tersebut dengan data yang diperlukan seperti merek, model, ukuran layar, sistem operasi, dan kemampuan layar sentuh.

#### ➤ Mengakses Informasi tentang Ponsel:

- Program ini mencetak informasi tentang objek basicPhone dan smartphone dengan menggunakan metode-metode yang telah didefinisikan dalam kelas-kelas tersebut. Misalnya, program mencetak merek, model, ukuran layar, sistem operasi, dan apakah layar sentuh (touchscreen) tersedia atau tidak.

#### ➤ Menjalankan Tindakan Ponsel:

- Program ini juga menjalankan beberapa tindakan yang dapat dilakukan oleh ponsel, seperti membuat panggilan telepon, mengirim pesan, dan menginstal aplikasi:
  - Untuk basicPhone, program menggunakan metode makeCall() untuk mensimulasikan proses membuat panggilan dan metode sendMessage() untuk mengirim pesan teks.

- Untuk smartphone, program menggunakan metode makeCall(), sendMessage(), dan installApp() sesuai dengan fungsionalitas yang telah didefinisikan dalam kelas Smartphone.

## ❖ OUTPUTNYA

```

Basic Phone:
Brand: Nokia
Model: 3310
Screen Size: 2.4 inches
Making a call to 123-456-789
Sending a message to 123-456-789: Hello!

Smartphone:
Brand: Samsung
Model: Galaxy S21
Screen Size: 6.2 inches
Operating System: Android
Has Touch Screen: true
Making a call to 987-654-321
Sending a message to 987-654-321: Hi there!
Installing Facebook on Samsung Galaxy S21
  
```

### 1. Bagian "Basic Phone":

- Brand: Nokia
  - Merek ponsel basicPhone adalah "Nokia".
- Model: 3310
  - Model ponsel basicPhone adalah "3310".
- Screen Size: 2.4 inches
  - Ukuran layar ponsel basicPhone adalah 2.4 inci.
- Making a call to 123-456-789
  - Pesan ini menunjukkan bahwa sebuah panggilan dibuat dengan nomor "123-456-789".
- Sending a message to 123-456-789: Hello!
  - Pesan ini menunjukkan bahwa pesan dengan isi "Hello!" dikirim ke nomor "123-456-789".

### 2. Bagian "Smartphone":

- Brand: Samsung
  - Merek ponsel smartphone adalah "Samsung".
- Model: Galaxy S21
  - Model ponsel smartphone adalah "Galaxy S21".
- Screen Size: 6.2 inches
  - Ukuran layar ponsel smartphone adalah 6.2 inci.
- Operating System: Android
  - Smartphone ini menjalankan sistem operasi "Android".
- Has Touch Screen: true
  - Smartphone ini memiliki layar sentuh (touchscreen) berdasarkan nilai boolean "true".
- Making a call to 987-654-321
  - Pesan ini menunjukkan bahwa sebuah panggilan dibuat dengan nomor "987-654-321".
- Sending a message to 987-654-321: Hi there!
  - Pesan ini menunjukkan bahwa pesan dengan isi "Hi there!" dikirim ke nomor "987-654-321".
- Installing Facebook on Samsung Galaxy S21
  - Pesan ini menunjukkan bahwa aplikasi "Facebook" diinstal pada smartphone "Samsung Galaxy S21". Pesan ini dihasilkan oleh pemanggilan metode `installApp()` pada objek smartphone.

Hasil output ini mencerminkan penggunaan atribut dan metode dalam kelas Handphone dan Smartphone, serta interaksi dengan objek-objek yang telah diinisialisasi dalam program utama.