

LAPORAN MINGGU KE 3

Penerapan Konsep Inheritance pada Sistem Informasi Produk Pertanian (Agri-POS)

NAMA : Hilda Sava Akzena

KELAS : 3 ikra

NIM : 240202865

TUJUAN


1. Mahasiswa memahami konsep **inheritance (pewarisan class)** dalam pemrograman berorientasi objek (OOP).
2. Mahasiswa mampu membuat **superclass dan subclass** yang merepresentasikan kategori produk pertanian.
3. Mahasiswa dapat menerapkan **pemanggilan konstruktor dan method superclass** menggunakan kata kunci **super**.
4. Mahasiswa mampu mendemonstrasikan **hierarki class** dalam program yang terstruktur dan reusable.
5. Mahasiswa dapat menjelaskan **perbedaan penggunaan inheritance** dibandingkan dengan penggunaan class tunggal pada sistem informasi produk.

DASAR TEORI

1. **Inheritance** adalah konsep dalam OOP yang memungkinkan suatu class mewarisi atribut dan method dari class lain.
2. **Superclass** adalah class induk yang berisi atribut dan method umum.
Subclass adalah class turunan yang mewarisi dari superclass dan bisa menambah fitur baru.
3. Kata kunci **extends** digunakan untuk membuat hubungan antara superclass dan subclass.
4. Kata kunci **super** digunakan untuk memanggil konstruktor atau method dari superclass.
5. Dengan inheritance, program menjadi lebih **terstruktur, efisien, dan mudah dikembangkan**.
6. Dalam sistem **Agri-POS**, inheritance digunakan untuk membuat class:
 - o Produk sebagai superclass,
 - o Benih, Pupuk, dan AlatPertanian sebagai subclass yang memiliki atribut khusus masing-masing.

LANGKAH PRAKTIKUM

1. Membuat Superclass Produk

- Buat folder dan package sesuai struktur

- Di dalam package model, buat file Produk.java.
- Class ini berfungsi sebagai **superclass (class induk)** yang menyimpan atribut umum untuk semua jenis produk pertanian.
- Tambahkan atribut: kode (String), nama (String), harga (double), stok (int)

```
public Produk(String kode, String nama, double harga, int stok) {  
    this.kode = kode;  
    this.nama = nama;  
    this.harga = harga;  
    this.stok = stok;  
}
```

2. Membuat Subclass dari Produk

Buat tiga class turunan (subclass) di package model, yaitu Benih, Pupuk, dan AlatPertanian.

a. Benih.java

- Tambahkan atribut baru varietas.

```
public class Benih extends Produk {  
    private String varietas;
```

- Gunakan super() untuk memanggil konstruktor dari Produk.

```
    super(kode, nama, harga, stok);
```

b. Pupuk.java

- Tambahkan atribut jenis.

```
    private String jenis;
```

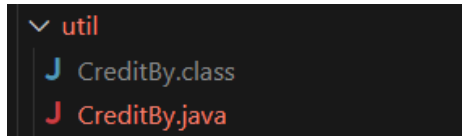
c. AlatPertanian.java

- Tambahkan atribut material.

```
    private String material;
```

3. Membuat Class CreditBy

- Buat folder dalam package util sesuai struktur



- Class ini digunakan untuk menampilkan identitas mahasiswa.

```
public static void print() {  
    System.out.println(x: "\nCredit By: 240202865 - hilda");  
}
```

4. Membuat Class Utama MainInheritance

- Class ini digunakan untuk menjalankan program dan menguji konsep inheritance.

5. Melakukan komplikasi dan menjalankan program

- Jalankan perintah berikut di terminal

```
PS C:\Users\Hilda\OneDrive\Documents\Praktikum\oop-202501-240202865\praktikum\week3-inheritance\src\main\java>  
javac com/upb/agripos/util/CreditBy.java  
>>  
PS C:\Users\Hilda\OneDrive\Documents\Praktikum\oop-202501-240202865\praktikum\week3-inheritance\src\main\java>  
javac com/upb/agripos/model/*.java  
>>  
PS C:\Users\Hilda\OneDrive\Documents\Praktikum\oop-202501-240202865\praktikum\week3-inheritance\src\main\java>  
javac com/upb/agripos/MainInheritance.java  
PS C:\Users\Hilda\OneDrive\Documents\Praktikum\oop-202501-240202865\praktikum\week3-inheritance\src\main\java>  
java com.upb.agripos.MainInheritance
```

KODE PROGRAM

a. Benih.java

```
praktikum > week3-inheritance > src > main > java > com > upb > agripos > model > J Benih.java > ...  
1 package com.upb.agripos.model;  
2  
3 public class Benih extends Produk {  
4     private String varietas;  
5  
6     public Benih(String kode, String nama, double harga, int stok, String varietas) {  
7         super(kode, nama, harga, stok);  
8         this.varietas = varietas;  
9     }  
10  
11     public void deskripsi() {  
12         System.out.println("Kode: " + kode + " | Nama: " + nama +  
13             " | Harga: Rp" + harga + " | Stok: " + stok +  
14             " | Varietas: " + varietas);  
15     }  
16 }  
17
```

b. Pupuk.java

```
praktikum > week3-inheritance > src > main > java > com > upb > agripas > model > J Pupuk.java > ...
1  package com.upb.agripas.model;
2
3  public class Pupuk extends Produk {
4      private String jenis;
5
6      public Pupuk(String kode, String nama, double harga, int stok, String jenis) {
7          super(kode, nama, harga, stok);
8          this.jenis = jenis;
9      }
10
11     public void deskripsi() {
12         System.out.println("Kode: " + kode + " | Nama: " + nama +
13             " | Harga: Rp" + harga + " | Stok: " + stok +
14             " | Jenis: " + jenis);
15     }
16 }
17
```

c. AlatPertanian.java

```
praktikum > week3-inheritance > src > main > java > com > upb > agripas > model > J AlatPertanian.java > ...
3  public class AlatPertanian extends Produk {
4
5
6      public AlatPertanian(String kode, String nama, double harga, int stok, String material) {
7          super(kode, nama, harga, stok);
8          this.material = material;
9      }
10
11     public void deskripsi() {
12         System.out.println("Kode: " + kode + " | Nama: " + nama +
13             " | Harga: Rp" + harga + " | Stok: " + stok +
14             " | Material: " + material);
15     }
16 }
17
18
```

d. CreditBy.java

```
praktikum > week3-inheritance > src > main > java > com > upb > agripas > util > J CreditBy.java
1  package com.upb.agripas.util;
2
3  public class CreditBy {
4      public static void print() {
5          System.out.println(x:"\nCredit By: 240202865 - hilda");
6      }
7  }
8
```

e.MainInheritance.java

```
praktikum > week3-inheritance > src > main > java > com > upb > agripos > MainInheritance.java > MainInheritance > main(String[] args)
1  package com.upb.agripos;
2
3  import com.upb.agripos.model.*;
4  import com.upb.agripos.util.CreditBy;
5
6  public class MainInheritance {
7      Run | Debug
8      public static void main(String[] args) {
9          Benih b = new Benih("BNH-001", "Benih Coklat 50", 90000.0, 70, "50");
10         Pupuk p = new Pupuk("PPK-101", "Pupuk Kandang 25kg", 70000.0, 50, "Organik");
11         AlatPertanian a = new AlatPertanian("ALT-501", "Penyiram Tanaman", 40000.0, 20, "Plastik");
12
13         System.out.println(x:"=== Data Produk Pertanian ===");
14         b.deskripsi();
15         p.deskripsi();
16         a.deskripsi();
17
18         System.out.println(x:"\n=== Menambah Stok Produk ===");
19         System.out.println(x:"Menambah stok Benih Coklat 50 sebanyak 25");
20         b.tambahStok(25);
21         b.deskripsi();
22
23         System.out.println(x:"\n=== Mengurangi Stok Produk ===");
24         System.out.println(x:"Mengurangi stok Penyiram Tanaman sebanyak 10");
25         a.kurangiStok(10);
26         a.deskripsi();
27
28         CreditBy.print();
29     }
30 }
31
```

HASIL EKSEKUSI

```
PS C:\Users\Hilda\OneDrive\Documents\Praktikum\oop-202501-240202865\praktikum\week3-inheritance\src\main\java>
java com.upb.agripos.MainInheritance
=== Data Produk Pertanian ===
Kode: BNH-001 | Nama: Benih Coklat 50 | Harga: Rp90000.0 | Stok: 70 | Varietas: 50
Kode: PPK-101 | Nama: Pupuk Kandang 25kg | Harga: Rp70000.0 | Stok: 50 | Jenis: Organik
Kode: ALT-501 | Nama: Penyiram Tanaman | Harga: Rp40000.0 | Stok: 20 | Material: Plastik

=== Menambah Stok Produk ===
Menambah stok Benih Coklat 50 sebanyak 25
Kode: BNH-001 | Nama: Benih Coklat 50 | Harga: Rp90000.0 | Stok: 90 | Varietas: 50

=== Mengurangi Stok Produk ===
Mengurangi stok Penyiram Tanaman sebanyak 10
Kode: ALT-501 | Nama: Penyiram Tanaman | Harga: Rp40000.0 | Stok: 10 | Material: Plastik

Credit By: 240202865 - hilda
PS C:\Users\Hilda\OneDrive\Documents\Praktikum\oop-202501-240202865\praktikum\week3-inheritance\src\main\java>
```

ANALISIS

1. Jalannya Program

- Program dimulai dari class MainInheritance.
- Di dalamnya dibuat tiga objek dari subclass yaitu Benih, Pupuk, dan AlatPertanian.
- Setiap subclass menggunakan konstruktor super() untuk memanggil atribut dari superclass Produk.
- Setelah objek dibuat, program menampilkan data masing-masing produk dengan memanggil method getName(), getVarietas(), getJenis(), dan getMaterial().
- Terakhir, class CreditBy dipanggil untuk menampilkan identitas mahasiswa.
- Semua proses ini menunjukkan bahwa subclass dapat mewarisi atribut dan method dari superclass tanpa perlu menulis ulang kode yang sama.

2. Perbedaan dengan Minggu Sebelumnya

- Pada minggu sebelumnya (Bab 2), program hanya menggunakan class tunggal, yaitu Produk, untuk semua jenis produk.
- Pada minggu ini, digunakan konsep inheritance untuk membedakan kategori produk melalui subclass (Benih, Pupuk, dan AlatPertanian).
- Dengan inheritance, struktur kode menjadi lebih terorganisir, efisien, dan mudah dikembangkan, karena setiap jenis produk bisa memiliki atribut dan perilaku khusus tanpa mengubah class utama.

3. Kendala dan Cara Mengatasinya

- Kendala: Awalnya program error karena lupa menambahkan super() di konstruktor subclass.
Solusi: Menambahkan pemanggilan super(kode, nama, harga, stok) di konstruktor setiap subclass.

KESIMPULAN

Dengan menerapkan konsep **inheritance**, program menjadi lebih terstruktur dan mudah dikembangkan. Subclass seperti `Benih`, `Pupuk`, dan `AlatPertanian` dapat mewarisi atribut dari `Produk`, sehingga kode lebih efisien dan tidak berulang.

QUIZ

1. Apa keuntungan menggunakan inheritance dibanding membuat class terpisah tanpa hubungan?

Jawaban: Kode menjadi lebih efisien, tidak berulang, dan lebih mudah dikembangkan karena subclass bisa mewarisi atribut dan method dari superclass.

2. Bagaimana cara subclass memanggil konstruktor superclass?

Jawaban: Dengan menggunakan keyword **`super`** () di dalam konstruktor subclass.

3. Berikan contoh kasus di POS pertanian selain Benih, Pupuk, dan Alat Pertanian yang bisa dijadikan subclass.

Jawaban: Contohnya **Pestisida**, **Bibit Ikan**, atau **Pakan Ternak** yang semuanya dapat mewarisi atribut umum dari class **Produk**.

CHECKLIST KEBERHASILAN

- Superclass `Produk` digunakan kembali tanpa duplikasi kode.
- Subclass `Benih`, `Pupuk`, dan `AlatPertanian` berhasil dibuat dengan atribut tambahan.
- Program berjalan menampilkan objek dari setiap subclass.
- `CreditBy` ditampilkan dengan benar.
- Commit sesuai instruksi.
- Laporan singkat lengkap dengan analisis.