

PERTEMUAN KESEMBILAN

Java Packages, API, Enum

Prepared by Adi Wahyu Pribadi

Pendahuluan

Package atau paket di Java digunakan untuk mengelompokkan class-class yang terkait. Package di Java terbagi menjadi dua macam:

- Package Built-in (Paket dari Java API)
- Package buatan sendiri

Built-in Package

Java API adalah library kumpulan Class yang sudah ditulis dan dapat langsung digunakan secara bebas. Java API terdapat dalam Java Development Environment. Library berisi komponen untuk mengelola input, pemrograman basis data, dan banyak lagi lainnya. Daftar lengkapnya dapat ditemukan di situs web Oracles:

<https://docs.oracle.com/javase/8/docs/api/>

<https://docs.oracle.com/en/java/javase/11/docs/api/index.html>

Library terbagi atas packages dan class. Berarti kita dapat mengimpor satu class (beserta metode dan atributnya), atau seluruh package yang berisi semua class yang ada di dalam package yang kita pilih.

Untuk menggunakan sebuah class atau package dari library, kita gunakan keyword **import**

Syntax

```
import package.name.Class;    // Import sebuah class
import package.name.*;        // Import seluruh package
```

Import sebuah class

```
import java.util.Scanner;
```

Pada baris di atas, nama package-nya adalah java.util dan nama class-nya adalah Scanner. Untuk menggunakan buat objek class dan gunakan salah satu metode yang tersedia yang ditemukan dalam dokumentasi class Scanner. Kita akan menggunakan method `nextLine()`, yang digunakan untuk membaca baris lengkap:

```
import java.util.Scanner;
class MyClass {
    public static void main(String[] args) {
        Scanner myObj = new Scanner(System.in);
        System.out.println("Masukkan nama Anda");

        String userName = myObj.nextLine();
        System.out.println("Nama Anda adalah " + userName);
    }
}
```

Import Sebuah Package

Ada banyak paket untuk dipilih. Pada contoh sebelumnya, kita menggunakan class Scanner dari package java.util. Package ini juga berisi fungsi utilitas seperti tanggal dan waktu, pembuat nomor acak, dan class utilitas lainnya.

Untuk mengimpor seluruh package, akhiri kalimat dengan tanda bintang (*). Contoh berikut akan mengimpor semua class di package java.util:

```
import java.util.*;
```

Package Buatan Sendiri

Untuk membuat package sendiri, Java menggunakan sistem folder file untuk menyimpan, sama seperti folder di komputer. Misalkan:

```
|__ src
    |__ App
        |__ ClassSaya.java
```

Untuk menggunakan sebuah package gunakan keyword package;

```
package App;
class ClassSaya {
    public static void main(String[] args) {
        System.out.println("Ini sebuah package!");
    }
}
```

Simpan file di atas dengan nama ClassSaya.java di dalam folder src

Compile File ClassSaya.java dengan perintah

```
C:\Users\Adiwa\src> javac ClassSaya.java
```

Menjalankan file ClassSaya.class

```
C:\Users\Adiwa\src>java ClassSaya
Error: Could not find or load main class ClassSaya
Caused by: java.lang.NoClassDefFoundError: App/ClassSaya (wrong name:
ClassSaya)
```

Muncul Error dikarenakan ClassSaya terdapat package App

Maka gunakan perintah javac -d . ClassSaya.java

```
C:\Users\Adiwa\src>javac -d . ClassSaya.java
```

Dan untuk menjalankan ClassSaya adalah dengan perintah java NamaPackage>NamaClass

```
C:\Users\Adiwa\src>java App.ClassSaya  
Ini sebuah package!
```

Java Enums

Enum adalah Class spesial yang merepresentasikan grup KONSTAN/CONSTANTS (variable yang tidak bisa diubah, seperti variabel final). Untuk membuath sebuah enum, gunakan keyword enum, dan pisahkan constant dengan tanda koma (.). Ingat, constant ditulis dengan HURUF BESAR.

```
enum Level {  
    LOW, MEDIUM, HIGH  
}
```

Untuk mengakses konstan di enum gunakan titik / dot syntax:

```
Level kemahiran = Level.MEDIUM;  
Level rasaPedas = Level.MEDIUM;  
Level rasaPedas = Level.HIGH;
```

Enum di dalam sebuah Class

```
Public class ClassKu {  
    enum Level {  
        LOW, MEDIUM, HIGH  
    }  
    Public static void main(String[] args) {  
        Level kemahiran = Level.HIGH;  
        System.out.println(kemahiran);  
    }  
}
```

Outputnya:

```
HIGH
```

Enum di dalam Statement Switch

```
enum Level {  
    LOW, MEDIUM, HIGH  
}  
  
public class ClassKu {  
    public static void main(String[] args) {  
        Level kemahiran = Level.MEDIUM;  
  
        switch(kemahiran) {  
            case LOW:  
                System.out.println("Anak bawang");  
                break;  
            case MEDIUM:  
                System.out.println("Anggota");  
                break;  
            case HIGH:  
                System.out.println("Suhu");  
                break;  
        }  
    }  
}
```

Outputnya:

Anggota

Looping di dalam Enum

Tipe enum memiliki sebuah method bernama `values()`, method tersebut akan mengembalikan array dari seluruh constant yang ada di dalam enum tersebut. Method ini bermanfaat ketika ingin menampilkan seluruh constant yang ada di enum.

```
for (Level kemahiran : Level.values()) {  
    System.out.println(kemahiran);  
}
```

Outputnya:

```
LOW  
MEDIUM  
HIGH
```