

## RESPONSI PBO

NAMA : FIKRY MUMTAZ PRATAMA  
NIM : H1D024106  
SHIFT : SHIFT B  
TANGGAL : 07/12/2025

### JAWAB

#### 1. Interface KaryawanKontrak

```
public interface KaryawanKontrak {  
    double hitungGaji(int jamKerja);  
    void perpanjangKontrak(int durasiBulan);  
    default String getStatusCuti() {  
        return "Tersedia 12 hari";  
    }  
}
```

#### Penjelasan

Interface ini mendefinisikan aturan dasar yang harus dimiliki semua karyawan kontrak. Dua method abstract wajib diimplementasikan oleh setiap class yang menggunakannya, yaitu `hitungGaji` dan `perpanjangKontrak`. Sementara itu, default method `getStatusCuti` memberikan nilai standar yang bisa langsung digunakan atau diganti oleh class yang mengimplementasikannya. Dengan interface ini, semua karyawan kontrak dijamin memiliki kemampuan menghitung gaji dan memperpanjang kontrak.

#### 2. Interface AksesSistem

```
public interface AksesSistem {  
    void login(String pin);  
    void logout();  
    default String getRoleAkses() {  
        return "Staff Biasa";  
    }  
}
```

#### Penjelasan

Interface ini mengatur proses login dan logout pengguna pada sistem internal. Dua method abstract harus diisi oleh class yang menggunakannya. Default method `getRoleAkses` menyediakan role standar bagi pengguna sistem, tetapi dapat diubah sesuai kebutuhan. Interface ini memastikan bahwa siapa pun yang memiliki akses sistem wajib mempunyai mekanisme login dan logout.

### 3. Class ProgrammerMagang

public class ProgrammerMagang implements KaryawanKontrak, AksesSistem {

private String nama;

private double gajiPerJam;

private String pinRahasia;

private boolean sedangLogin;

public ProgrammerMagang(String nama, double gajiPerJam, String pinRahasia) {

    this.nama = nama;

    this.gajiPerJam = gajiPerJam;

    this.pinRahasia = pinRahasia;

    this.sedangLogin = false;

}

@Override

public double hitungGaji(int jamKerja) {

    double total = jamKerja \* gajiPerJam;

    System.out.println("Total gaji: Rp " + total);

    return total;

}

@Override

public void perpanjangKontrak(int durasiBulan) {

    System.out.println("Kontrak diperpanjang selama " + durasiBulan + " bulan.");

}

@Override

public String getStatusCuti() {

    return "Tersedia 5 hari";

```
}
```

```
@Override
```

```
public void login(String pin) {  
    if (pin.equals(pinRahasia)) {  
        sedangLogin = true;  
        System.out.println("Login berhasil.");  
    } else {  
        System.out.println("PIN salah. Login gagal.");  
    }  
}
```

```
@Override
```

```
public void logout() {  
    sedangLogin = false;  
    System.out.println("Logout berhasil.");  
}
```

```
@Override
```

```
public String getRoleAkses() {  
    return "Magang IT";  
}  
}
```

#### Penjelasan

Class ini menggabungkan dua interface sehingga programmer magang memiliki peran sebagai karyawan kontrak dan sekaligus pengguna sistem. Atribut nama, gaji per jam, dan PIN digunakan untuk menyimpan data diri serta keamanan login. Atribut sedangLogin digunakan untuk mengetahui apakah pengguna sedang aktif dalam sistem.

Method dari KaryawanKontrak mengatur perhitungan gaji dan perpanjangan kontrak, sementara method default getStatusCuti diganti sehingga programmer magang hanya mendapatkan lima hari cuti. Method dari AksesSistem mengatur proses login berdasarkan kecocokan PIN, proses logout, serta pemberian role akses khusus yaitu Magang IT.

#### 4. Class UjiSDM (Main)

```
public class UjiSDM {  
    public static void main(String[] args) {  
  
        ProgrammerMagang magang = new ProgrammerMagang("Andi", 50000.0, "1234");  
  
        magang.hitungGaji(160);  
  
        System.out.println("Status cuti: " + magang.getStatusCuti());  
  
        magang.login("9999");  
  
        magang.login("1234");  
  
        System.out.println("Role akses: " + magang.getRoleAkses());  
  
        magang.perpanjangKontrak(6);  
  
        magang.logout();  
    }  
}
```

#### Penjelasan

Program dimulai dengan membuat objek bernama Andi dengan gaji lima puluh ribu per jam dan PIN 1234. Program lalu menghitung gaji berdasarkan 160 jam kerja dan menampilkan jumlah cuti yang tersedia. Setelah itu, program menguji login menggunakan PIN yang salah sehingga akses ditolak, kemudian mencoba lagi dengan PIN benar sehingga login berhasil. Program kemudian menampilkan role akses yang diberikan kepada programmer magang. Setelah login sukses, program memperpanjang kontrak selama enam bulan dan akhirnya menutup sesi akses dengan logout. Alur ini menunjukkan bagaimana kedua interface bekerja secara bersamaan dalam satu objek.

```
ProgrammerMagang.java  AksesSistem.java  Pengayaan.txt  UjiSDM.java  KaryawanKontrak.java x
1 public interface KaryawanKontrak { 1 usage 1 implementation
2     double hitungGaji(int jamKerja); no usages 1 implementation
3     void perpanjangKontrak(int durasiBulan); no usages 1 implementation
4
5     default String getStatusCuti() { no usages 1 override
6         return "Tersedia 12 hari";
7     }
8 }
9
```

```
ProgrammerMagang.java  AksesSistem.java x  Pengayaan.txt  UjiSDM.java  KaryawanKontrak.java
1 public interface AksesSistem { 1 usage 1 implementation
2     void login(String pin); no usages 1 implementation
3     void logout(); no usages 1 implementation
4
5     default String getRoleAkses() { no usages 1 override
6         return "Staff Biasa";
7     }
8 }
9
```

```
ProgrammerMagang.java x  AksesSistem.java  Pengayaan.txt  UjiSDM.java  KaryawanKontrak.java
1 public class ProgrammerMagang implements KaryawanKontrak, AksesSistem { no usages
26
27     @Override no usages
28     public String getStatusCuti() {
29         return "Tersedia 5 hari";
30     }
31
32     @Override no usages
33     public void login(String pin) {
34         if (pin.equals(pinRahasia)) {
35             sedangLogin = true;
36             System.out.println("Login berhasil.");
37         } else {
38             System.out.println("PIN salah. Login gagal.");
39         }
40     }
41
42     @Override no usages
43     public void logout() {
44         sedangLogin = false;
45         System.out.println("Logout berhasil.");
46     }
47
48     @Override no usages
49     public String getRoleAkses() {
50         return "Magang IT";
51     }
52 }
```

```
ProgrammerMagang.java x  AksesSistem.java  Pengayaan.txt  UjiSDM.java  KaryawanKontrak.java

1 public class ProgrammerMagang implements KaryawanKontrak, AksesSistem { no usages
2
3     private String nama; 1 usage
4     private double gajiPerJam; 2 usages
5     private String pinRahasia; 2 usages
6     private boolean sedangLogin; 3 usages
7
8     public ProgrammerMagang(String nama, double gajiPerJam, String pinRahasia) { no usages
9         this.nama = nama;
10        this.gajiPerJam = gajiPerJam;
11        this.pinRahasia = pinRahasia;
12        this.sedangLogin = false;
13    }
14
15    @Override no usages
16    public double hitungGaji(int jamKerja) {
17        double total = jamKerja * gajiPerJam;
18        System.out.println("Total gaji: Rp " + total);
19        return total;
20    }
21
22    @Override no usages
23    public void perpanjangKontrak(int durasiBulan) {
24        System.out.println("Kontrak diperpanjang selama " + durasiBulan + " bulan.");
25    }
26
27    @Override no usages
28    public String getStatusCuti() {
```

```

1 public class UjiSDM {
2     public static void main(String[] args) {
3
4         ProgrammerMagang pm = new ProgrammerMagang( nama: "Andi", gajiPerJam: 50000.0, pinRahasia: "1234");
5
6         pm.hitungGaji( jamKerja: 160);
7
8         System.out.println(pm.getStatusCuti());
9
10        pm.login( pin: "9999");
11
12        pm.login( pin: "1234");
13
14        System.out.println(pm.getRoleAkses());
15
16        pm.perpanjangKontrak( durasiBulan: 6);
17
18        pm.logout();
19    }
20 }

```

```
Run UjISDM x
C:\Program Files\Eclipse Adoptium\jdk-21.0.7-hotspot\bin\java.exe " -javaagent:C:\Program Files\JetBrains\IntelliJ IDEA 2025.2.1\lib\idea_rt.jar=64439 -Dfile.encoding=UTF-8
Total gaji: Rp 8000000.0
Tersedia 5 hari
PIN salah. Login gagal.
Login berhasil.
Magang IT
Kontrak diperpanjang selama 6 bulan.
Logout berhasil.

Process finished with exit code 0
```