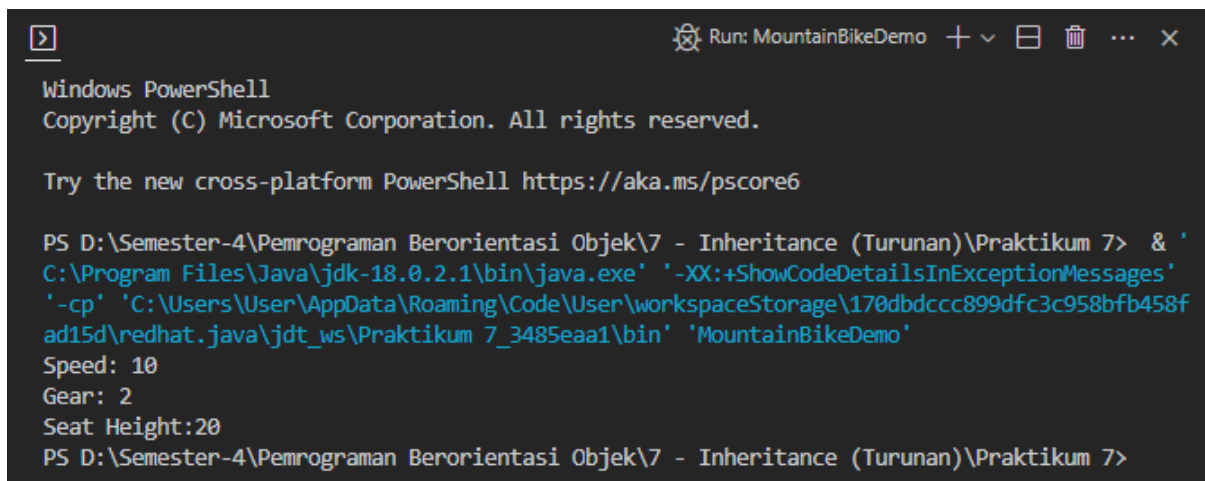# Dokumentasi Praktikum PBO 7

Mata Kuliah   :   PBO - TI - S1

Pertemuan   :   7

NIM   :   A11.2021.13254

Nama   :   Yohanes Dimas Pratama

Contoh Program Inheritance

Hasil Program:

Code Program:

*Bicycle.java

```java
public class Bicycle {
    int speed = 0;
    int gear = 0;

    void changeGear(int newValue) {
        gear = gear + newValue;
        System.out.println("Gear: " + gear);
    }

    void speedUp(int increment) {
        speed = speed + increment;
        System.out.println("Speed: " + speed);
    }
}

class MountainBike extends Bicycle {
    int seatHeight;
```
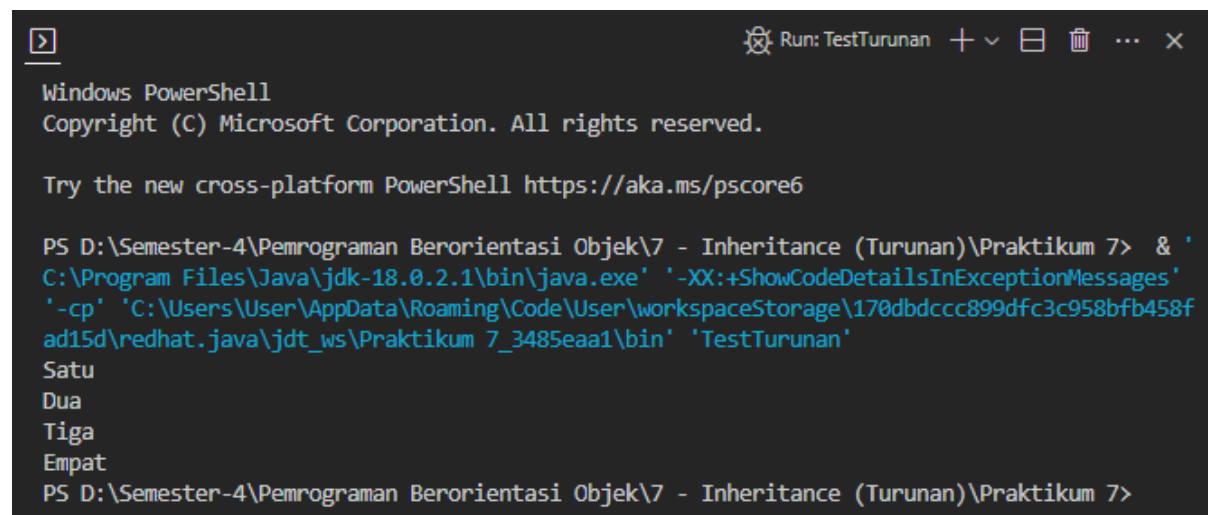
```java
    public void setHeight(int newValue) {
        seatHeight = newValue;
        System.out.println("Seat Height:" +seatHeight);
    }
}

class MountainBikeDemo {
    public static void main(String[] args) {
        MountainBike mbike = new MountainBike();
        mbike.speedUp(10);
        mbike.changeGear(2);
        mbike.setHeight(20);
    }
}
```

Contoh Program Inheritance 2

Hasil Program:

Code Program:

*TestTurunan.java

```java
class Satu {
    Satu() {
        System.out.println("Satu");
    }
}

class Dua extends Satu {
    Dua() {
        System.out.println("Dua");
    }
}
```

```java
class Tiga extends Dua {
    Tiga() {
        System.out.println("Tiga");
    }
}

class Empat extends Tiga {
    Empat() {
        System.out.println("Empat");
    }
}

public class TestTurunan {
    public static void main(String[] args) {
        new Empat();
    }
}
```

Contoh Program Inheritance 3

Hasil Program:

Code Program:

*TestTurunan2.java

```java
class Satu {
    Satu(String s) {
        System.out.println("Satu... " + s);
    }
}

class Dua extends Satu {
    Dua(String s) {
        super(s);
        System.out.println("Dua...");
    }
}
```

```java
class Tiga extends Dua {
    Tiga(String s) {
        super(s);
        System.out.println("Tiga...");
    }
}

class Empat extends Tiga {
    Empat(String s) {
        super(s);
        System.out.println("Empat...");
    }
}

public class TestTurunan2 {
    public static void main(String[] args) {
        new Empat("Kiriman untuk mbah buyut... alfatihah...");
    }
}
```
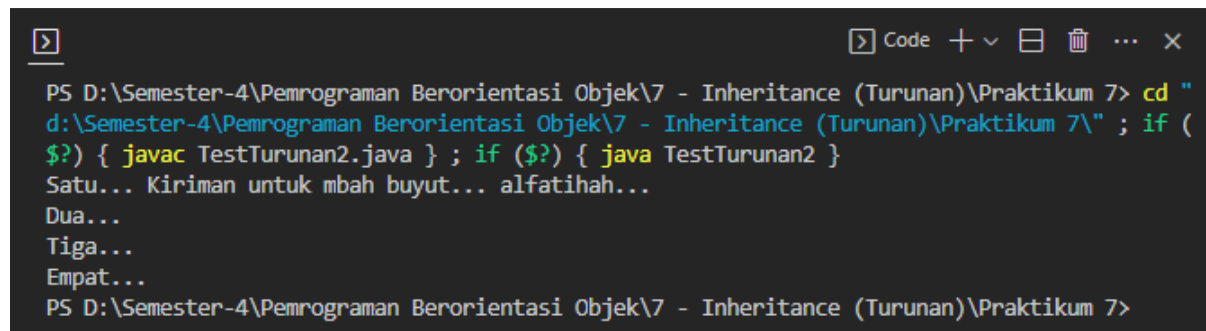
Latihan 1

Hasil Program:

```
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

Try the new cross-platform PowerShell https://aka.ms/pscore6

PS D:\Semester-4\Pemrograman Berorientasi Objek\7 - Inheritan
ce (Turunan)\Praktikum 7>  & 'C:\Program Files\Java\jdk-18.0.
2.1\bin\java.exe' '-XX:+ShowCodeDetailsInExceptionMessages' '
-cp' 'C:\Users\User\AppData\Roaming\Code\User\workspaceStorag
e\170dbdccc899dfc3c958bfb458fad15d\redhat.java\jdt_ws\Praktik
um 7_3485eaa1\bin' 'Mobil2BMWDemo'
Mesin BMW hidup...
Ganti gigi BMWsukses...
Mesin BMW mati...
TV dihidupkan
TV mencari channel
TV menampilkan gambar
PS D:\Semester-4\Pemrograman Berorientasi Objek\7 - Inheritan
ce (Turunan)\Praktikum 7>
```

Code Program:

*Mobil2BMW.java

```java
class Mobil2 {
```

```java
    int roda = 4;
    int body = 1;
    int mesin = 1;
    String nama;

    void maju() {
        System.out.println("Maju...");
    }
    void mundur() {
        System.out.println("Mundur...");
    }
    void belok() {
        System.out.println("Belok...");
    }

    void hidupkanMobil(String nama) {
        System.out.println("Mesin " + nama + " hidup...");
    }
    void matikanMobil(String nama) {
        System.out.println("Mesin " + nama + " mati...");
    }
    void gantiGigi(String nama) {
        System.out.println("Ganti gigi " + nama + "sukses...");
    }

    void setRoda(int roda) {
        this.roda = roda;
    }
    int getRoda() {
        return roda;
    }
    void setBody(int body) {
        this.body = body;
    }
    int getBody() {
        return body;
    }
}

class Mobil2BMW extends Mobil2 {
    void nontonTV() {
        System.out.println("TV dihidupkan");
        System.out.println("TV mencari channel");
        System.out.println("TV menampilkan gambar");
    }
}

class Mobil2BMWDemo {
```

```
    public static void main(String[] args) {
        Mobil2BMW bmw = new Mobil2BMW();
        bmw.hidupkanMobil("BMW");
        bmw.gantiGigi("BMW");
        bmw.matikanMobil("BMW");
        bmw.nontonTV();
    }
}
```

Latihan 2

Hasil Program:

```
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

Try the new cross-platform PowerShell https://aka.ms/pscore6

PS D:\Semester-4\Pemrograman Berorientasi Objek\7 - Inheritan
ce (Turunan)\Praktikum 7>  & 'C:\Program Files\Java\jdk-18.0.
2.1\bin\java.exe' '-XX:+ShowCodeDetailsInExceptionMessages' '
-cp' 'C:\Users\User\AppData\Roaming\Code\User\workspaceStorag
e\170dbdccc899dfc3c958bfb458fad15d\redhat.java\jdt_ws\Praktik
um 7_3485eaa1\bin' 'karyawanDemo'

Menu Pilihan Karyawan PT KIA

1. Karyawan Tetap
2. Karyawan Kontrak
3. Keluar

Masukkan pilihan: 1

Nama: Dimas
Jumlah Anak: 3

Karyawan Tetap
Nama: Dimas
Jumlah Anak: 3
Tunjangan Anak: 500000
Gaji Pokok: 5000000
Total Gaji: 6500000


Menu Pilihan Karyawan PT KIA

1. Karyawan Tetap
2. Karyawan Kontrak
3. Keluar

Masukkan pilihan: 2

Nama: Akbar
Masukkan jumlah hari kerja: 26

Karyawan Kontrak
Nama: Akbar
Jumlah Anak: 0
Tunjangan Anak: 500000
Upah Harian: 100000
Jumlah Hari Kerja: 26
Total Gaji: 2600000


Menu Pilihan Karyawan PT KIA

1. Karyawan Tetap
2. Karyawan Kontrak
3. Keluar

Masukkan pilihan:
```

Code Program:

*Karyawan.java

```java
import java.util.Scanner;

class Karyawan {
    String nama;
    int jumlahAnak;
    int tunjanganAnak = 500000;

    void isiData() {
        Scanner input = new Scanner(System.in);
        System.out.print("Nama: ");
        nama = input.nextLine();
        System.out.print("Jumlah Anak: ");
        jumlahAnak = input.nextInt();
    }
}

class KaryawanTetap extends Karyawan {
    int gajiPokok = 5000000;

    int totalGaji() {
        int totalGaji = gajiPokok + (tunjanganAnak * jumlahAnak);
        return totalGaji;
    }

    void tampilGaji() {
        System.out.println("Karyawan Tetap");
        System.out.println("Nama: " + nama);
        System.out.println("Jumlah Anak: " + jumlahAnak);
        System.out.println("Tunjangan Anak: " + tunjanganAnak);
        System.out.println("Gaji Pokok: " + gajiPokok);
        System.out.println("Total Gaji: " + totalGaji());
    }
}

class KaryawanKontrak extends Karyawan {
    int upahHarian = 100000;
    int jumlahHariKerja;

    void isiData() {
        Scanner input = new Scanner(System.in);
        System.out.print("Nama: ");
        nama = input.nextLine();
        System.out.print("Masukkan jumlah hari kerja: ");
        jumlahHariKerja = input.nextInt();
    }
```

```java
    int totalGaji() {
        int totalGaji = upahHarian * jumlahHariKerja;
        return totalGaji;
    }

    void tampilGaji() {
        System.out.println("Karyawan Kontrak");
        System.out.println("Nama: " + nama);
        System.out.println("Jumlah Anak: " + jumlahAnak);
        System.out.println("Tunjangan Anak: " + tunjanganAnak);
        System.out.println("Upah Harian: " + upahHarian);
        System.out.println("Jumlah Hari Kerja: " + jumlahHariKerja);
        System.out.println("Total Gaji: " + totalGaji());
    }
}

class karyawanDemo {
    public static void main(String[] args) {
        Scanner input = new Scanner(System.in);
        int menu;

        do {
            System.out.println("");
            System.out.println("Menu Pilihan Karyawan PT KIA");
            System.out.println("");
            System.out.println("1. Karyawan Tetap");
            System.out.println("2. Karyawan Kontrak");
            System.out.println("3. Keluar");
            System.out.println("");
            System.out.print("Masukkan pilihan: ");
            menu = input.nextInt();

            switch (menu) {
                case 1:
                    KaryawanTetap karyawanTetap = new KaryawanTetap();
                    System.out.println("");
                    karyawanTetap.isiData();
                    System.out.println("");
                    karyawanTetap.tampilGaji();
                    System.out.println("");
                    break;
                case 2:
                    KaryawanKontrak karyawanKontrak = new KaryawanKontrak();
                    System.out.println("");
                    karyawanKontrak.isiData();
                    System.out.println("");
                    karyawanKontrak.tampilGaji();
```

```java
                    System.out.println("");
                    break;
                case 3:
                    System.out.println("");
                    System.out.println("Terima kasih telah menggunakan program
ini.");
                    break;
                default:
                    System.out.println("");
                    System.out.println("Pilihan tidak tersedia.");
                    break;
            }
        }
        while (menu <= 4);
    }
}
```
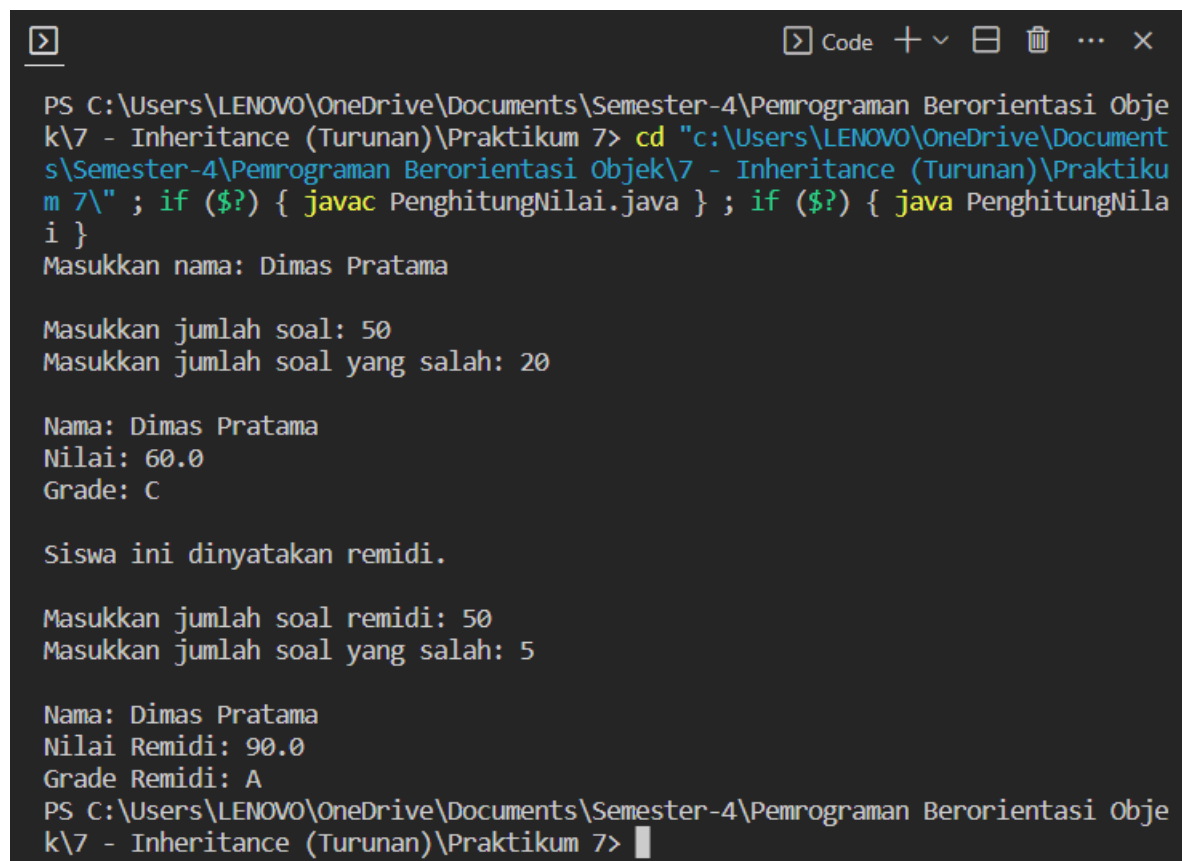
Latihan 3

Hasil Program:

Code Program:

*PenghitungNilai.java

```java
import java.util.Scanner;

class GradedActivity {
    double score;
    String name;

    public GradedActivity() {
    }

    public GradedActivity(String a) {
        this.name = a;
    }

    public void setScore(double b) {
        score = b * 100;
    }

    public double getScore() {
        return score;
    }

    public char getGrade(){
        if (score >= 85) {
            return 'A';
        }
        else if (score < 85 && score >= 75) {
            return 'B';
        }
        else if (score < 75 && score >= 60) {
            return 'C';
        }
        else if (score < 60 && score >= 50) {
            return 'D';
        }
        else if (score < 50) {
            return 'E';
        }
        else {
            return 0;
        }
    }
}

class FinalExam extends GradedActivity {
    int numberOfQuestions;
    int numberMissed;
    double pointsPerQuestion;
```

```java
import java.util.Scanner;

import javax.swing.plaf.basic.BasicBorders.MenuBarBorder;

class GradedActivity {
    double score;
    String name;

    public GradedActivity() {
    }

    public GradedActivity(String a) {
        this.name = a;
    }

    public void setScore(double b) {
        score = b * 100;
    }

    public double getScore() {
        return score;
    }

    public char getGrade(){
        if (score >= 85) {
            return 'A';
        }
        else if (score < 85 && score >= 75) {
            return 'B';
        }
        else if (score < 75 && score >= 60) {
            return 'C';
        }
        else if (score < 60 && score >= 50) {
            return 'D';
        }
        else if (score < 50) {
            return 'E';
        }
        else {
            return 0;
        }
    }
}

class FinalExam extends GradedActivity {
    int numberOfQuestions;
    int numberMissed;
```

```java
        double pointsPerQuestion;

    public FinalExam(int questions, int missed) {
        this.numberOfQuestions = questions;
        this.numberMissed = missed;
    }

    public double getPointsPerQuestion() {
        pointsPerQuestion = (double) (numberOfQuestions - numberMissed) /
(double) numberOfQuestions;
        return pointsPerQuestion;
    }

    public int getNumberMissed() {
        return numberMissed;
    }
}

class PassFailActivity extends GradedActivity {
    double minimumPassingScore;

    public PassFailActivity(double min) {
        this.minimumPassingScore = min;
    }

    public char getGrade(){
        if (getScore() >= 85) {
            return 'A';
        }
        else if (getScore() < 85 && getScore() >= 75) {
            return 'B';
        }
        else if (getScore() < 75 && getScore() >= 60) {
            return 'C';
        }
        else if (getScore() < 60 && getScore() >= 50) {
            return 'D';
        }
        else if (getScore() < 50) {
            return 'E';
        }
        else {
            return 0;
        }
    }
}

class PassFailExam extends PassFailActivity {
```

```java
    int numberOfQuestions;
    double pointsPerQuestion;
    int numberMissed;

    public PassFailExam(int questions, int missed, double minPassing) {
        super(minPassing);
        this.numberOfQuestions = questions;
        this.numberMissed = missed;
    }

    public double getPointsEach() {
        pointsPerQuestion = (double) (numberOfQuestions - numberMissed) /
(double) numberOfQuestions;
        return pointsPerQuestion;
    }

    public int getNumMissed() {
        return numberMissed;
    }
}

public class PenghitungNilai {
    public static void main(String[] args) {
        Scanner input = new Scanner(System.in);
        System.out.print("Masukkan nama: ");
        String nama = input.nextLine();
        System.out.println("");
        System.out.print("Masukkan jumlah soal: ");
        int soal = input.nextInt();
        System.out.print("Masukkan jumlah soal yang salah: ");
        int salah = input.nextInt();
        System.out.println("");

        FinalExam exam = new FinalExam(soal, salah);
        exam.setScore(exam.getPointsPerQuestion());

        System.out.println("Nama: " + nama);
        System.out.println("Nilai: " + exam.getScore());
        System.out.println("Grade: " + exam.getGrade());
        System.out.println("");

        if (exam.getScore() < 75) {
            System.out.println("Siswa ini dinyatakan remidi.");
            System.out.println("");
            System.out.print("Masukkan jumlah soal remidi: ");
            int soalRemidi = input.nextInt();
            System.out.print("Masukkan jumlah soal yang salah: ");
            int salahRemidi = input.nextInt();
```

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

            PassFailExam remidi = new PassFailExam(soalRemidi, salahRemidi,
75);
            remidi.setScore(remidi.getPointsEach());
            exam.setScore(exam.getScore() + remidi.getScore());

            System.out.println("Nama: " + nama);
            System.out.println("Nilai Remidi: " + remidi.getScore());
            System.out.println("Grade Remidi: " + remidi.getGrade());
            String menu = input.nextLine();
        }
        else {
            System.out.println("Siswa ini dinyatakan lulus.");
        }
    }
}
```