Nama : Muhammad Rasyad Trialgi

NIM : F1B020097

Kelompok : 4

**JOBSHEET P7**

|  |  |  |  |
| --- | --- | --- | --- |
| NO | Kegiatan | Script | Hasil Running |
| 1 | Buatlah program bebas menggunakan virtual method invocation | package P7;  class Animal {  public void makeSound() {  System.out.println("Suara Hewan");  }  }  class Anjing extends Animal {  @Override  public void makeSound() {  System.out.println(" Guguk");  }  }  class Kucing extends Animal {  @Override  public void makeSound() {  System.out.println(" Miauw");  }  }  public class Hewan {  public static void main(String[] args) {  Animal animal = new Animal();  Animal Anjing = new Anjing();  Animal Kucing = new Kucing();  System.out.println("Calling makeSound on Animal :");  animal.makeSound();  System.out.println("\nSuara Anjing :");  Anjing.makeSound();  System.out.println("\nSuara Kucing :");  Kucing.makeSound();  }  } |  |
| 2 | Buatlah program bebas atau modifikasi program disamping menggunakan Heterogeneous Collection | package P7;  import java.util.ArrayList;  class Animal {  public void makeSound() {  System.out.println("Animal makes a sound");  }  }  class Anjing extends Animal {  public void makeSound() {  System.out.println("Guguk");  }  }  class Kucing extends Animal {  public void makeSound() {  System.out.println("Miauw");  }  }  public class Hetero {  public static void main(String[] args) {  ArrayList<Object> animals = new ArrayList<>();  animals.add(new Animal());  animals.add(new Anjing());  animals.add(new Kucing());  System.out.println("Suara Hewan :");  for (Object animal : animals) {  if (animal instanceof Animal) {  Animal a = (Animal) animal;  a.makeSound();  }  }  }  } |  |
| 3 | Buatlah program bebas atau modifikasi program disamping menggunakan polymorphic argument | package P7;  class Animal {  public void makeSound() {  System.out.println("Suara Hewan");  }  }  class Anjing extends Animal {  public void makeSound() {  System.out.println("Guguk");  }  }  class Kucing extends Animal {  public void makeSound() {  System.out.println("Miauw");  }  }  class AnimalShelter {  public void introduce(Animal animal) {  System.out.println("Selamat Datang di Penangkaran");  animal.makeSound();  }  }  public class Polymorph {  public static void main(String[] args) {  AnimalShelter shelter = new AnimalShelter();    Animal animal = new Animal();  Anjing anjing = new Anjing();  Kucing kucing = new Kucing();    shelter.introduce(animal);  shelter.introduce(anjing);  shelter.introduce(kucing);  }  } |  |
| 4 | Buatlah program bebas atau modifikasi program disamping menggunakan operator instanceof | package P7;  class Animal {  public void makeSound() {  System.out.println("Suara Hewan");  }  }  class Dog extends Animal {  public void makeSound() {  System.out.println("Guguk");  }  }  class Cat extends Animal {  public void makeSound() {  System.out.println("Miauw");  }  }  public class Instance {  public static void main(String[] args) {  Animal animal = new Animal();  Dog dog = new Dog();  Cat cat = new Cat();  if (animal instanceof Animal) {  System.out.println("animal instansiasi dari Animal");  animal.makeSound();  }  if (dog instanceof Animal) {  System.out.println("anjing instansiasi dari Animal");  dog.makeSound();  }  if (cat instanceof Animal) {  System.out.println("kucing instansiasi dari Animal");  cat.makeSound();  }  }  } |  |
| 5 | Buatlah program bebas atau modifikasi program disamping menggunakan Object Casting | package P7;  class Animal {  public void makeSound() {  System.out.println("Suara Hewan");  }  }  class Anjing extends Animal {  public void makeSound() {  System.out.println("Guguk");  }  public void playandrun() {  System.out.println("Anjing sedang bermain kejar-kejaran");  }  }  public class Casting {  public static void main(String[] args) {  Anjing anjing = new Anjing();  Animal animal = anjing;  if (animal instanceof Anjing) {  Anjing castedDog = (Anjing) animal;  System.out.println("Casting ke anjing:");  castedDog.makeSound();  castedDog.playandrun();  } else {  System.out.println("Failed to cast to Dog");  }  }  } |  |
| 6 | Buatlah program bebas dengan Up Casting | package P7;  class Animal {  void makeSound() {  System.out.println("Suara Hewan");  }  }  class Anjing extends Animal {  void makeSound() {  System.out.println("Guguk");  }  }  public class UpCasting {  public static void main(String[] args) {  Anjing anjing = new Anjing();  Animal animal = anjing;  animal.makeSound();  }  } |  |
| 7 | Buatlah program bebas dengan Down Casting | package P7;  class Animal {  public void makeSound() {  System.out.println("Suara Hewan");  }  }  class Anjing extends Animal {  public void makeSound() {  System.out.println("Guguk");  }  public void playandrun() {  System.out.println("Anjing sedang bermain kejar-kejaran");  }  }  public class Casting {  public static void main(String[] args) {  Anjing anjing = new Anjing();  Animal animal = anjing;  if (animal instanceof Anjing) {  Anjing castedDog = (Anjing) animal;  System.out.println("Casting ke anjing:");  castedDog.makeSound();  castedDog.playandrun();  } else {  System.out.println("Failed to cast to Dog");  }  }  } |  |
| 8 | Buatlah program untuk membandingkan kedua nilai menggunakan polimorfis statis | package P7;  public class StaticPolymorph {  public static int compare(int a, int b) {  if (a > b) {  return 1;  } else if (a < b) {  return 0;  } else {  return 0;  }  }  public static void main(String[] args) {  int intResult = compare(100, 500);  System.out.println("Comparison of integers: " + intResult);  }  } |  |
| 9 | Buatlah program bebas menggunakan polimorfis dinamis dengan jumlah class : Akhiran NIM ganjil : 3 class Akhiran NIM genap : 4 class | package P7;  class Shape {  public double getArea() {  return 0.0;  }  public void printInfo() {  System.out.println("This is a generic shape.");  }  }  class Lingkaran extends Shape {  private double radius;  public Lingkaran(double radius) {  this.radius = radius;  }  public double getArea() {  return Math.PI \* radius \* radius;  }  @Override  public void printInfo() {  System.out.println("Lingkaran memiliki radius " + radius);  }  }  class PersegiPanjang extends Shape {  private double width;  private double height;  public PersegiPanjang(double width, double height) {  this.width = width;  this.height = height;  }  public double getArea() {  return width \* height;  }  public void printInfo() {  System.out.println("Persegi Panjang memiliki Lebar " + width + " dan Tinggi " + height);  }  }  public class BangunDatar {  public static void main(String[] args) {  Shape shape1 = new Lingkaran(7.0);  Shape shape2 = new PersegiPanjang(7.0, 9);  displayShapeInfo(shape1);  displayShapeInfo(shape2);  }  public static void displayShapeInfo(Shape shape) {  System.out.println("Area: " + shape.getArea());  shape.printInfo();  }  } |  |
| 10 | Modifikasi program disamping menggunakan inputan dinamis | package P7;  import java.util.Scanner;  public class Employee {  private String name;  private double salary;  private static double salary\_rise\_percent = 0.2;  public Employee(String nm, double sly) {  this.setName(nm);  this.setSalary(sly);  }  public void setName(String nm) {  name = nm;  }  public void setSalary(double sly) {  salary = sly;  }  public static void setPresentase(double percent) {  salary\_rise\_percent = percent;  }  public String getName() {  return name;  }  public double getSalary() {  return salary;  }  public static double getPresentase() {  return salary\_rise\_percent;  }  public void salaryUp() {  salary += (salary \* salary\_rise\_percent);  }  }  class Manager extends Employee {  private static double bonus = 500;  public Manager(String nm, double sly) {  super(nm, sly);  }  public double getBonus() {  return bonus;  }  public void setBonus(double bns) {  bonus = bns;  }  public double getSalary() {  double salaryBase = super.getSalary();  return (salaryBase + bonus);  }  }  class TestManager {  public static void main(String[] args) {  Scanner input = new Scanner(System.in);  System.out.print("Masukkan nama: ");  String name = input.next();  System.out.print("Masukkan gaji: ");  double salary = input.nextDouble();  Manager mng = new Manager(name, salary);  System.out.println("Nama: " + mng.getName());  System.out.println("Bonus: " + mng.getBonus());  System.out.println("Gaji: " + mng.getSalary());  }  } |  |
| 11 | Buatlah program tambahan mengikuti contoh disamping lalu mengganti keyword super menjadi this | package P7;  class Person {  String name = "Dori";  int age = 21;  }  class Lecture extends Person {  float salary = 5000000;  String name = "Rismon";  int age = 30;  public void showInfo() {  System.out.println("Nama : " + this.name);  System.out.println("Umur : " + this.age);  System.out.println("Gaji: RP." + salary);  }  }  public class Penggajian {  public static void main(String[] args) {  Lecture rismon = new Lecture();  rismon.showInfo();  }  } |  |
| 12 | Modifikasi program mengikuti contoh disamping (bebas) | package P4;  import java.util.Date;  // Employee.java  class Employee {      private static final double BASE\_SALARY = 15000.00;      private String name;      private double salary;      private Date birthDate;      public Employee(String *name*, double *salary*, Date *birthDate*) {          this.name = *name*;          this.salary = *salary*;          this.birthDate = *birthDate*;      }      public Employee(String *name*, double *salary*) {          this(*name*, *salary*, null);      }      public Employee(String *name*) {          this(*name*, BASE\_SALARY);      }      public String getName() {          return name;      }      public double getSalary() {          return salary;      }      public Date getBirthDate() {          return birthDate;      }  }  // Manager.java  class Manager extends Employee {      private String department;      public Manager(String *name*, double *salary*, String *department*) {          super(*name*, *salary*);          this.department = *department*;      }      public String getDepartment() {          return department;      }  }  // Main.java  public class Salary {      public static void main(String[] *args*) {          Employee man = new Manager("Dori", 16000.00, "Electrical");          System.out.println("Nama: " + man.getName());          System.out.println("Gaji: RP." + man.getSalary());          if (man instanceof Manager) {              Manager manager = (Manager) man;              System.out.println("Department: " + manager.getDepartment());          }      }  } |  |