Exercise 01:

Create a class called “Employee” which has 3 private variables (empID, empName, empDesignation) and create getters and setters for each field. Please note that this has no main method since this is just a blueprint not a application. Now crate a test class to invoke the Employee class. Create two objects for Mr.Bogdan and Ms.Bird and set required values using setters and print them back on the console using getters.

public class Employee {

  private int empId;

  private String empName,empDesig;

  public void setId(int a){

      this.empId=a;

  }

   public void setName(String a){

      this.empName=a;

  }

    public void setDesig(String a){

      this.empDesig=a;

  }

      public int getId(){

      return (+empId);

  }

   public String getName(){

       return ("Employee Name:  "+empName);

  }

    public String getDesig(){

       return ("Employee designation:  "+empDesig);

  }

}

public class Testclass {

    public static void main(String[]args){

        Employee e1 = new Employee();

        e1.setId(123);

        e1.setName("Mr.Bodgan");

        e1.setDesig("Manager");

         System.out.println("Employee ID: "+e1.getId());

         System.out.println(e1.getName());

        System.out.println(e1.getDesig());

        Employee e2 = new Employee();

        e2.setId(234);

        e2.setName("Ms.Bird");

        e2.setDesig(" HR Manager");

        System.out.println("Employee ID: "+e2.getId());

         System.out.println(e2.getName());

         System.out.println(e2.getDesig());

    }

}

Exercise 02:

Develop the following class execute and discuss the answer: Please note that each class stored in separate files. Write down the answer.

class SuperB {

int x;

void setIt (int n) { x=n;}

void increase () { x=x+1;}

void triple () {x=x\*3;};

int returnIt () {return x;}

}

class SubC extends SuperB {

void triple () {x=x+3;} // override existing method

void quadruple () {x=x\*4;} // new method

}

public class TestInheritance {

public static void main(String[] args) {

SuperB b = new SuperB();

b.setIt(2);

b.increase();

b.triple();

System.out.println( b.returnIt() );

SubC c = new SubC();

c.setIt(2);

c.increase();

c.triple();

System.out.println( c.returnIt() ); }

}

The output:

9

6

Exercise 03:

Recall the following scenario discussed during the class. Develop a code base to represent the scenario. Add a test class to invoke Lecturer and Student class by creating atleast one object from each.

Note: All the common attributes and behavior stored in the super class and only the specific fields and behavior stored in subclasses.

|  |
| --- |
| Student |
| * name |
| * id |
| * course |
| + setName()/getName() |
| + setID()/getID() |
| + setCourse()/getCourse() |

|  |
| --- |
| Lecturer |
| * name |
| * id |
| * programme |
| + setName()/getName() |
| + setID()/getID() |
| + setProg()/getProg() |

|  |
| --- |
| Person |
| Identify field and attributes to be stored in this class |

public class Person {

String name;

int id;

public Person(String a, int b){

    this.name=a;

    this.id=b;

}

public void setName(String a){

    name=a;

}

public String getName(){

    return name;

}

public void setID(int a){

    id=a;

}

public int getId(){

    return id;

}

}

public class Student extends Person {

private String course;

public Student(String a,String b, int c){

    super(a,c);

    this.course=b;

}

public void setCourse(String a){

    course=a;

}

public String getCourse(){

    return course;

 }

}

public class Lecturer extends Person{

 private String programe;

 public Lecturer(String a,String b,int c){

     super(a,c);

     programe = b;

 }

public void setProg(String a){

    programe=a;

}

public String getProg(){

    return programe;

 }

}

public class Practical4 {

    public static void main(String[]args){

        Student s1 = new Student("Kamal","SE",123);

        s1.setCourse("CS");

        s1.getCourse();

        System.out.println(s1.getCourse());

        Lecturer l1 = new Lecturer("Amarasiri","FOC",987);

        l1.setProg("FOC.SE");

        System.out.println(l1.getProg());

    }

}

Exercise 04

Develop the following class execute and discuss the answer: Please note that each public class stored in separate files. Write down the answer.

public class Animal{}

public class Mammal extends Animal{}

public class Reptile extends Animal{}

sss

public class Dog extends Mammal{

public static void main(String args[]){

Animal a = new Animal();

Mammal m = new Mammal();

Dog d = new Dog();

System.out.println(m instanceof Animal);

System.out.println(d instanceof Mammal);

System.out.println(d instanceof Animal);

}

}