

­­­­

*DEPARTMENT OF SOFTWARE*

*ENGINEERING*

***ASSIGNMENT***

**DATE OF SUBMISSION**

**16TH August 2021**

* **Course Title:** Object Oriented Programming
* **Section:** II
* **Roll no:** 20SW060
* **Prepared by:** Naveed
* **Assignment : 2**

***Submitted to:***

Mam Maryam

**Task #1**

Make a class called Author (as shown in the class diagram). It is designed to model a book's author. It contains:

Three private instance variables: name (String), email (String), and• gender (char of either 'm' or 'f');

One constructor to initialize the name, email and gender with the given•

values;

public Author (String name, String email, char gender) {......}

There is no default constructor for Author, as there are no defaults for name, email and gender.)

public getters/setters: getName(), getEmail(), setEmail(), and• getGender(); (There are no setters for name and gender, as these attributes cannot be changed.)

A toString() method that returns

"Author[name=?,email=?,gender=?]",• e.g., "Author[name=Tan AhTeck, email=ahTeck@somewhere.com, gender=m]".

**Source Code**

**class Author**

**{**

**private String name;**

**private String Email;**

**private char gender;**

**public Author(String name, String Email, char gender)**

**{**

**this.name=name;**

**this.Email=Email;**

**this.gender=gender;**

**}**

**public String getname()**

**{**

**return name;**

**}**

**public String getEmail()**

**{**

**return Email;**

**}**

**public void setEmail(String Email)**

**{**

**this.Email=Email;**

**}**

**public char getGender()**

**{**

**return gender;**

**}**

**public String toString()**

**{**

**//System.out.println();**

**return "Author[name="+name+", email="+Email+", Gender="+gender;**

**}**

**}**

**public class Task1 {**

**public static void main(String[] args)**

**{**

**Author obj = new Author("Robert Lafore", "Lafore123@gmail.com",'m');**

**System.out.println(obj.toString());**

**}**

**}­­­­­**

**Task #2**

Create a class book as shown in the class diagram.

Four private instance variables: name (String), an array of authors (of• the class Author you have just created), price (double), and qty (int);

Two constructors:•

public Book (String name, Author[] author, double price) { ...... }

Author[] author, double price, int qty) { ...... }

public methods getName(), getAuthors(), getPrice(), setPrice(),• getQty(), setQty().

The toString() method shall return• "Book[name=?,authors={Author[name=?,email=?,gender=?],......}, price=?,qty=?]". Reuse Author class built in the first task•

**Source Code**

**class Author**

**{**

**public String name;**

**public String Email;**

**public char gender;**

**public Author(){}**

**public Author(String name, String Email, char gender)**

**{**

**this.name=name;**

**this.Email=Email;**

**this.gender=gender;**

**}**

**public String getname()**

**{**

**return name;**

**}**

**public String getEmail()**

**{**

**return Email;**

**}**

**public void setEmail(String Email)**

**{**

**this.Email=Email;**

**}**

**public char getGender()**

**{**

**return gender;**

**}**

**public String toString()**

**{**

**//System.out.println();**

**return "Author[name="+name+", email="+Email+", Gender="+gender;**

**}**

**}**

**class Book {**

**private String name\_of\_book;**

**private double price;**

**private int qty;**

**public Author authors =new Author();**

**public Book(String name\_of\_book, Author obj\_of\_Author, double price)**

**{**

**this.name\_of\_book=name\_of\_book;**

**this.price=price;**

**this.authors=obj\_of\_Author;**

**}**

**public Book(String name\_of\_book, Author obj\_of\_Author, double price, int qty)**

**{**

**this.name\_of\_book=name\_of\_book;**

**this.price=price;**

**this.qty=qty;**

**this.authors=obj\_of\_Author;**

**}**

**public String getName( )**

**{ System.out.print("Name of book is ");**

**return name\_of\_book;**

**}**

**public double getPrice()**

**{**

**System.out.print("price of book is ");**

**return price;**

**}**

**public void setprice(double price)**

**{**

**this.price=price;**

**}**

**public int getQty()**

**{ System.out.print("Books Quantity");**

**return qty;**

**}**

**public void setqty(int qty)**

**{**

**this.qty=qty;**

**}**

**public void tostring()**

**{**

**System.out.println("Book[name="+name\_of\_book+" , price="+price+" , qty="+qty);**

**System.out.println("Author[name="+authors.name+" ,Email="+authors.Email+" , Gender="+authors.gender);**

**}**

**}**

**public class Task2 {**

**public static void main(String[] args)**

**{**

**Author author=new Author("Robert", "lafore@gmail.com",'M');**

**Book book= new Book("Cpp",author,40.25,50);**

**book.tostring();**

**}**

**}­­­**

**Task #3**

Create a class that has overloaded methods to print an integer and a character with two methods having the same name but different sequence of the integer and the character parameters.

For example, if the parameters of the first method are of the form (int n, char c), then that of the second method will be of the form (char c, int n).

**Source Code**

class Overloaded

{

private int age;

private char gender;

public void print\_values(int age , char gender)

{ this.age=age;

this.gender=gender;

System.out.println("Age = "+age +" Gender = "+gender);

}

public void print\_values(char gender , int age)

{

this.age=age;

this.gender=gender;

System.out.println("Gender = "+gender+" Age = "+age);

}

}

public class Task3 {

public static void main(String[] args)

{

Overloaded obj = new Overloaded();

obj.print\_values('M', 20);

obj.print\_values(40, 'F');

}

}

**Task #4**

Create a class to print the area of a square and a rectangle. The

class has two methods with the same name but different number of

parameters. The method for printing area of rectangle has two

parameters which are length and breadth respectively while the

other method for printing area of square has one parameter which is

side of square.

**Source Code**

class Shape

{

public double Area(double side)

{ System.out.println("Area of Square");

return 4\*side;

}

public double Area(double length, double width)

{ System.out.println("Area of Rectangle");

return length\*width;

}

}

public class Task4 {

public static void main(String[] args)

{

Shape obj = new Shape();

System.out.println(obj.Area(4.0));

System.out.println(obj.Area(3.2, 4.8));

}

}

**Task #5**

class Student

{

private String name;

private int age;

private String Adress;

public Student()

{

name="unknown";

age=0;

Adress="not set";

}

public void setinfo(String name, int age)

{

this.name=name;

this.age=age;

}

public void setinfo(String name,int age,String Adress)

{

this.name=name;

this.age=age;

this.Adress=Adress;

}

public void getinfo()

{ System.out.println();

System.out.print("Name="+name);

System.out.print(" Age="+age);

System.out.print(" Address="+Adress);

}

}

public class Task5 {

public static void main(String[] args)

{

Student[] students = new Student[5];

for(int i=0; i<students.length; i++)

{

students[i]=new Student();

}

students[1].setinfo("Naveed", 21,"xyz");

students[2].setinfo("Asim", 18,"xyz");

students[3].setinfo("Faizan", 16,"xyz");

students[4].setinfo("Ahmed", 14,"xyz");

students[4].setinfo("Rohan", 12, "xyz");

for (Student student : students) {

student.getinfo();

}

}

}

**Task #6**

. Consider a right-angle triangle.

Write a java class with three instance variables hypotenuse, perpendicular and base.

Create two overloaded constructors for initializing these lengths including a default constructor that gives default values. Create three functions,

• getHyp(double , double) returns hyp after calculation,

• getPerp(double, double) returns perp after calculation,

• getBase(double, double) returns base after calculation.

**Source Code**

import java.util.\*;

class Triangle

{

private double hypotenuse;

private double base;

private double perpendicular;

public Triangle()

{

hypotenuse=0;

base=0;

perpendicular=0;

}

public Triangle(double hypotenuse , double base, double perpendicular) {

this.base=base;

this.hypotenuse=hypotenuse;

this.perpendicular=perpendicular;

}

public double getHyp()

{

double hyp=Math.sqrt(base\*base+perpendicular\*perpendicular);

System.out.print("Hypotenues = ");

return hyp;

}

public double getperp()

{

double perp=Math.sqrt(hypotenuse\*hypotenuse-base\*base);

System.out.print("perpendicular = ");

return perp;

}

public double getbase()

{

double base=Math.sqrt(hypotenuse\*hypotenuse-perpendicular\*perpendicular);

System.out.print("Base = ");

return base;

}

}

public class Task6 {

public static void main(String[] args){

Triangle T1= new Triangle(3.5,2.5,1.5);

System.out.println(T1.getHyp());

System.out.println(T1.getperp());

System.out.println(T1.getbase());

}

}