**Assignment-2**

**1. Write a class, Grader, which has an instance variable, score, an appropriate constructor and appropriate methods. A method, letterGrade(), that returns the letter grade as O/E/A/B/C/F. Now write a demo class to test the Grader class by reading a score from the user, using it to create a Grader object after validating that the value is not negative and is not greater than 100. Finally, call the letterGrade() method to get and print the grade.**

import java.util.Scanner;

class Grade{

float score;

public Grade(float score) {

if (!(score < 0 || score > 100))

this.score = score;

else {

System.out.println("Score is not Valid. System Terminating.");

System.exit(0);

}

}

char letterGrade() {

if (score <= 20)

return 'F';

else if (score > 20 && score <= 40)

return 'C';

else if (score > 40 && score <= 60)

return 'B';

else if (score > 60 && score <= 80)

return 'A';

else if (score > 80 && score <= 90)

return 'E';

else

return 'O';

}

}

public class Grader {

public static void main(String[] args) {

Scanner scan = new Scanner(System.in);

System.out.println("Name : Gokul Sarkar \nRoll : 46");

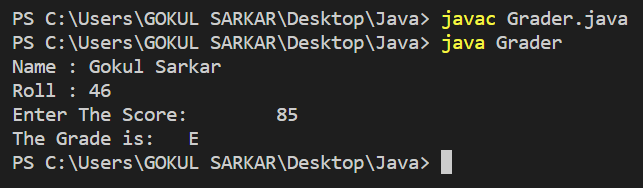
System.out.print("Enter The Score:\t");

System.out.println("The Grade is:\t" + new Grade(scan.nextFloat()).letterGrade());

}

}

**Output:**



**2. Write a class, Commission, which has an instance variable, sales; an appropriate constructor; and a method, commission() that returns the commission. Now write a demo class to test the Commission class by reading a sale from the user, using it to create a Commission object after validating that the value is not negative. Finally, call the commission() method to get and print the commission. If the sales are negative, your demo should print the message “Invalid Input”.**

import java.util.Scanner;

class Calculate{

float sales;

public Calculate(float sales) {

if (sales < 0) {

System.out.println("Invalid Input");

System.exit(0);

}

this.sales = sales;

}

public float commission() {

return (sales \* 18) / 100;

}

}

public class Commission {

public static void main(String[] args) {

Scanner scan = new Scanner(System.in);

System.out.println("Name : Gokul Sarkar \nRoll : 46");

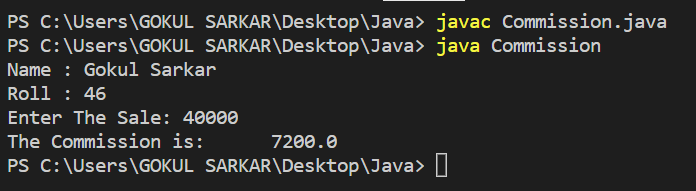
System.out.print("Enter The Sale:\t");

System.out.println("The Commission is:\t" + new Calculate(scan.nextFloat()).commission());

}

}

**Output:**



**3. For a Mobile Shop project, create “Telephone” class with details like mobile\_id, model\_name and available\_quantity in “Phone” package. Inherit from this class and create a class for “smart\_phone” with necessary details like enabled\_5G, foldable and dual\_screen in package “Smart”. The customer e ecutive tries to display all smart\_phone details (mobile\_id, model\_name, available\_quantity, enabled\_5G, foldable and dual\_screen) and updates the quantity information, whenever the customer purchases the smart\_phone. Write the necessary java programs to implement this scenario and test with user inputs.**

**4. An educational institution maintains a database of its employees. The database is divided into a number of classes whose hierarchical relationships are shown below. Write all the classes and deﬁne the methods to create the database and retrieve individual information as and when needed. Write a driver program to test the classes.**

**StaI (code, name)**

**Teacher (subject, publication) is a StaI**

**O cer (grade) is a StaI**

**Typist (speed) is a StaI**

**RegularTypist (remuneration) is a Typist**

**CasualTypist (daily wages) is a Typist.**

class Staff {

String code, name;

public Staff(String code, String name) {

this.code = code;

this.name = name;

}

}

class Teacher extends Staff {

String subject, publication;

public Teacher(String subject, String publication) {

super("2580", "Rahul");

this.subject = subject;

this.publication = publication;

}

void display() {

System.out.println("Name : Gokul Sarkar \nRoll : 46");

System.out.println("Code:\t"+code);

System.out.println("Name:\t"+name);

System.out.println("Subject:\t"+subject);

System.out.println("Publication:\t"+publication);

}

}

class Officer extends Staff {

String grade;

public Officer(String grade) {

super("1234", "Anupam");

this.grade = grade;

}

void display() {

System.out.println("Code:\t"+code);

System.out.println("Name:\t"+name);

System.out.println("Grade:\t"+grade);

}

}

class Typist extends Staff {

int speed;

public Typist(int speed) {

super("TypistCode", "TypistName");

this.speed = speed;

}

}

class RegularTypist extends Typist {

int remuneration;

public RegularTypist(int speed, int remuneration) {

super(speed);

this.remuneration = remuneration;

}

void display() {

System.out.println("Speed:\t"+speed);

System.out.println("Remuneration:\t"+remuneration);

}

}

class CasualTypist extends Typist {

int daily\_wages;

public CasualTypist(int speed, int daily\_wages) {

super(speed);

this.daily\_wages = daily\_wages;

}

void display() {

System.out.println("Speed:\t"+speed);

System.out.println("Daily Wages:\t"+daily\_wages);

}

}

public class Employee\_Database {

public static void main(String[] args) {

Teacher t = new Teacher("Biology", "Roy and Martin");

t.display();

Officer o = new Officer("A");

o.display();

RegularTypist r = new RegularTypist(90, 1200);

r.display();

CasualTypist c = new CasualTypist(60, 800);

c.display();

}

}

**Output:**

