**Term -1**

1. Create a class called Employee. Define two class variables employee\_name and employee\_salary in it. Create three methods in the class named getData() to accept values from the user, and a method annualSalary() to compute the annual salary of the employee, and displayInfo() to print name and salary of employee.

**Program :**

**//Employee.java**

**package** hcl.term1.com;

**import** java.util.Scanner;

//create class Employee

**public** **class** Employee {

//intialise class variables

String Employee\_name;

**double** Employee\_salary;

**double** annual\_salary;

Scanner sc= **new** Scanner(System.***in***);

//getData() method to accept values from the user

**public** **void** getData() {

System.***out***.println("Enter name of the Employee");

Employee\_name=sc.next();

System.***out***.println("Enter salary of the Employee");

Employee\_salary = sc.nextDouble();

}

//a method annualSalary() to compute the annual salary of the employee

**public** **void** annualSalary() {

annual\_salary = Employee\_salary\*12;

System.***out***.println("annual salary of the employee is :" + annual\_salary);

}

//displayInfo() to print name and salary of employee

**public** **void** displayInfo() {

System.***out***.println("=====================================");

System.***out***.println("Details of the employee : ");

System.***out***.println("name of the employee is : " +Employee\_name +" salary is : "+Employee\_salary+" annual\_income is : " +annual\_salary);

}

}

//EmployeeMain.java

//main function

**package** hcl.term1.com;

**public** **class** EmployeeMain {

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

// **TODO** Auto-generated method stub

//create a employee object

Employee e = **new** Employee();

//call methods using Employee object

e.getData();

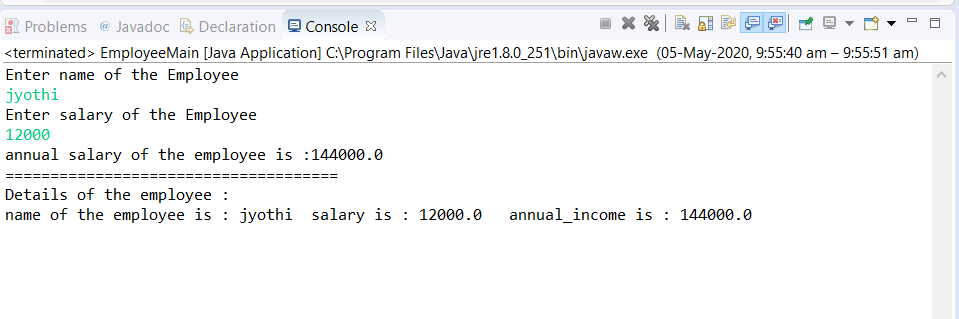
e.annualSalary();

e.displayInfo();

}

}

**Output :**

****

1. **Write a program where calculate() method of super class - "Shapes" is overridden by the "Calculate" method of two subclasses "Rectangle" and "Triangle". The behavior of the calculate method is dynamically decided depending on the object it is called for.**

**Program:**

**//shapes.java**

**package** hcl.term1.com;

//create class shapes

**public** **class** Shapes {

//create method calculate

**public** **void** calculate(**double** h , **double** r) {

}

}

**// Rectangle.java**

**package** hcl.term1.com;

//create class Rectangle using overridden method

**public** **class** Rectangle **extends** Shapes{

//create a mehod calculate

**public** **void** calculate(**double** length , **double** width) {

**double** area = length\*width;

System.***out***.println("area of rectangle is : "+area);

}

}

**//Triangle.java**

**package** hcl.term1.com;

//create class Triangle

**public** **class** Triangle **extends** Shapes{

//calculate method

**public** **void** calculate( **double** height, **double** breadth) {

**double** area =0.5\*breadth\*height;

System.***out***.println("area of the triangle is :"+area);

}

}

**//ShapesMain.java**

**package** hcl.term1.com;

**import** java.util.Scanner;

//ShapesMain class

**public** **class** ShapesMain {

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

// **TODO** Auto-generated method stub

Scanner sc= **new** Scanner(System.***in***);

//length of rectangle

System.***out***.println("Enter lenghth of rectangle : ");

**double** len=sc.nextDouble();

//breadth of rectangle

System.***out***.println("enter breadth of rectangle : ");

**double** wid = sc.nextDouble();

//rectangle object called

Shapes r = **new** Rectangle();

r.calculate(len,wid);

System.***out***.println("==============================================");

//height of triangle

System.***out***.println("Enter height of the triangle : ");

**double** height = sc.nextDouble();

//breadth of triangle

System.***out***.println("Enter breadth of triangle : ");

**double** bre = sc.nextDouble();

//Triangle object called

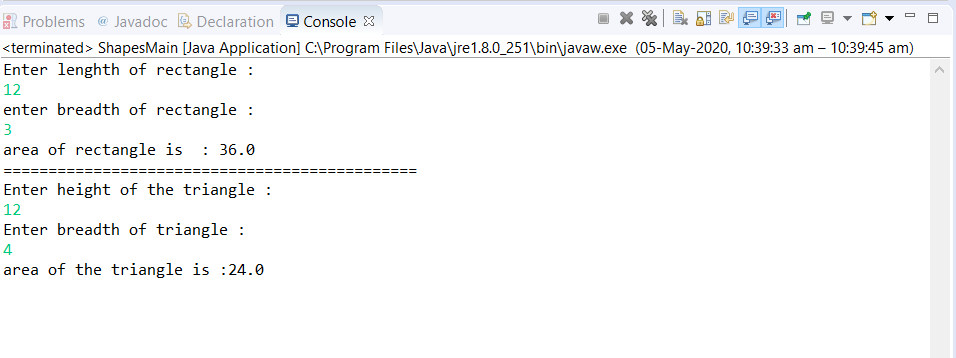
Shapes t = **new** Triangle();

t.calculate(height,bre);

}

}

**Output :**

****

1. Lets say we have a requirement where we need to register the students when their age is less than 12 and weight is less than 40, if any of the condition is not met then the user should get an ArithmeticException with the warning message “Student is not eligible for registration”. We have to implement the logic by placing the code in the method that checks student eligibility if the entered student age and weight doesn’t meet the criteria then throw the exception using throw keyword.

**Program:**

**//CheckStudentData.java**

**public** **class** CheckStudentData {

//check method for checking the age and weight met the given condition or not

**public** **void** check(**int** student\_id,String name,**int** age,**float** weight){

//try block for raising an exception who have age<12 and weight <40

**try**{

**if**(age<12 || weight <40){

**throw** **new** ArithmeticException();

}

**else**{

System.out.println(name +" you are eligible ");

}

}

//catching an arithmetic exception

**catch**(ArithmeticException ae){

System.out.println(name+" is not eligible for registration");

}

}

}

//StudentData.java:

**import** java.util.\*;

**public** **class** StudentData {

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

Scanner sc=**new** Scanner(System.***in***);

//Reading the student id,name,age,weight

System.***out***.println("enter student id:");

intstudent\_id=sc.nextInt();

System.***out***.println("enter name");

String name=sc.next();

System.***out***.println("enter age");

**int** age=sc.nextInt();

System.***out***.println("enter weight");

Float weight=sc.nextFloat();

//creating an object for CheckStudentData class

CheckStudentData obj1=**new** CheckStudentData();

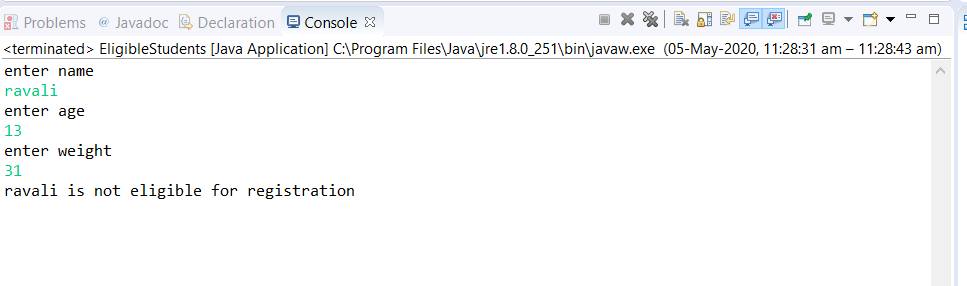
//calling the check() method

obj1.check(student\_id,name,age,weight);

}

}

Output :



1. Create a project Bank using Eclipse. Create a class Account that stores customer name, account number and type of account. From this derive the classes Curr-acct and Sav-acct to make them more specific to their requirements. Include the necessary methods in order to achieve the following tasks: 1) Accept deposit from a customer and update the balance. 2) Display the balance. 3) Compute and deposit interest. 4) Permit withdrawal and update the balance. 5) Check for the minimum balance, impose penalty, if necessary, and update the balance. "

**Program :**

Account.java

**package** hcl.term1.com;

//Account.java

**public** **class** Account {

String customer\_name;

**int** acc\_number;

String acc\_type;

}

**//savingsAccount.java**

//SavingsAccount

**package** hcl.term1.com;

**public** **class** SavingsAccount **extends** Account{

**double** balance=30000;

//deposit() method

**public** **void** deposit(**double** dep\_amount){

**if**(dep\_amount >0){

balance=balance+dep\_amount;

System.***out***.println(" Your Balance :"+balance);

}

**else**

System.***out***.println("Enter the correct amount to deposit..");

}

//getInterst() method

**public** **void** getInterest(){

**double** interest;

interest=balance\*0.15;

**this**.deposit(interest);

}

//pentalty() method

**public** **void** penalty(){

**int** penalty=50;

**if**(balance<500){

System.***out***.println("Minimum balance should be 500");

System.***out***.println("Penalty fined due to lack of minimum balance is : "+penalty);

}

}

//withdrawal() method

**public** **void** withdrawal(**double** wd\_amount){

**if**(wd\_amount<0)

System.***out***.println("Enter the value greater than zero");

**else**{

**if**(wd\_amount>balance)

System.***out***.println("Insufficinet balance");

**else**{

balance=balance-wd\_amount;

System.***out***.println("Your AccountBalnce is :"+balance);

}

}

}

//displayBalnce() method

**public** **void** displayBalance(){

System.***out***.println("Your AccountBalnce is :"+balance);

**this**.penalty();

}

}

**//currentAccount.java**

**package** hcl.term1.com;

**public** **class** currentAccount **extends** Account{

**double** balance=30000;

//deposit() method

**public** **void** deposit(**double** dep\_amount){

**if**(dep\_amount >0){

balance=balance+dep\_amount;

System.***out***.println(" Your Balance :"+balance);

}

**else**

System.***out***.println("Enter the correct amount to deposit..");

}

//getInterst() method

**public** **void** getInterest(){

**double** interest;

interest=balance\*0.15;

**this**.deposit(interest);

}

//pentalty() method

**public** **void** penalty(){

**int** penalty=50;

**if**(balance<500){

System.***out***.println("Minimum balance should be 500");

System.***out***.println("Penalty fined due to lack of minimum balance is : "+penalty);

}

}

//withdrawal() method

**public** **void** withdrawal(**double** wd\_amount){

**if**(wd\_amount<0)

System.***out***.println("Enter the value greater than zero");

**else**{

**if**(wd\_amount>balance)

System.***out***.println("Insufficinet balance");

**else**{

balance=balance-wd\_amount;

System.***out***.println("Your AccountBalnce is :"+balance);

}

}

}

//displayBalnce() method

**public** **void** displayBalance(){

System.***out***.println("Your AccountBalnce is :"+balance);

**this**.penalty();

}

}

**//AccountMain.java**

**package** hcl.term1.com;

**import** java.util.Scanner;

**public** **class** AccountMain {

**public** **static** <CurrentAccount> **void** main(String args[]){

Scanner sc=**new** Scanner(System.***in***);

//customer name

System.***out***.println("Enter the customer name:");

String name=sc.next();

//account number

System.***out***.println("Enter the Account number:");

**int** account=sc.nextInt();

//account type

System.***out***.println("Enter the Account type[savings/current]:");

String type=sc.next();

**int** opt;

String flag="yes";

**double** amount;

//if account type is savings

**if**(type.equals("savings")){

//SavingsAccount object is created

SavingsAccount s=**new** SavingsAccount();

System.***out***.println("Customer Name:"+name);

System.***out***.println("Customer Account Number:"+account);

System.***out***.println("Account type:"+type);

//use do-while for continue the process

**do**{

//enter option

System.***out***.println("Choose operations 1.deposit\n 2.Viewbalance\n 3.withdrawal");

opt=sc.nextInt();

//use switch case

**switch**(opt){

**case** 1:

//deposit

System.***out***.println("Enter deposit amount");

amount=sc.nextInt();

s.deposit(amount);

**break**;

**case** 2:

//view balance

s.displayBalance();

**break**;

**case** 3:

//withdraw

System.***out***.println("Enter withdrawl amount");

amount=sc.nextInt();

s.withdrawal(amount);

**break**;

**default**:

System.***out***.println("Choose correct option...");

**break**;

}

System.***out***.println("Do you want to continue:");

flag=sc.next();

}

**while**(flag.equals("Yes"));

}

**if**(type.equals("current")){

CurrentAccount c=**new** CurrentAccount();

System.***out***.println("Customer Name:"+name);

System.***out***.println("Customer Account Number:"+account);

System.***out***.println("Account type:"+type);

//use do-while for continue process

**do**{

//choose option

System.***out***.println("Choose operations 1.deposit\n 2.Viewbalance\n 3.withdrawal");

opt=sc.nextInt();

//use switch case

**switch**(opt){

**case** 1:

//deposit

System.***out***.println("Enter deposit amount");

amount=sc.nextInt();

c.deposit(amount);

**break**;

**case** 2:

//view balance

c.displayBalance();

**break**;

**case** 3:

//withdraw

System.***out***.println("Enter withdrawl amount");

amount=sc.nextInt();

c.withdrawal(amount);

**break**;

**default**:

System.***out***.println("Choose correct option...");

**break**;

}

System.***out***.println("Do you want to continue:");

flag=sc.next();

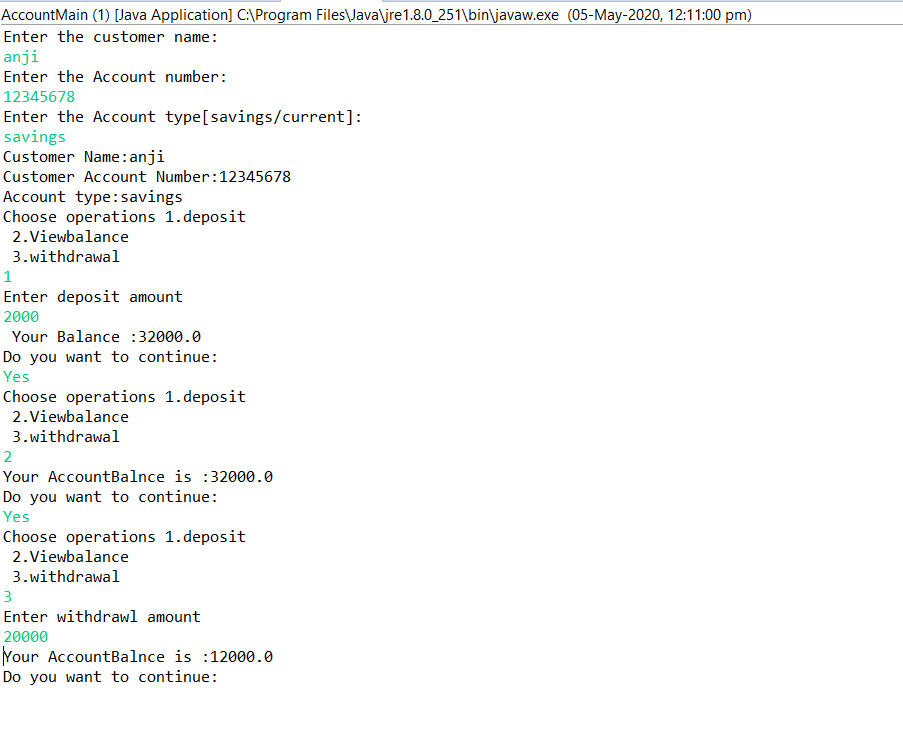
}**while**(flag.equals("Yes"));

}

}

}

Output :



1. Write a program which shows how to rad data from the keyboard and write it to myfile.txt file using FileOutputStream. Also read data from myfile.txt using FileInputStream and display it on the monitor. "

Program :

**package** hcl.term1.com;

**import** java.io.\*;

**import** java.util.\*;

//cclass File

**class** File

{

**public** **static** **void** main(String[] args) **throws** IOException,ClassNotFoundException

{

Scanner sc = **new** Scanner(System.***in***);

//number of students

System.***out***.print("Enter number of students : ");

**int** n =sc.nextInt();

//Initialize array of string

String[] str = **new** String[n];

//names of students

**for**(**int** i=0;i<n;i++) {

System.***out***.println("Enter name of the student : ");

str[i]=sc.next();

}

// Exception Handling

**try**

{

// Creating a new file

FileOutputStream fos = **new** FileOutputStream("myfile.txt");

// Converting to byte stream

ObjectOutputStream oos = **new** ObjectOutputStream(fos);

// Writing to new file from keyboard

oos.writeObject(str);

// Closing the file

System.***out***.println("==================================");

System.***out***.println("Successfylly store name in myfile.txt ");

oos.close();

// Opening the created file

ObjectInputStream ois = **new** ObjectInputStream(**new** FileInputStream("myfile.txt"));

// Converting and storing in string from byte stream

String[] s = (String[]) ois.readObject();

System.***out***.println("============================");

System.***out***.println("read name from myfile.txt ");

System.***out***.println("===============================");

**for**(**int** i=0;i<n;i++){

System.***out***.println("The name is : " + s[i]);

}

// Closing the file

ois.close();

}

**catch**(IOException e)

{

e.printStackTrace();

}

/\*catch(FileNotFoundException e)

{

System.out.print("The file could not found");

} \*/

**catch**(ClassNotFoundException e)

{

e.printStackTrace();

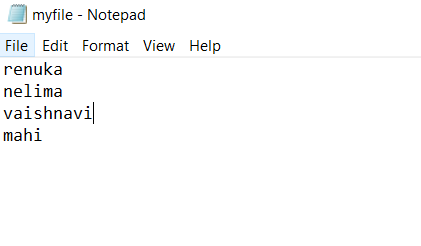
}

}

}

**Output :**

****

****

1. **Write a Java Program to store product objects in TreeSet. The project class has pid,pname,price and category instance variables. Create appropriate constructors and setters-getters and display method. The products should be stored in TreeSet in ascending order of their price.**

**Program :**

//Products.java

pa

ckage hcl.term1.com;

**public** **class** Products{

//attributes like pid,pname,price,category

**int** pid;

String pname;

**float** price;

String category;

//constructor using above attributes

**public** Products(**int** pid,String pname,**float** price,String category){

**this**.pid=pid;

**this**.pname=pname;

**this**.price=price;

**this**.category=category;

}

//getters and setters methods of above attributes

**public** **int** getPid() {

**return** pid;

}

**public** **void** setPID(**int** pid) {

**this**.pid = pid;

}

**public** String getPName() {

**return** pname;

}

**public** **void** setPName(String pname) {

**this**.pname=pname;

}

**public** **float** getPrice() {

**return** price;

}

**public** **void** setPrice(**float** price) {

**this**.price=price;

}

**public** String getCategory() {

**return** category;

}

**public** **void** setCategory(String category) {

**this**.category=category;

}

**public** String toString(){

**return** "pid: "+**this**.pid+"-- pname: "+**this**.pname+"-- price: "+**this**.price+"--- category: "+**this**.category;

}

}

**//productMain.java**

//ProductMain.java

**package** hcl.term1.com;

**import** java.util.Comparator;

**import** java.util.TreeSet;

**public** **class** ProductMain {

**public** **static** **void** main(String a[]){

//TreeSet

TreeSet<Products> Prod = **new** TreeSet<Products>(**new** ProductComparator ());

//storing products using TreeSet

Prod.add(**new** Products(1,"MObile",4000,"Personal"));

Prod.add(**new** Products(2,"Washing Machine",9000,"Home"));

Prod.add(**new** Products(3,"Television",500,"Home"));

Prod.add(**new** Products(4,"Desktop",5300,"office"));

Prod.add(**new** Products(5,"laptop",10000,"Home"));

Prod.add(**new** Products(6,"headsets",100,"personal"));

//Display products

**for**(Products number : Prod){

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

}

}

}

//ProductComparator.java

**public** **class** ProductComparator **implements** Comparator<Products>{

@Override

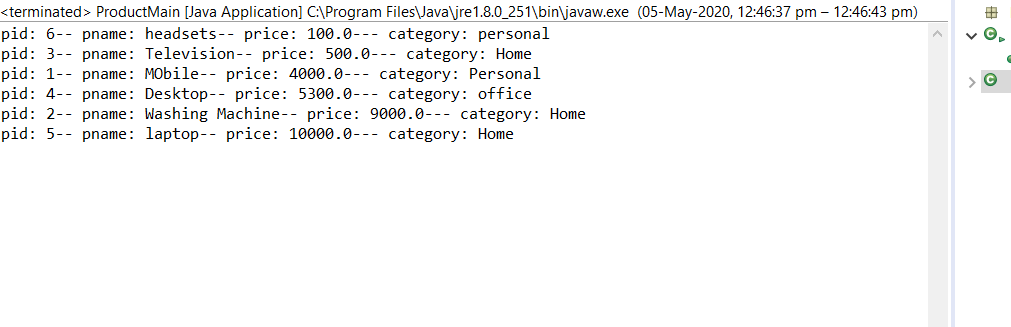
**public** **int** compare(Products p1, Products p2) {

**return** (**int**) (p1.getPrice()-p2.getPrice());

}

}

**Output :**

****