Q1)

import java.io.\*;

import java.util.Scanner;

class singletonA{

static singletonA obj =new singletonA();

int i;

String s="hello";

private singletonA(){

}

public static singletonA getInstance(){

return obj;

}

public void hello(){

return obj;

System.out.println(s);

}

}

class singletonB extends singletonA{

int a=0;

public void bMethord(){

System.out.println("hello fromB");

}

}

public class singleton {

public static void main(String[] args){

singletonA obj1 = singletonA.getInstance();

singletonA obj2 = singletonA.getInstance();

System.out.println("value of obj1 " + obj1.i);

System.out.println("value of obj2 " + obj2.i);

obj1.i=10000;

System.out.println("value of obj1 " + obj1.i);

System.out.println("value of obj2 " + obj2.i);

singletonB obj3 = new singletonB();

obj3.bMethord();

obj3.hello();

}

}

Q2)

**import** java.util.Random;

**import** java.util.stream.DoubleStream;

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

String name;

**int** age;

**public** **void** salary(){

System.***out***.println("Welcome to Employee Module");

}

}

**public** **class** Manager **extends** Employee{

**int** cur\_sal;

**int** incentive;

**int** sal;

Random random= **new** Random();

**public** **void** salary() {

cur\_sal = random.nextInt(25000);

incentive = random.nextInt(2000);

sal = cur\_sal + incentive;

System.***out***.println("Manager's salary is : "+ cur\_sal);

System.***out***.println("Manager's Total Salary is: "+ sal);

}

}

**public** **class** Labour **extends** Employee{

**int** cur\_sal;

**int** ot;

**int** sal;

Random random= **new** Random();

**public** **void** salary() {

cur\_sal = random.nextInt(80000);

ot = random.nextInt(1000);

sal = cur\_sal + ot;

System.***out***.println("Labour's salary is: "+ cur\_sal);

System.***out***.println("Labour's total salary with overtime is: "+ sal);

}

}

Q3)

**import** java.util.Random;

**interface** savingsAcc{

**int** ***upperbound*** = 80;

Random ***r*** = **new** Random();

**default** **int** cash() {

**return** ***r***.nextInt(***upperbound***);

}

}

**import** java.util.Random;

**interface** currentAcc {

**int** ***upperbound*** = 100;

Random ***r*** = **new** Random();

**default** **int** cash() {

**return** ***r***.nextInt(***upperbound***);

}

}

**import** java.util.Scanner;

**public** **class** TotalCash **implements** savingsAcc, currentAcc{

**public** **int** cash() {

**int** sCash = savingsAcc.**super**.cash();

**int** dCash = currentAcc.**super**.cash();

//System.out.println("Total Cash in the bank is: "+ (sCash+dCash));

**return** sCash+dCash;

}

**public** **void** print(**int** tcc) {

System.***out***.println("Total Cash in the bank is: "+ tcc);

}

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

TotalCash tc = **new** TotalCash();

TotalCash tc1 = **new** TotalCash();

**int** x = tc.cash();

tc1.print(x);

}

}

Q4)

import java.io.\*;

import java.util.Scanner;

abstract class Animal{

private String name;

public Animal(String name){

this.name=name;

}

public abstract void move();

}

class Dog extends Animal{

public Dog(String name){

super(name);

}

public void move(){

System.out.println("\nDog runs");

}

}

class Monkey extends Animal{

private int cashCredit=0;

public Monkey(String name){

super(name);

}

public void move(){

System.out.println("\nMonkey climbs");

}

}

public class abstract {

public static void main(String[] args){

Monkey m= new Monkey("Ceacer");

Dog d= new Dog("Rambo");

m.move();

d.move();

}

}

Q5)

import java.io.\*;

import java.util.Scanner;

abstract class shape{

public abstract void draw();

}

class line extends shape{

public void draw(){

System.out.println("\nDrawing line - line is 1d");

}

}

class rectangle extends shape{

public void draw(){

System.out.println("\nDrawing rectangle - rectangle is 2d");

}

}

class cube extends shape{

public void draw(){

System.out.println("\nDrauing cube - cube is 3d");

}

}

public class draw {

public static void main(String[] args){

line l1 = new line();

rectangle r1 = new rectangle();

cube c1 = new cube();

l1.draw();

r1.draw();

c1.draw();

}

}

Q6)

import java.io.\*;

import java.util.Scanner;

import java.lang.Math;

abstract class Persistance{

public abstract void persist();

}

class DatabasePersistance extends Persistance{

public void persist(){

System.out.println("\nYour data have DatabasePersistance.\n");

}

}

class FilePersistance extends Persistance{

public void persist(){

System.out.println("\nYour data have FilePersistance.\n");

}

}

public class DataPersistance {

public static void main(String[] args){

int randomNum = (int)(Math.random() \* 9);

if((randomNum%2)==0){

Persistance client = new DatabasePersistance();

client.persist();

}

else{

Persistance client = new FilePersistance();

client.persist();

}

}

}

Q7)

import java.io.\*;

import java.util.Scanner;

import java.lang.Math;

abstract class Dessert{

protected int tax=0,stock=0;

protected float price=0f;

Dessert(float p,int t,int s){

price=p;

tax=t;

stock=s;

}

public abstract float getCost(int n);

public abstract void addItem(int n);

public abstract void buyItem(int n);

}

class Candy extends Dessert{

Candy(float p,int t,int s){

super(p,t,s);

}

public void getDetails(){

System.out.println("1.Candy \n\t Price : "+price+" $ \n\t Stock : "+stock+"\n\t Tax\t: "+tax+"%");

}

public float getCost(int n){

float cost=0,taxAmount=0;

cost=price\*n;

taxAmount = (cost\*tax)/100;

cost+=taxAmount;

cost\*=60;

return cost;

}

public void addItem(int n){

stock+=n;

System.out.println("\nItem to stock upadated successfully\n Current stock :\n");

getDetails();

}

public void buyItem(int n){

stock-=n;

System.out.println("\n Order successfull\nDetails:\n");

System.out.print("Item : Candy\nQty \t: "+n+"\nTotal Cost : ");

System.out.print(getCost(n));

System.out.print("₹");

}

}

class Cookie extends Dessert{

Cookie(float p,int t,int s){

super(p,t,s);

}

public void getDetails(){

System.out.println("2.Cookie \n\t Price : "+price+" € \n\t Stock : "+stock+"\n\t Tax\t: "+tax+"%");

}

public float getCost(int n){

float cost=0,taxAmount=0;

cost=price\*n;

taxAmount = (cost\*tax)/100;

cost+=taxAmount;

cost\*=70;

return cost;

}

public void addItem(int n){

stock+=n;

System.out.println("\nItem to stock upadated successfully\n Current stock\n");

getDetails();

}

public void buyItem(int n){

stock-=n;

System.out.println("\n Order successfull\nDetails:\n");

System.out.print("Item : Cookie\nQty \t: "+n+"\nTotal Cost : ");

System.out.print(getCost(n));

System.out.print("₹");

}

}

class IceCream extends Dessert{

IceCream(float p,int t,int s){

super(p,t,s);

}

public void getDetails(){

System.out.println("3.IceCream \n\t Price : "+price+" ₹ \n\t Stock : "+stock+"\n\t Tax\t: "+tax+"%");

}

public float getCost(int n){

float cost=0,taxAmount=0;

cost=price\*n;

taxAmount = (cost\*tax)/100;

cost+=taxAmount;

return cost;

}

public void addItem(int n){

stock+=n;

System.out.println("\nItem to stock upadated successfully\n Current stock\n");

getDetails();

}

public void buyItem(int n){

stock-=n;

System.out.println("\n Order successfull\nDetails:\n");

System.out.print("Item : IceCream\nQty : "+n+"\nTotal Cost : ");

System.out.print(getCost(n));

System.out.print("₹");

}

}

public class DessertItem {

static Candy candy= new Candy(1,10,10);

static Cookie cookie= new Cookie(1,20,3);

static IceCream iceCream= new IceCream(50,10,3);

public static void listItem(){

candy.getDetails();

cookie.getDetails();

iceCream.getDetails();

}

public static void main(String[] args){

int i=0;

do{

int itemNo=0,qty=0;

System.out.println("\n\nPlease select your role \n1. Owner \n2. Customer\n0. Exit\n");

Scanner sin= new Scanner(System.in);

i = sin.nextInt();

if(i==1){

listItem();

System.out.println("\nPlease select item to add");

itemNo = sin.nextInt();

System.out.print("\nPlease enter quantity : ");

qty = sin.nextInt();

switch(itemNo) {

case 1:

candy.addItem(qty);

break;

case 2:

cookie.addItem(qty);

break;

case 3:

iceCream.addItem(qty);

break;

default:

System.out.println("\nInvalid choice.\n");

}

}

else if(i==2){

listItem();

System.out.println("\nPlease select an item to order");

itemNo = sin.nextInt();

System.out.print("\nPlease enter quantity : ");

qty = sin.nextInt();

switch(itemNo) {

case 1:

candy.buyItem(qty);

break;

case 2:

cookie.buyItem(qty);

break;

case 3:

iceCream.buyItem(qty);

break;

default:

System.out.println("\nInvalid choice.\n");

}

}

else if(i!=0){

System.out.println("\nInvalid choice.\n");

}

}while(i!=0);

}

}