**OOPS ASSIGNMENT. 1**

**SUBMITTED BY**:

NAME: PASHUPATI NATH VERMA

REG.NO:14CO130

**1. TO PRINT HELLO WORLD**

import java.io.\*;

import java.util.\*;

public class Hello{

public static void main(String args[])

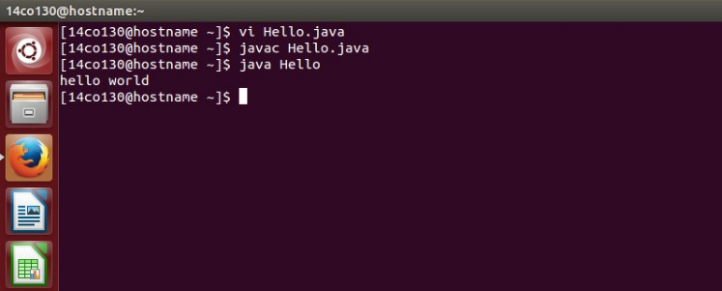
{

System.out.println("Hello World");

}

}

**OUTPUT**



**2. Program to print all the numbers until a zero is encountered.**

import java.io.\*;

import java.util.Scanner;

public class Print

{

public static void main(String args[])

{

int n;

Scanner sc=new Scanner(System.in);

System.out.println("enter the integer");

n=sc.nextInt();

print(n);

}

public static void print(int n)

{

int i;

for(i=n;i>=0;i--)

{

System.out.print(i);

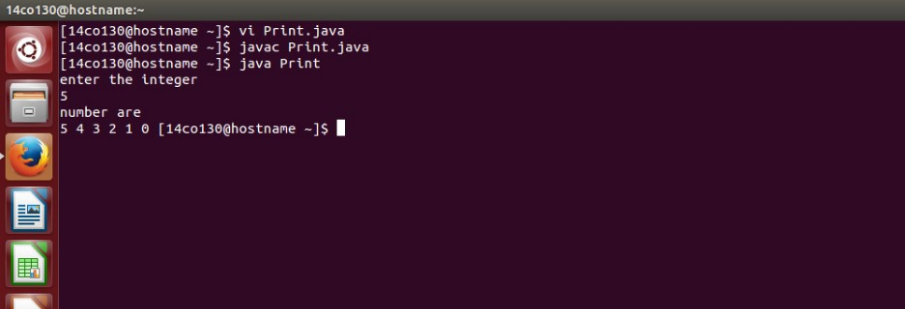
System.out.print(" ");

}

}

}

**OUTPUT**



**3. PROGRAM TO PRINT CHARACTERS FROM ‘a’ TO ‘z’ USING LOOP.**

import java.io.\*;

import java.util.Scanner;

public class Char

{

public static void main(String args[])

{

char ch;

for(ch='a';ch<='z';ch++)

{

System.out.print(ch);

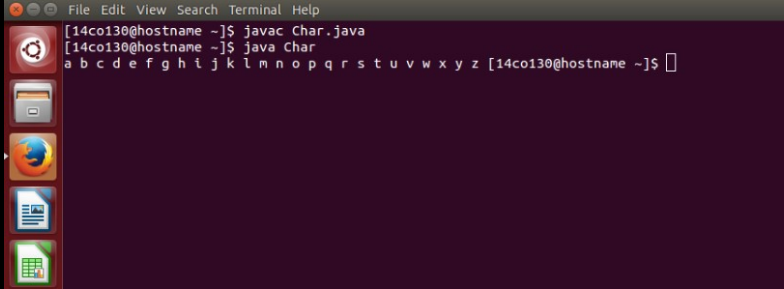
System.out.print(" ");

}

}

}

**OUTPUT**



**4. Create a class DateTime containting private fields: Day, Month,Year, Hour, Minutes, Seconds.**

**Constructors:**  
(i) DateTime()  
(ii) DateTime(dd,mm,yyyy)  
(iii) DateTime(dd,mm,yyyy,hh,mm,ss)

**Methods:**

(i) **String toDateString()** - will return the date as string in the format - dd/mm/yyyy hh:mm:ss

(ii) **String toDateString(String Format)**  
- if Format is "DM" then return the date string in the format - mm/dd/yyyy  
- if "DD" - dd/mm/yyyy  
- if "TS" - hh:mm  
- if "TL" - hh:mm:ss  
- "DMTS" - mm/dd/yyyy hh:mm  
- Other valid formats are : "DMTL", "DDTS", "DDTL". Rest are invalid.

import java.io.\*;

import java.util.\*;

class DateTime {

private int d,m,yr,hr,mi,s;

public DateTime(int day,int month,int year,int hour,int min,int sec)

{

this.d=day;

this.m=month;

this.yr=year;

this.hr=hour;

this.mi=min;

this.s=sec;

}

public DateTime()

{

}

public DateTime(int day,int month,int year)

{

this.d=day;

this.m=month;

this.yr=year;

}

public String toString()

{

String s="";

s=s+this.d+"/"+this.m+"/"+this.yr+" "+this.hr+"/"+this.mi+"/"+this.s;

return s;

}

public String toString(String a)

{

String s="";

if(a.equals("DM"))

{

s=s+this.m+"/"+this.d+"/"+this.yr;

}

else if(a.equals("DD"))

{

s=s+this.d+"/"+this.m+"/"+this.yr;

}

else if(a.equals("TL"))

{

s=s+this.hr+"/"+this.mi+"/"+this.s;

}

else if(a.equals("TS"))

{

s=s+this.hr+"/"+this.mi;

}

else if(a.equals("DMTL"))

{

s=s+this.m+"/"+this.d+"/"+this.yr+" "+this.hr+"/"+this.mi+"/"+this.s;

}

else if(a.equals("DDTL"))

{

s=s+this.d+"/"+this.m+"/"+this.yr+" "+this.hr+"/"+this.mi+"/"+this.s;

}

else if(a.equals("DDTS"))

{

s=s+this.d+"/"+this.m+"/"+this.yr+" "+this.hr+"/"+this.mi;

}

else if(a.equals("DMTS"))

{

s=s+this.m+"/"+this.d+"/"+this.yr+" "+this.hr+"/"+this.mi;

}

return s;

}

public class Time{

public static void main(String args[])

{

Scanner sc=new Scanner(System.in);

int d,m,yr,hr,mi,s;

System.out.println("Enter the date");

d=sc.nextInt();

System.out.println("Enter the month");

m=sc.nextInt();

System.out.println("Enter the year");

yr=sc.nextInt();

System.out.println("Enter the hour");

hr=sc.nextInt();

System.out.println("Enter the minute");

mi=sc.nextInt();

System.out.println("Enter the second");

s=sc.nextInt();

DateTime d1=new DateTime(d,m,yr,hr,mi,s);

int ch;

System.out.println("Do you want to give Output Format 1.Yes 2.No\nEnter Choice");

ch=sc.nextInt();

String str="";

if(ch==2)

{

str=d1.toString();

}

else if(ch==1)

{

String st="",s1="";

System.out.println("Enter choice");

s1=sc.nextLine();

st=sc.nextLine();

str=d1.toString(st);

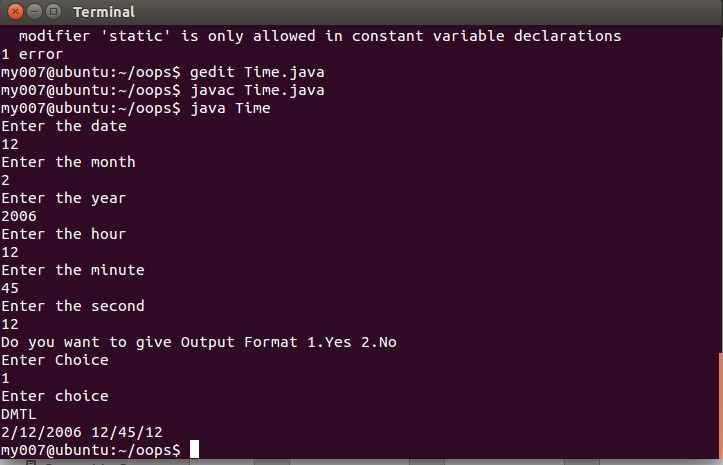
}

System.out.println(str);

}

}

**OUTPUT**



**5. Program to add, subtract two Complex number.**

import java.util.Scanner;

import java.lang.Math;

class complex

{

int real,imag;

complex(int x,int y)

{

real=x;

imag=y;

}

complex()

{}

complex complexsum(complex c1,complex c2)

{

complex c3=new complex();

c3.real=c1.real+c2.real;

c3.imag=c1.imag+c2.imag;

return(c3);

}

complex complexsub(complex c1,complex c2)

{

complex c3=new complex();

c3.real=c1.real-c2.real;

c3.imag=c1.imag-c2.imag;

return(c3);

}

double conjugate(int num1,int num2)

{

return(Math.sqrt(num1\*num1+num2\*num2));

}

}

class Complexx

{

public static void main(String args[])

{

Scanner num=new Scanner(System.in);

int real1,real2,imag1,imag2,choice;

System.out.println("enter 1st complex no :");

real1=num.nextInt();

imag1=num.nextInt();

System.out.println("enter 2nd complex no :");

real2=num.nextInt();

imag2=num.nextInt();

complex c1=new complex(real1,imag1);

complex c2=new complex(real2,imag2);

complex c3=new complex();

do

{

System.out.println("enter the choice\n1 for sum\n2 for sub\n3 for exit\n");

choice=num.nextInt();

switch(choice)

{

case 1: c3=c3.complexsum(c1,c2);

System.out.println(c3.real+"+"+c3.imag+"i");

System.out.println("conjugate of sum is "+c3.conjugate(c3.real,c3.imag));

break;

case 2: c3=c3.complexsub(c1,c2);

System.out.println(c3.real+"+"+c3.imag+"i");

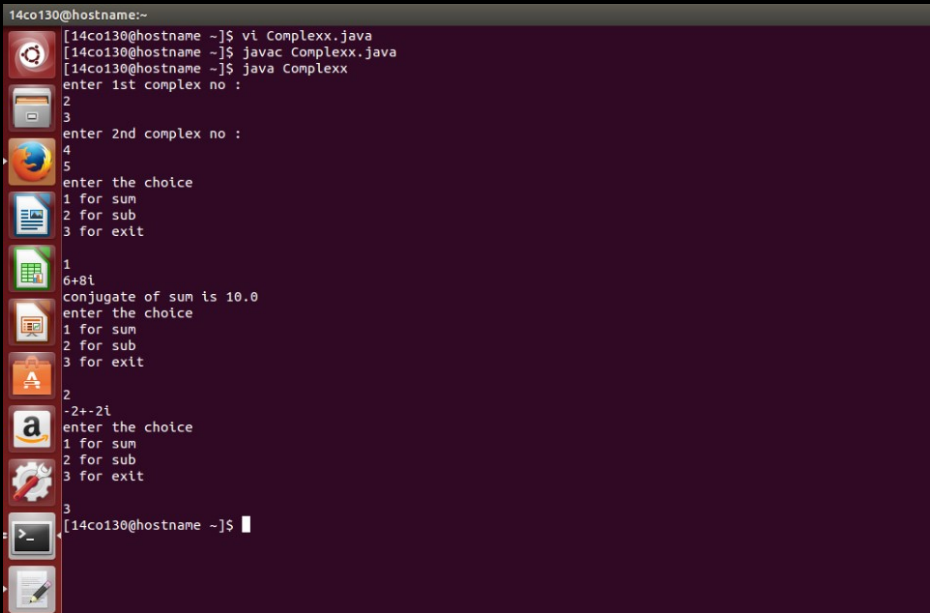
}

}while(choice!=3);

}

}

**OUTPUT**



**6. Create a class Person with the following properties:**

**Public fields:** ID, Name, YearOfBirth

Now take input (ID, Name, YearOfBirth) from the user and print it in the format :

ID | Name | Age

import java.io.\*;

import java.util.\*;

import java.text.\*;

import java.lang.\*;

class Date{

private int id;

private String str;

private int yob;

public Date()

{

}

public Date(int i1,String s,int y)

{

this.id=i1;

this.str=s;

this.yob=y;

}

public Date(String s,int y)

{

this.str=s;

this.yob=y;

Random rn = new Random();

int i = rn.nextInt()%100000;

this.id=i;

}

public void setId(int a)

{

id=a;

}

public int getId()

{

return id;

}

public void setName(String s)

{

str=s;

}

public String getName()

{

return str;

}

public void setYOB(int y)

{

yob=y;

}

public int getYOB()

{

return yob;

}

public int getAge()

{

int year = Calendar.getInstance().get(Calendar.YEAR);

return year=year-getYOB();

}

public void CopyTo(Date a)

{

a.id=this.id;

a.str=this.str;

a.yob=this.yob;

}

public void display()

{

//int year = Calendar.getInstance().get(Calendar.YEAR);

//year=year-this.yob;

System.out.println(getId()+"|"+getName()+"|"+getAge());

}

}

public class Age{

public static void main(String args[])

{

Scanner sc=new Scanner(System.in);

int a,b,ch;

String st,x;

Date d=new Date();

while(true){

System.out.println("1.With id 2.Without id 3.Copy 4.Exit\n");

ch=sc.nextInt();

if(ch==1)

{

System.out.println("Enter id");

a=sc.nextInt();

x=sc.nextLine();

System.out.println("Enter name");

st=sc.nextLine();

System.out.println("Enter year of birth");

b=sc.nextInt();

d.setId(a);

d.setName(st);

d.setYOB(b);

//Date n=new Date();

//d.year();

d.display();

}

else if(ch==2)

{

x=sc.nextLine();

System.out.println("Enter name");

st=sc.nextLine();

System.out.println("Enter year of birth");

b=sc.nextInt();

Date z=new Date(st,b);

z.display();

}

else if(ch==3)

{

Date n= new Date();

d.CopyTo(n);

n.display();

}

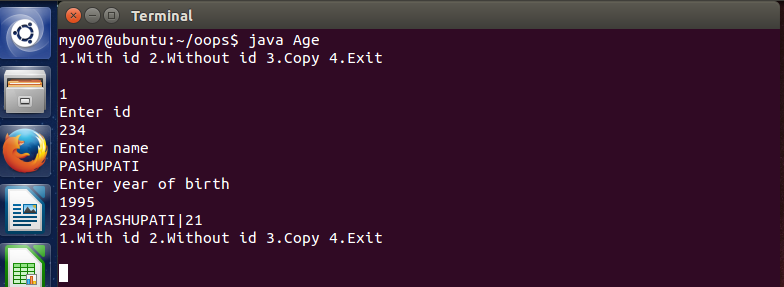
else

break;

}}

}

**OUTPUT**

****

**7. Create a BankAccount class.**

Some of the fields can be **CustomerID, Amount, AccountType (Current, Saving).**

The customer must deposit some amount (>= Rs 1000) while opening the account. The account can never have Zero balance.   
The valid operations are **deposit, withdraw, checkBalance, closeAccount.**

In your BankAccount class (from the previous lab), Add a new field**password**. Modify the constructors and methods accordingly.

Add the following restrictions:

(i) A current/saving account holder should not be able to deposit more than Rs 5000/Rs 10000  at a time respectively.

(ii) if three successive withdraw operations are performed at a time. The account should get Blocked. For any next Withdraw user should be asked to enter the password. After that account should again become unblocked.

 From your**BankAccount** class (Prev labs), derive two classes**SavingBankAccount, CurrentBankAccount.**

Make the appropriate changes in your code.

Create a package **Bank**which includes **BankAccount, SavingBankAccount, and CurrentBankAccount**classes.

package Bank;

import java.util.\*;

class BankAccount {

private String CustId;

private int balance;

private String password;

public int countw;

int checkBalance() {

return balance;

}

boolean deposit(int amount) {

balance += amount;

countw = 0;

return true;

}

boolean withdraw(int amount) {

balance -= amount;

countw++;

return true;

}

String getType(){

return "null";

}

BankAccount(String id,int bal,String pass) {

CustId = id;

balance = bal;

countw = 0;

password = pass;

}

boolean checkPassword(String pass) {

if(pass.equals(password))

return true;

else

return false;

}

}

class SavingsBankAccount extends BankAccount {

SavingsBankAccount(String id,int bal,String pass) {

super(id,bal,pass);

}

String getType(){

return "Savings";

}

boolean deposit(int amount) {

if (amount > 10000)

return false;

super.deposit(amount);

return true;

}

}

class CurrentAccount extends BankAccount {

CurrentAccount(String id,int bal,String pass) {

super(id,bal,pass);

}

String getType(){

return "Current";

}

boolean deposit(int amount) {

if (amount > 5000)

return false;

super.deposit(amount);

return true;

}

}

class Bank {

public static void main(String args[]) {

String id,password;

int balance,ch;

System.out.print("Enter the customer id : ");

Scanner s = new Scanner(System.in);

id = s.nextLine();

System.out.print("Enter the opening balance : ");

balance = s.nextInt();

while(balance < 1000)

{

System.out.println("Insufficient balance.");

System.out.print("Enter the opening balance : ");

balance = s.nextInt();

}

System.out.print("Enter the account type : \n 1. Savings \n 2. Current : " );

ch = s.nextInt();

System.out.print("Enter the password : ");

s.nextLine();

password = s.nextLine();

BankAccount a = null;

boolean flag = true;

switch(ch) {

case 1:

a = new SavingsBankAccount(id,balance,password);

break;

case 2:

a = new CurrentAccount(id,balance,password);

break;

default:

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

flag = false;

}

while(flag) {

System.out.print(" 1. Check Balance \n 2. Deposit \n 3. Withdraw \n 4. Exit \n Enter the choice : ");

ch = s.nextInt();

switch(ch) {

case 1 : ch = a.checkBalance();

System.out.println("Current Balance : " + ch );

s.nextLine();

break;

case 2 :System.out.print("Enter the amount to be deposited : ");

int amount = s.nextInt();

if(!a.deposit(amount)){

System.out.println("Amount exceeds the limit ");

s.nextLine();

break;

}

ch = a.checkBalance();

System.out.println("Current Balance : " + ch );

s.nextLine();

s.nextLine();

break;

case 3 :System.out.print("Enter the amount to be withdrawn : ");

amount = s.nextInt();

if(amount > a.checkBalance() ) {

System.out.print("Insufficient balance.");

s.nextLine();

}

else {

if(a.countw == 3)

{

s.nextLine();

System.out.print(" Enter the password : ");

String pass = s.nextLine();

if(a.checkPassword(pass)){

a.countw = 0;

}

else {

System.out.print("Invalid Password");

s.nextLine();

break;

}

}

a.withdraw(amount);

ch = a.checkBalance();

System.out.println("Current Balance : " + ch );

s.nextLine();

}

break;

case 4 : flag = false;

break;

default : System.out.println("Invalid choice.");

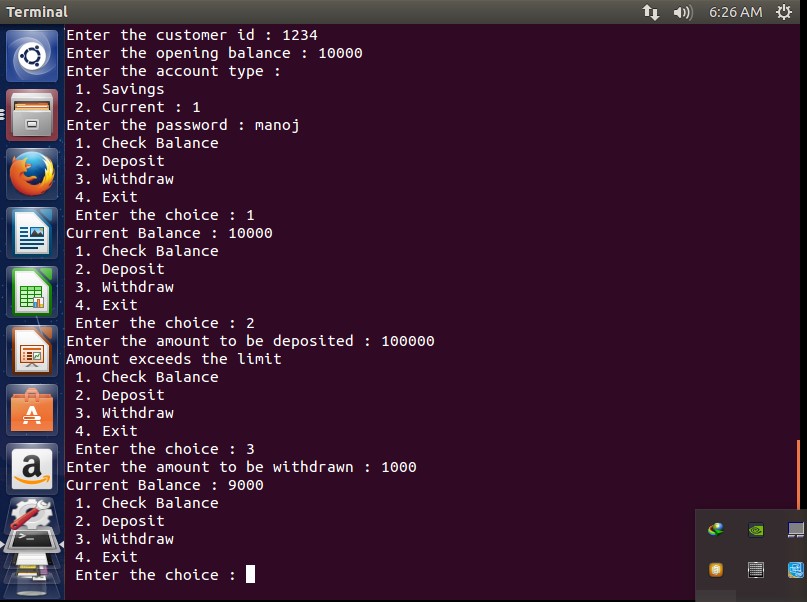
}

}

}

}

**OUTPUT**



8.Create a class of pizza with following fields:

Add topping,remove topping,get cost

import java.io.\*;

import java.util.\*;

public class Piza

{

public static void main(String args[])

{

int m,p,i,j;

String s;

Scanner sc=new Scanner(System.in);

Pizza ctor=new Pizza(3,2);

System.out.println("enter the Nooftoppings to be add");

m=sc.nextInt();

ctor.AddToppings(m);

System.out.println("enter the NoofToppings to be remove");

p=sc.nextInt();

ctor.Remove(p);

int r=ctor.getcost();

System.out.println("cost="+r);

s=sc.nextLine();

System.out.println("have u coupen if u have press 1 if u dont have press 0");

j=sc.nextInt();

if(j==1)

{

switch(j)

{

case 1:

{

int f=ctor.getcost(s);

System.out.println("cost after discount="+f);

}

}

}

else if(j==0)

{

System.out.println("cost="+r);

}

}

}

class Pizza

{

int thickness,Nooftoppings;

int cost;

Pizza(int x,int y)

{

thickness=x;

Nooftoppings=y;

}

public void AddToppings(int count)

{

Nooftoppings=Nooftoppings+count;

System.out.println("Nooftopping="+Nooftoppings);

}

public void Remove(int count)

{

Nooftoppings=Nooftoppings-count;

if(Nooftoppings<0)

{

Nooftoppings=0;

}

System.out.println("Nooftopping="+Nooftoppings);

}

public int getcost()

{

/\* if(Nooftoppings>0)

{

Nooftoppings=0;

}

\*/

cost=thickness\*100+Nooftoppings\*500;

// return(cost);

return(cost);

}

public int getcost(String s)

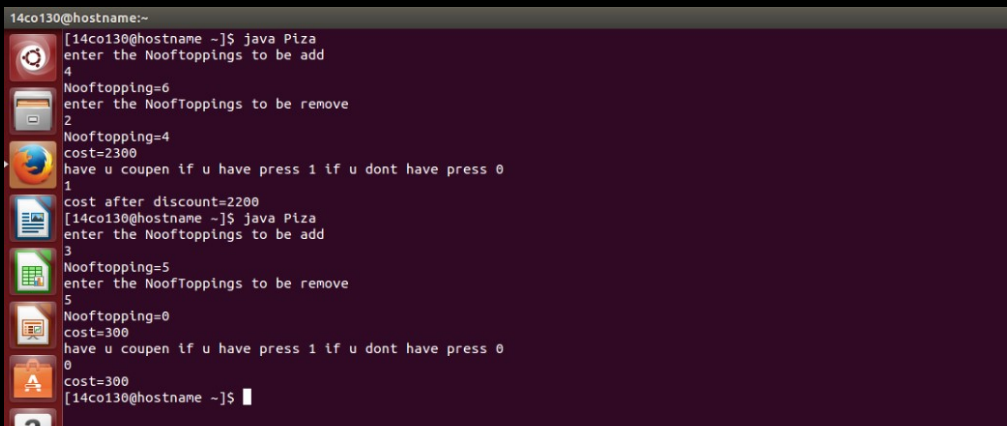
{

cost=cost-100;

return(cost);

}

}



**9.Create a base class Shape with the following methods:**

**double getArea() -** return the area of the shape.  
**double getPerimeter() -** return the perimeter of the shape.  
**void Draw() -** print the shape type like circle, rectangle etc.

The definitions of these functions in**Shape** class can include some default logic. e.g. in this case you can return a 0 from the first two methods and print some default value in 3rd method.

Now derive two classes from the Shape class.  
**Circle** with a field Radius  
**Rectangle** with fields length and breadth.  
These classes should override the methods defined in the Shape class.

Create a menu driven program to allow user to create new shapes and do operations on them.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Q - Now derive a class**Square** from the class**Rectangle.**  
From the constructor of Square class, call the constructor of Rectangle class.  
Do not override first two methods in this case, because the logic of finding area and perimeter is same for both**Square** and**Rectangle**.

Add the option of creating a **Square** in your menu.

import java.io.\*;

import java.util.\*;

public class Shapee {

public static void main(String args[])

{

int re,l,b,j,i,h;

Scanner sc=new Scanner(System.in);

Circle c=new Circle();

Rectangle r=new Rectangle();

Square s=new Square();

Cylinder ce=new Cylinder();

System.out.println("enter choise \n1 choose shape");

i=sc.nextInt();

switch(i)

{

case 1:

{

System.out.println("enter ur choise\n1 circle\n2 Rectangle\n3 Square\n4 Cylinder");

j=sc.nextInt();

switch(j)

{

case 1:

{

System.out.println("enter the radius");

re=sc.nextInt();

c.circle(re);

break;

}

case 2:

{

System.out.println("enter the length and breadth");

l=sc.nextInt();

b=sc.nextInt();

r.rec(l,b);

break;

}

case 3:

{

System.out.println("enter length");

l=sc.nextInt();

s.sqe(l);

break;

}

case 4:

{

System.out.println("enter height of cylinder and radius ");

h=sc.nextInt();

int ree=sc.nextInt();

ce.cylinder(h,ree);

break;

}

}

}

case 2:

{

System.out.println("enter ch \n1 find area and perimeter of circle\n2 find area and perimeter of rectangle\n 3 find area and perimeter of square\n 4 find area and volume of cylinder");

int k=sc.nextInt();

switch(k)

{

case 1:

{

System.out.println("Area of circle="+c.area+"perimeter of circle="+c.perimeter);

break;

}

case 2:

{

System.out.println("Area of Rectangle="+r.area+"perimeter of Rectangle="+r.perimeter);

break;

}

case 3:

{

System.out.println("Area of square="+s.area+"perimete="+s.perimeter);

break;

}case 4:

{

System.out.println("Area of cylinder="+ce.sa+"volume="+ce.vol);

break;

}

}

}

}

}

}

class Shape

{

double area,perimeter;

double getare()

{

return(area);

}

double getperi()

{

return(perimeter);

}

}

class Circle extends Shape

{

int r;

void circle(int x)

{

r=x;

area=3.14\*r\*r;

perimeter=2\*3.14\*r;

}

}

class Rectangle extends Shape

{

int x,y;

void rec(int l,int b)

{

x=l;

y=b;

area=(double)x\*y;

perimeter=(double)2\*(x+y);

}

}

class Square extends Shape

{

int l;

void sqe(int x)

{

l=x;

area=(double)x\*x;

perimeter=(double)2\*(x+x);

}

}

class Cylinder extends Circle

{

int h,radius;

double vol;

double sa;

//circle(radius);

void cylinder(int x, int y)

{

h=x;

radius=y;

circle(radius);

sa=h\*perimeter;

vol=h\*area;

}

double getsa()

{

return(sa);

}

double getvol()

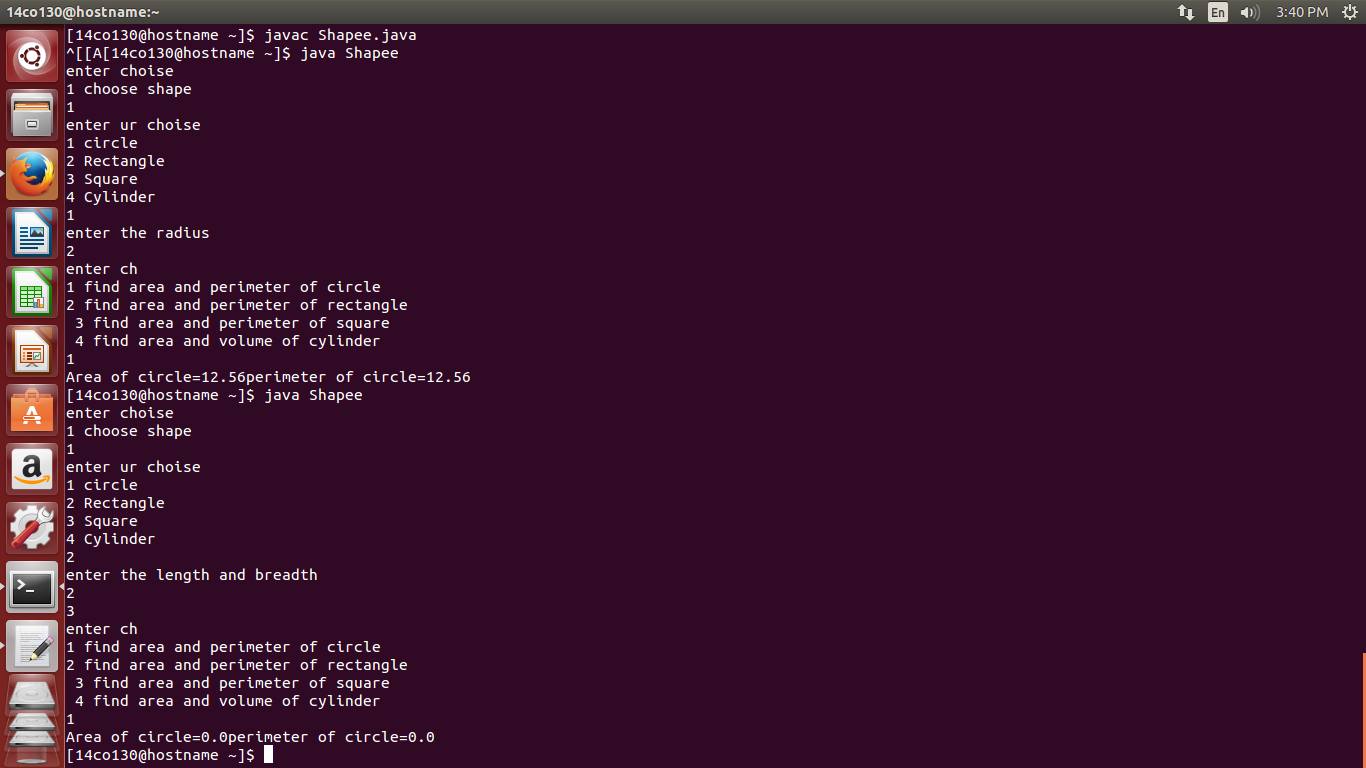
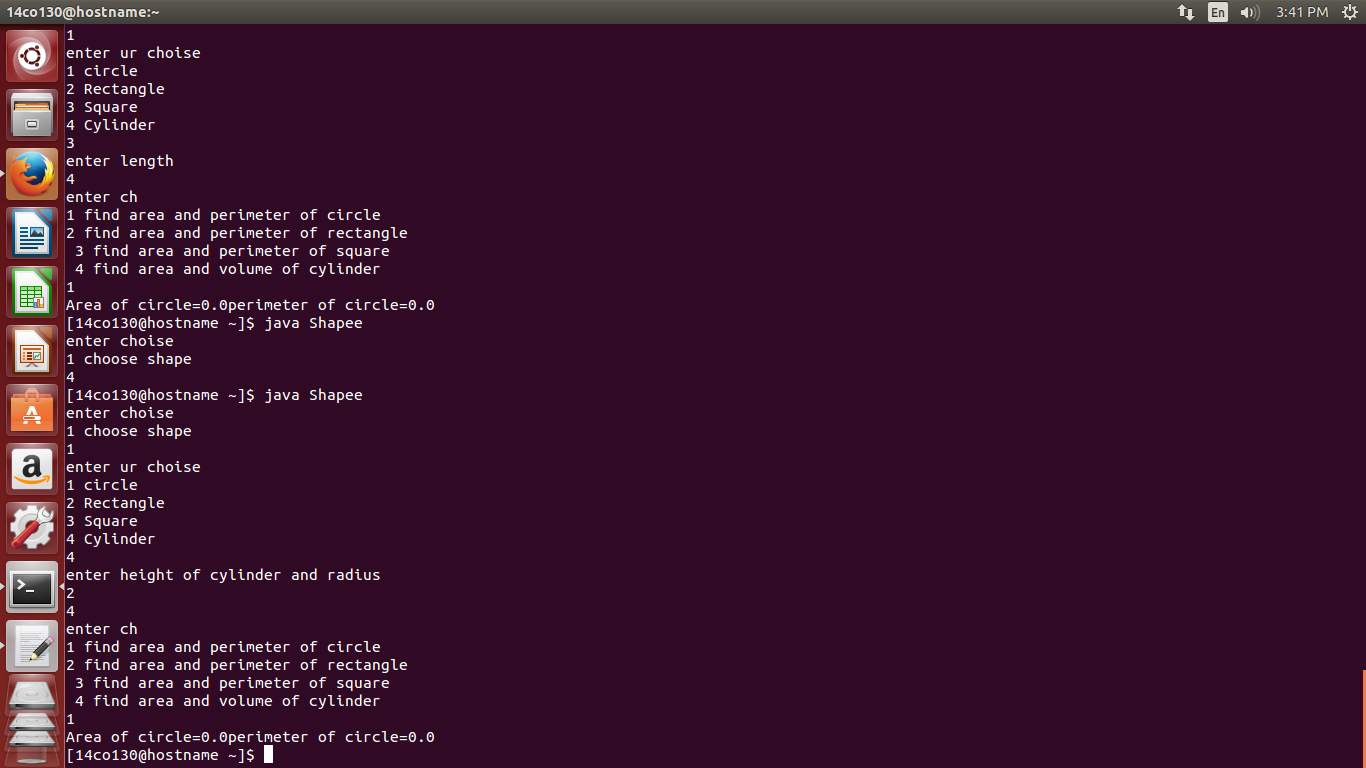
{

return(vol);

}

}

**OUTPUT**

**10. Create an interface Vehicle with the methods:**

turnIgnitionOn(), turnIgnitionOff(), IncreaseSpeed(int amt), DecreaseSpeed(int amt),  GetSpeed()

Now create a class **Car** that implements this interface.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Q2 - Create another interface **Flyable** with the following methods:

takeOff(), land()

Now create a class **Helicopter** that will implement both **Vehicle**and **Flyable** interfaces.

**Note** -   
  
In the definition of the functions, you can just print "Taking off..." / "Landing..."

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Q3 - Create a package named **vehicles.**And include the above interfaces and classes in that package. Now import this package and use the classes in a new file.

Vehicle.java

interface Vehicle{

public void turnIgnitionOn();

public void turnIgnitionOff();

public void IncreaseSpeed(int amt);

public void DecreaseSpeed(int amt);

public int GetSpeed();

}

Car.java

import java.util.\*;

public class Car implements Vehicle{

int ign;

int speed;

public void turnIgnitionOn()

{

ign=1;

speed=0;

}

public void turnIgnitionOff()

{

ign=0;

speed=0;

}

public void IncreaseSpeed(int amt)

{

if(ign==1)

speed+=amt;

}

public void DecreaseSpeed(int amt)

{

if(ign==1)

{

if(speed-amt>=0)

speed-=amt;

else

System.out.println("Underflow of speed");

}

}

public int GetSpeed()

{

return speed;

}

public static void main(String args[])

{

Scanner sc=new Scanner(System.in);

int a;

Car c=new Car();

while(true)

{

System.out.println("1.Turn ignition on\n2.Turn ignition off \n3.Increase speed\n4.Decrease speed\n5.Get speed\n6.Exit");

int ch=sc.nextInt();

if(ch==1)

c.turnIgnitionOn();

else if(ch==2)

c.turnIgnitionOff();

else if(ch==3)

{

System.out.println("Enter speed increase amount:");

a=sc.nextInt();

c.IncreaseSpeed(a);

}

else if(ch==4)

{

System.out.println("Enter speed decrease amount:");

a=sc.nextInt();

c.DecreaseSpeed(a);

}

else if(ch==5)

{

System.out.println("Speed: "+c.GetSpeed());

}

else

break;

}

}

}

Flyable.java

interface Flyable{

public void takeOff();

public void land();

}

Helicopter.java

import java.util.\*;

public class Helicopter implements Vehicle,Flyable{

int ign;

int speed;

public void takeOff()

{

System.out.println("Taking off...");

}

public void land()

{

System.out.println("Landing...");

}

public void turnIgnitionOn()

{

ign=1;

speed=0;

//System.out.println("Taking off...");

}

public void turnIgnitionOff()

{

ign=0;

speed=0;

//System.out.println("Landing...");

}

public void IncreaseSpeed(int amt)

{

if(ign==1)

speed+=amt;

}

public void DecreaseSpeed(int amt)

{

if(ign==1)

{

if(speed-amt>=0)

speed-=amt;

else

System.out.println("Underflow of speed");

}

}

public int GetSpeed()

{

return speed;

}

public static void main(String args[])

{

Scanner sc=new Scanner(System.in);

int a;

Helicopter h=new Helicopter();

while(true)

{

System.out.println("1.Turn ignition on\n2.Turn ignition off \n3.Increase speed\n4.Decrease speed\n5.Get speed\n6.Exit");

int ch=sc.nextInt();

if(ch==1)

{

h.turnIgnitionOn();

h.takeOff();

}

else if(ch==2)

{

h.turnIgnitionOff();

h.land();

}

else if(ch==3)

{

System.out.println("Enter speed increase amount:");

a=sc.nextInt();

h.IncreaseSpeed(a);

}

else if(ch==4)

{

System.out.println("Enter speed decrease amount:");

a=sc.nextInt();

h.DecreaseSpeed(a);

}

else if(ch==5)

{

System.out.println("Speed: "+h.GetSpeed());

}

else

break;

}

}

}

Driver.java

import vehicles.\*;

import java.util.\*;

public class Driver{

public static void main(String args[])

{

Scanner sc=new Scanner(System.in);

int a;

System.out.println("Drive\n1.Car\n2.Helicopter\nEnter ur choice:");

int ch1=sc.nextInt();

if(ch1==1)

{

Car c=new Car();

while(true)

{

System.out.println("1.Turn ignition on\n2.Turn ignition off \n3.Increase speed\n4.Decrease speed\n5.Get speed\n6.Exit");

int ch=sc.nextInt();

if(ch==1)

c.turnIgnitionOn();

else if(ch==2)

c.turnIgnitionOff();

else if(ch==3)

{

System.out.println("Enter speed increase amount:");

a=sc.nextInt();

c.IncreaseSpeed(a);

}

else if(ch==4)

{

System.out.println("Enter speed decrease amount:");

a=sc.nextInt();

c.DecreaseSpeed(a);

}

else if(ch==5)

{

System.out.println("Speed: "+c.GetSpeed());

}

else

break;

}

}

else

{

Helicopter h=new Helicopter();

while(true)

{

System.out.println("1.Turn ignition on\n2.Turn ignition off \n3.Increase speed\n4.Decrease speed\n5.Get speed\n6.Exit");

int ch=sc.nextInt();

if(ch==1)

{

h.turnIgnitionOn();

h.takeOff();

}

else if(ch==2)

{

h.turnIgnitionOff();

h.land();

}

else if(ch==3)

{

System.out.println("Enter speed increase amount:");

a=sc.nextInt();

h.IncreaseSpeed(a);

}

else if(ch==4)

{

System.out.println("Enter speed decrease amount:");

a=sc.nextInt();

h.DecreaseSpeed(a);

}

else if(ch==5)

{

System.out.println("Speed: "+h.GetSpeed());

}

else

break;

}

}

}

}

**OUTPUT**

