T.Y.B.Sc. Computer Science Practical

SEMESTER – V

(Core Java)

Slip-1

Q1) Write a Program to print all Prime numbers in an array of ‘n’ elements. (use command line arguments)

→

public class slip1a\_prime

{ public static void main(String[] args)

{ int cnt = 0; int i = 0;

int flag = 0; int n=args.length; int intArr[]=new int[n]; for(int j=0;j<n;j++)

{ intArr[j] = Integer.parseInt(args[j]);

} for (cnt = 0; cnt < intArr.length; cnt++)

{ flag = 0; for (i = 2; i <=intArr[cnt] / 2; i++)

{ if (intArr[cnt] % i == 0)

{ flag = 1;

break;

}

}

if(flag==0)

System.out.println(" Prime No. is :- "+intArr[cnt]);

}

}

}

Q2) Define an abstract class Staff with protected members id and name. Define a parameterized constructor. Define one subclass OfficeStaff with member department. Create n objects of OfficeStaff and display all details.

→

import java.util.\*; abstract class Staff

{ protected int id; protected String name;

public Staff(int sid,String nm)

{ id=sid;

name=nm;

}

abstract public void display();

} class officeStaff extends Staff

{

String dept;

officeStaff(int sid,String nm,String dpt)

{ super(sid,nm);

dept=dpt;

} public void display()

{

System.out.println("Id:"+id);

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

System.out.println("Department:"+dept);

} }

public class slip1b\_staff

{ public static void main(String g[])

{

Scanner sc=new Scanner(System.in); System.out.println("Enter number of staff"); int n=sc.nextInt();

officeStaff[] obj=new officeStaff[n]; for(int i=0;i<n;i++)

{

System.out.println("Enter id,name,department");

int oid=sc.nextInt();

String onm=sc.next(); String odept=sc.next(); obj[i]=new officeStaff(oid,onm,odept); }

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

obj[i].display();

}

}

Slip -2

Q1) Write a program to read the First Name and Last Name of a person, his weight and height using command line arguments. Calculate the BMI Index which is defined as the individual's body mass divided by the square of their height.

(Hint : BMI = Wts. In kgs / (ht)2)

→

import java.util.Scanner; public class slip2a\_BMI {

public static void main(String args[])

{

String fnm=args[0]; String lnm=args[1]; double weight=Double.parseDouble(args[2]);

double height=Double.parseDouble(args[3]);

double BMI = weight / (height \* height);

System.out.print("The Body Mass Index (BMI) is " + BMI + " kg/m2");

}

}

Q2) Define a class CricketPlayer (name,no\_of\_innings,no\_of\_times\_notout, totatruns, bat\_avg). Create an array of n player objects .Calculate the batting average for each player using static method avg(). Define a static sort method which sorts the array on the basis of average. Display the player details in sorted order.

→

import java.io.\*;

public class CricketPlayer

{

String name; int runs,no\_of\_innigs,no\_of\_times\_notout; float bat\_avg; public CricketPlayer()

{

} public static CricketPlayer[] sortPlayer(CricketPlayer [] player)

{

CricketPlayer temp; for(int i=0;i<player.length;i++)

{ for(int j=i+1;j<player.length;j++)

{ if(player[i].bat\_avg<player[i].bat\_avg)

{

temp=player[i]; player[i]=player[j];

player[j]=temp;

}

}

}

return player;

} public static CricketPlayer[] avg(CricketPlayer [] players)

{

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

{

players[i].bat\_avg=players[i].runs/players[i].no\_of\_innigs;

}

return players;

}

public void accept() throws Exception

{

BufferedReader br=new BufferedReader(new

InputStreamReader(System.in));

System.out.println("Enter player Name:"); name=br.readLine();

System.out.println("Enter Number of Innings:\t"); no\_of\_innigs=Integer.parseInt(br.readLine());

if(no\_of\_innigs<=0)

System.out.println("Innigs cant be less than zero");

System.out.println("Enter Total Runs:\t"); runs=Integer.parseInt(br.readLine());

if(runs<0)

System.out.println("Runs cant be less than negative");

System.out.println("Enter Number of time notout:\t"); no\_of\_times\_notout=Integer.parseInt(br.readLine());

if(no\_of\_times\_notout>no\_of\_innigs)

System.out.println("Number of notout cant not be greater

than no of innigs");

}

public static void main(String[] args) throws Exception

{ BufferedReader br=new BufferedReader(new

InputStreamReader(System.in));

CricketPlayer players[];

System.out.println("How many players you want?:\t");

int n=Integer.parseInt(br.readLine()); if(n<0)

System.out.println("N cant not be negative");

else {

players=new CricketPlayer[n]; for(int i=0;i<n;i++)

{ players[i]=new CricketPlayer(); players[i].accept();

}

CricketPlayer.avg(players);

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

System.out.println("Name\t\tInnings\t\tRuns\t\tNotout\t\tBat Avg");

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

{

System.out.print(players[i].name+"\t\t"+players[i].no\_of\_innigs+"\t\t");

System.out.print(players[i].runs+"\t\t"+players[i].no\_of\_times\_notout+"\t\t");

System.out.println(players[i].bat\_avg);

}

CricketPlayer.sortPlayer(players);

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

System.out.println("Name\t\tInnings\t\tRuns\t\tNotout\t\tBat Avg");

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

{

System.out.print(players[i].name+"\t\t"+players[i].no\_of\_innigs+"\t\t");

System.out.print(players[i].runs+"\t\t"+players[i].no\_of\_times\_notout+"\t\t");

System.out.println(players[i].bat\_avg);

}

}

}

}

Slip -3

Q1) Write a program to accept ‘n’ name of cities from the user and sort them in ascending order.

→

import java.util.Scanner;

class sortCity

{ public static void main(String args[])

{

String temp;

Scanner SC = new Scanner(System.in);

System.out.print("Enter the value of N : "); int N= SC.nextInt();

SC.nextLine();

String names[] = new String[N];

System.out.println("Enter names of cities: "); for(int i=0; i<N; i++)

{

System.out.print("Enter name [ " + (i+1) +" ]: "); names[i] = SC.nextLine();

}

for(int i=0; i<N; i++)

{ for(int j=1; j<N; j++)

{

if(names[j-1].compareTo(names[j])>0)

{

temp=names[j-1]; names[j-1]=names[j]; names[j]=temp;

}

}

}

System.out.println("\nSorted city names are in Ascending Order: "); for(int i=0;i<N;i++)

{

System.out.println(names[i]);

}

}

}

Q2) Define a class patient (patient\_name, patient\_age, patient\_oxy\_level,patient\_HRCT\_report). Create an object of patient. Handle appropriate exception while patient oxygen level less than 95% and HRCT scan report greater than 10, then throw user defined Exception “Patient is Covid

Positive(+) and Need to Hospitalized” otherwise display its information.

→

import java.io.\*;

class report extends Exception

{ report()

{

System.out.println("Patient is Covid Positive(+) andNeed to Hospitalized");

} } class patient

{

String name; int age, oxylevel,HRCTreport; patient(String name, int age, int oxylevel, int HRCTreport)

{

this.name = name;

this.age = age; this.oxylevel = oxylevel;

this.HRCTreport = HRCTreport;

}

public static void main(String[] args) throws Exception

{

BufferedReader br=new BufferedReader(new InputStreamReader(System.in));

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

String name = br.readLine();

System.out.println("Enter Age "); int age = Integer.parseInt(br.readLine()); System.out.println("Enter oxygen level"); int oxylevel =Integer.parseInt(br.readLine());

System.out.println("Enter HRCT report"); int HRCTreport = Integer.parseInt(br.readLine());

patient obj = new patient(name, age, oxylevel, HRCTreport);

if(obj.oxylevel < 95 && obj.HRCTreport > 10)

{ throw new report();

} else

{

System.out.println("name: "+obj.name);

System.out.println("age " + obj.age);

System.out.println("oxygen level " +obj.oxylevel);

System.out.println("HRCT report " + obj.HRCTreport);

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

}

}

}

Slip -4

Q1) Write a program to print an array after changing the rows and columns of a given two-dimensional array. [10 marks]

→

import java.util.Scanner;

public class slip4a {

public static void main(String[] args) {

int[][] twodm = {

{10, 20, 30},

{40, 50, 60}

};

System.out.print("Original Array:\n"); print\_array(twodm);

System.out.println("After changing the rows and columns of the said array:");

transpose(twodm);

}

private static void transpose(int[][] twodm) {

int[][] newtwodm = new int[twodm[0].length][twodm.length];

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

for (int j = 0; j < twodm[0].length; j++) {

newtwodm[j][i] = twodm[i][j];

}

}

print\_array(newtwodm);

}

private static void print\_array(int[][] twodm) {

for (int i = 0; i < twodm.length; i++) { for (int j = 0; j < twodm[0].length; j++) { System.out.print(twodm[i][j] + " ");

}

System.out.println();

}

}

}

Q2) Write a program to design a screen using Awt that will take a user name and password. If the user name and password are not same, raise an Exception with appropriate message. User can have 3 login chances only. Use clear button to clear the TextFields. .

→

import java.awt.\*; import java.awt.event.\*; class InvalidPasswordException extends Exception

{

InvalidPasswordException()

{

System.out.println("User name and Password is not same");

} }

public class ass5login extends Frame implements ActionListener

{

Label uname,upass;

TextField nametext,passtext,msg; Button login,Clear; int attempt=0;

char c='\*';

ass5login()

{

uname=new Label("User Name:" ,Label.CENTER); upass=new Label ("Password: ",Label.RIGHT);

nametext=new TextField(20); passtext =new TextField(20); passtext.setEchoChar(c); msg=new TextField(10);

login=new Button("Login"); Clear=new Button("Clear"); login.addActionListener(this); Clear.addActionListener(this); setLayout(new GridLayout(4,2)); add(uname); add(nametext); add(upass); add(passtext); add(login); add(Clear); add(msg);

setTitle("Login "); setSize(300,200); setVisible(true);

}

public void actionPerformed(ActionEvent ae)

{ if(attempt<3) { if(ae.getSource()==Clear)

{ nametext.setText(""); passtext.setText("");

} if(ae.getSource()==login)

{ try

{

String user=nametext.getText(); String upass=passtext.getText();

if(user.compareTo(upass)==0)

{ msg.setText("Valid");

System.out.println("Username is valid");

} else {

throw new InvalidPasswordException();

} } catch(Exception e)

{

msg.setText("Error");

}

attempt++;

} } else

{

System.out.println("you are using 3 attempt");

System.exit(0);

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

{ new ass5login();

}}

Slip - 5

Q1) Write a program for multilevel inheritance such that Country is inherited from Continent. State is inherited from Country. Display the place, State, Country and Continent. [10 marks]

→

import java.util.\*; class continent {

String con;

Scanner sc=new Scanner(System.in); void cinput()

{

System.out.println("Enter Continent Name: "); con =sc.next();

}

} class country extends continent

{ String cou; void couinput()

{

System.out.println("Enter Country Name: ");

cou = sc.next();

} } class state extends country

{ String sta; void stainput()

{

System.out.println("Enter State Name: ");

sta = sc.next();

} } class place extends state

{ String pla; void pinput()

{

System.out.println("Enter Place Name : ");

pla = sc.next(); }

public static void main( String arg[] )

{

place s = new place();

s.cinput();

s.couinput();

s.stainput();

s.pinput();

System.out.println("Place :" + s.pla);

System.out.println("State: "+s.sta);

System.out.println("Country: "+s.cou);

System.out.println("Continent: "+s.con);

}

}

Q2) Write a menu driven program to perform the following operations on multidimensional array ie matrices :

▪ Addition

▪ Multiplication

▪ Exit

→

import java.util.\*;

public class Matrix

{

public static void main(String[] args)

{

Scanner scan = new Scanner(System.in);

int a[][] = { { 5, 6, 7 }, { 8, 9, 10 }, { 3, 1, 2 } }; int b[][] = { { 1, 2, 3 }, { 4, 5, 6 }, { 7, 8, 9 } };

int c[][] = new int[3][3];

System.out.println("A = " + Arrays.deepToString(a)); System.out.println("B = " + Arrays.deepToString(b));

int choice;

do

{

System.out.println("\nChoose the matrix operation,");

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

System.out.println("1. Addition");

System.out.println("2. Multiplication");

System.out.println("3. Exit");

System.out.println("----------------------------"); System.out.print("Enter your choice: "); choice = scan.nextInt();

switch (choice)

{ case 1: c = add(a, b);

System.out.println("Sum of matrix: "); System.out.println(Arrays.deepToString(c)); break;

case 2:

c = multiply(a, b);

System.out.println("Multiplication of matrix: "); System.out.println(Arrays.deepToString(c)); break;

}

} while (choice!=3);

}

public static int[][] add(int[][] a, int[][] b)

{ int row = a.length; int column = a[0].length;

int sum[][] = new int[row][column];

for (int i = 0; i < row; i++) { for (int j = 0; j < column; j++) { sum[i][j] = a[i][j] + b[i][j];

}

}

return sum; } public static int[][] multiply(int[][] a, int[][] b)

{

int row = a.length;

int column = b[0].length;

int product[][] = new int[row][column];

for (int i = 0; i < row; i++) {

for (int j = 0; j < column; j++) {

product[i][j] = 0;

for (int k = 0; k < a[0].length; k++) { product[i][j] += a[i][k] \* b[k][j];

}

}

} return product;

}

}

Slip - 6

Q1) Write a program to display the Employee(Empid, Empname, Empdesignation, Empsal) information using toString().

→

class Emp

{ int id,salary;

String name;

String city;

Emp(int id, String name, int salary ,String city)

{ this.id=id;

this.name=name;

this.salary=salary;

this.city=city;

} public String toString()

{

return id+" "+name+" "+salary+" "+city;

} public static void main(String args[])

{

Emp E1=new Emp(11,"Sachin",50000,"Pune");

Emp E2=new Emp(12,"Priya",25000,"Mumbai"); System.out.println("Employee details: "+E1);

System.out.println("Employee details: "+E2);

}

}

Q2) Create an abstract class “order” having members id, description. Create two subclasses “PurchaseOrder” and “Sales Order” having members customer name and Vendor name respectively. Definemethods accept and display in all cases. Create 3 objects each of Purchase Order and Sales Order and accept and display details.

→

import java.util.\*; abstract class order

{ int id; String des; abstract void accept();

abstract void display();

} class porder extends order

{

String cname,vname;

void accept()

{

Scanner sc=new Scanner(System.in);

System.out.println("Enter Purchase Order details");

System.out.println("\nEnter id,description,customer name,vendor name"); id=sc.nextInt(); des=sc.next(); cname=sc.next();

vname=sc.next();

}

void display()

{

System.out.println("\nID:-"+id);

System.out.println("Description:-"+des);

System.out.println("customer name:"+cname);

System.out.println("vendor name:-"+vname);

} }

class sorder extends order

{

String custname; String vendname;

void accept()

{

Scanner sc=new Scanner(System.in);

System.out.println("\nEnter Sales Order details");

System.out.println("\nEnter id,description,sales customer name,sales vendor name");

id=sc.nextInt(); des=sc.next(); custname=sc.next(); vendname=sc.next();

}

void display()

{

System.out.println("\nSales ID:-"+id);

System.out.println("Sales Description:-"+des);

System.out.println("Sales customer name:"+custname);

System.out.println("Sales vendor name:-"+vendname);

}

public static void main(String[] a)

{

porder[] p=new porder[3];

for(int i=0;i<3;i++)

{ p[i]=new porder();

p[i].accept();

} sorder[] s=new sorder[3]; for(int i=0;i<3;i++)

{ s[i]=new sorder();

s[i].accept();

}

System.out.println("\n\nPurchase Order Details\n"); for(int i=0;i<3;i++)

{

p[i].display();

}

System.out.println("\n\nSales Order Details\n"); for(int i=0;i<3;i++)

{

s[i].display();

}

}

}

Slip - 7

Q1) Design a class for Bank. Bank Class should support following operations; a. Deposit a certain amount into an account

1. Withdraw a certain amount from an account
2. Return a Balance value specifying the amount with details

→

import java.util.Scanner; class BankDetails { private String accno; private String name; private String acc\_type; private long balance;

Scanner sc = new Scanner(System.in);

public void openAccount() {

System.out.print("Enter Account No: "); accno = sc.next();

System.out.print("Enter Account type: "); acc\_type = sc.next();

System.out.print("Enter Name: "); name = sc.next();

System.out.print("Enter Balance: "); balance = sc.nextLong();

}

public void showAccount() {

System.out.println("Name of account holder: " + name);

System.out.println("Account no.: " + accno);

System.out.println("Account type: " + acc\_type);

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

}

public void deposit() { long amt;

System.out.println("Enter the amount you want to deposit: "); amt = sc.nextLong();

balance = balance + amt;

System.out.println("Balance after deposit: " + balance);

}

public void withdrawal() { long amt;

System.out.println("Enter the amount you want to withdraw: "); amt = sc.nextLong(); if (balance >= amt) { balance = balance - amt;

System.out.println("Balance after withdrawal: " + balance);

} else {

System.out.println("Your balance is less than " + amt + "\tTransaction failed...!!" );

}

}

}

public class BankingApp {

public static void main(String arg[])

{

Scanner sc = new Scanner(System.in);

BankDetails C = new BankDetails();

C.openAccount();

int ch;

do {

System.out.println("\n \*\*\*Banking System Application\*\*\*");

System.out.println("1. Display all account details \n 2.Deposit the amount \n

3. Withdraw the amount \n 4.Exit ");

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

ch = sc.nextInt();

switch (ch) {

case 1:

C.showAccount();

break;

case 2:

C.deposit();

break;

case 3:

C.withdrawal(); break;

case 4:

System.out.println("See you soon...");

break;

} } while (ch != 4);

}

}

Q2) Write a program to accept a text file from user and display the contents of a file in reverse order and change its case.

→

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

public class ReverseFile

{ public static void main(String[] args) throws Exception

{

BufferedReader br=new BufferedReader(new

InputStreamReader(System.in));

System.out.println("Enter source and target file name");

String source = br.readLine();

String target = br.readLine();

File sourceFile=new File(source);

Scanner content=new Scanner(sourceFile); PrintWriter pwriter =new PrintWriter(target);

while(content.hasNextLine())

{

String s=content.nextLine();

StringBuffer buffer = new StringBuffer(s); buffer=buffer.reverse();

String rs=buffer.toString();

pwriter.println(rs);

} content.close(); pwriter.close();

System.out.println("File is copied successful!");

}

}

Slip - 8

Q1) Create a class Sphere, to calculate the volume and surface area of sphere. (Hint : Surface area=4\*3.14(r\*r), Volume=(4/3)3.14(r\*r\*r))

→

import java.io.\*; class Sphere

{ double r; Sphere(double radius)

{

r=radius; } void area()

{ double a; a=4\*3.14\*r\*r;

System.out.println("Area of Sphere:"+a);

}

void volume()

{ double v; v=(4/3)\*3.14\*r\*r\*r;

System.out.println("Volume of Sphere:"+v);

} public static void main(String g[])throws Exception

{

BufferedReader br=new BufferedReader(new

InputStreamReader(System.in));

System.out.println("Enter radius:"); double p=Double.parseDouble(br.readLine()); Sphere s1=new Sphere(p);

s1.area();

s1.volume();

}

}

Q2) Design a screen to handle the Mouse Events such as

MOUSE\_MOVED and MOUSE\_CLICKED and display the position of the Mouse\_Click in a TextField.

→

import java.awt.\*; import java.awt.event.\*; class MyFrame extends Frame

{

TextField t,t1; Label l,l1; int x,y;

Panel p;

MyFrame(String title)

{ super(title);

setLayout(new FlowLayout());

p=new Panel();

p.setLayout(new GridLayout(2,2)); t=new TextField(20); l= new Label("Co-ordinates of clicking"); l1= new Label("Co-ordinates of movement"); t1=new TextField(20);

p.add(l);

p.add(t);

p.add(l1);

p.add(t1); add(p); addMouseListener(new MyClick()); addMouseMotionListener(new MyMove()); setSize(500,500);

setVisible(true);

}

class MyClick extends MouseAdapter

{

public void mouseClicked(MouseEvent me)

{

x=me.getX(); y=me.getY();

t.setText("X="+x+" Y="+y);

}

}

class MyMove extends MouseMotionAdapter

{

public void mouseMoved(MouseEvent me)

{

x=me.getX(); y=me.getY();

t1.setText("X="+ x +" Y="+y);

} }

public static void main(String args[])

{

MyFrame f = new MyFrame("Mouse Motion");

}

}

Slip – 9

Q1) Define a “Clock” class that does the following ;

1. Accept Hours, Minutes and Seconds
2. Check the validity of numbers
3. Set the time to AM/PM mode

Use the necessary constructors and methods to do the above task

→

class clock

{ int hrs,mins,sec; clock(int h,int m,int s)

{

hrs=h;

mins=m;

sec=s; }

void isTimeValid()

{

if(hrs>=0 && hrs<24 && mins>0 && mins<60 && sec>0 && sec<600)

System.out.println("Time is Valid");

else

System.out.println("Time is not Valid");

}

void setTimeMode()

{

if(hrs<12)

System.out.println("Time-"+hrs+":"+mins+":"+sec+"AM");

else

{

hrs=hrs-12;

System.out.println("Time-"+hrs+":"+mins+":"+sec+"PM");

} }

public static void main(String[] args)

{

clock c=new clock(16,34,45);

c.isTimeValid();

c.setTimeMode();

}

}

Q2) Write a program to using marker interface create a class Product (product\_id, product\_name, product\_cost, product\_quantity) default and parameterized constructor. Create objectsof class product and display the contents of each object and Also display the object count.

→

import java.util.\*;

interface MarkerInt {

}

class product implements MarkerInt {

int pid, pcost, quantity; String pname;

static int cnt; // Default constructor

product() { pid = 1; pcost = 10; quantity = 1; pname = "pencil"; cnt++;

}

// Parameterized constructor

product(int id, String n, int c, int q) { pid = id;

pname = n;

pcost = c; quantity = q; cnt++;

System.out.println("\nCOUNT OF OBJECT IS : " + cnt + "\n");

}

public void display() {

System.out.println("\t" +pid + "\t" + pname + "\t" + pcost + "\t" +

quantity

);

} } public class productinterface

{ public static void main(String[] args)

{

Scanner sc = new Scanner(System.in);

System.out.println("Enter Number of Product : ");

int n = sc.nextInt();

product pr[] = new product[n];

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

System.out.println("\nEnter " + (i + 1) + " Product Details :\n");

System.out.println("Enter Product ID: "); int pid = sc.nextInt();

sc.nextLine();

System.out.println("Enter Product Name: ");

String pn = sc.nextLine();

System.out.println("Enter Product Cost:");

int pc = sc.nextInt();

System.out.println("Enter Product Quantity:"); int pq = sc.nextInt();

pr[i] = new product(pid, pn, pc, pq);

}

System.out.println("\n\t\t Product Details\n"); System.out.println("\tId\tPname\tCost\tQuantity\n");

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

{

pr[i].display();

}

}

}

Slip - 10

Q1) Write a program to find the cube of given number using functional interface.

→

interface PrintNumber

{ public void print(int num1);

} public class PrintCube

{ public static void main(String[] a)

{

PrintNumber p = n -> System.out.println("Cube is: "+ n\*n\*n); p.print(5);

}

}

Q2) Write a program to create a package name student. Define class StudentInfo with method to display information about student such as rollno, class, and percentage. Create another class StudentPer with method to find percentage of the student. Accept student details like rollno, name, class and marks of 6 subject from user.

→

package student; public class studinfo

{ public int r\_no; public String name;

public int a,b,c,d,e,f;

int sum=0; public studinfo(int roll, String nm, int m1,int m2,int m3,int m4,int m5,int m6)

{ r\_no = roll;

name = nm; a = m1; b = m2;

c = m3;

d=m4;

e=m5;

f=m6; sum = a+b+c+d+e+f;

} public void display()

{

System.out.println("Roll\_no : "+r\_no);

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

System.out.println("-----MARKS-------");

System.out.println("Sub 1 : "+a);

System.out.println("Sub 2 : "+b);

System.out.println("Sub 3 : "+c);

System.out.println("Sub 4 : "+d);

System.out.println("Sub 5 : "+e);

System.out.println("Sub 6 : "+f);

System.out.println("Total : "+sum);

System.out.println("percentage: "+sum/6);

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

} }

import student.\*; import java.util.\*; import java.io.\*; class studdetails

{ public static void main(String[] args)

{

String nm;

int roll;

Scanner sc = new Scanner(System.in); System.out.print("Enter Roll no:= ");

roll = sc.nextInt();

System.out.print("Enter Name:= "); nm = sc.next();

int m1,m2,m3,m4,m5,m6;

System.out.print("Enter 6 sub mark:= "); m1 = sc.nextInt(); m2 = sc.nextInt(); m3 = sc.nextInt();

m4 = sc.nextInt(); m5 = sc.nextInt(); m6 = sc.nextInt(); studinfo s = new studinfo( roll,nm,m1,m2,m3,m4,m5,m6);

s.display();

}

}

Slip - 11

Q1) Define an interface “Operation” which has method volume( ).Define a constant PI having a value 3.142 Create a class cylinder which implements this interface

(members – radius,height). Create one object and calculate the volume.

→

import java.util.\*; interface operation

{ final double pi=3.14; abstract void area();

abstract void volume();

}

class cylinder implements operation

{ double radius;

double height;

cylinder(double r,double h)

{

radius=r;

height=h;

}

public void area()

{ double ra= (2\*(pi\*(radius\*height)));

System.out.println("Area is:-"+ra);

} public void volume()

{

double heig=(pi\*(radius\*radius)\*(height));

System.out.println("Volume is:-"+heig);

}

public static void main(String a[])

{

Scanner sc=new Scanner(System.in);

System.out.println("Enter the radius and height"); double rad=sc.nextFloat(); double hei=sc.nextFloat();

cylinder obj=new cylinder(rad,hei); obj.area();

obj.volume();

}

}

Q2) Write a program to accept the username and password from user if username and password are not same then raise "Invalid Password" with appropriate msg.

→

import java.util.\*; class InvalidUserNameException extends Exception

{

InvalidUserNameException(String s)

{

super(s);

} }

class InvalidPasswordException extends Exception

{

InvalidPasswordException(String s)

{

super(s);

} } class EmailId

{

String username,password;

EmailId()

{

username="abc"; password="abc123";

}

EmailId(String user,String pass)

{

username=user; password=pass;

}

public static void main(String[] args)

{ String user,pass; int u=0,p=0;

EmailId obj=new EmailId();

user=args[0]; pass=args[1];

EmailId obj1=new EmailId(user,pass); try { if((obj.username).equals(obj1.username))

{

u=1;

} if((obj.password).equals(obj1.password))

{

p=1;

}

if(u==0) {

throw new InvalidUserNameException("Invalid User Name");

}

if(p==0) {

throw new InvalidPasswordException("Invalid Password");

}

}catch(Exception e)

{System.out.println(e);}

}

}

Slip - 12

Q1) Write a program to create parent class College(cno, cname, caddr) and derived class Department(dno, dname) from College. Write a necessary methods to

display College details.

→

import java.util.\*; class college

{ int cno;

String name;

String caddr; college(int cno,String name,String caddr)

{ this.cno=cno; this.name=name;

this.caddr=caddr;

} public void display()

{

System.out.println("College cno:"+cno);

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

System.out.println("College Address:"+caddr);

}

}

class department extends college

{

int dno;

String dnm;

department(int cno,String name,String caddr,int dno,String dnm)

{

super(cno,name,caddr); this.dno=dno; this.dnm=dnm;

}

public void display()

{ super.display();

System.out.println("Department number:"+dno);

System.out.println("Department Name:"+dnm);

}

public static void main(String g[])

{

department obj=new department(101,"MMCC","Deccan",201,"Computer

Science");

department obj1=new department(102,"SPPU","Pune",301,"Computer

Science"); obj.display(); obj1.display();

}

}

Q2) Write a java program that works as a simple calculator. Use a grid layout to arrange buttons for the digits and for the +, -, \*, % operations. Add a text field to display the result.

|  |
| --- |
| Simple Calculator |
| 1 2 3 +  4 5 6 7 8 9 \*  0 . = / |

→

import java.awt.\*; import javax.swing.\*; import java.awt.event.\*; import java.applet.\*;

public class calcdemo extends Applet implements ActionListener

{ static int flag=0; static int result=0,x=0,y=0; char ch;

Button b[]=new Button[10];

Button d=new Button(".");

Button eq=new Button("=");

Button op1=new Button("\*");

Button op2=new Button("+");

Button op3=new Button("-");

Button op4=new Button("/");

Panel p1=new Panel();

Panel p2=new Panel();

Panel p3=new Panel();

TextField t1=new TextField(25);

public void init()

{ for(int i=0;i<10;i++)

{

b[i]=new Button(""+i);

} this.setLayout(new GridLayout(2,1)); p1.setLayout(new GridLayout(1,2)); p2.setLayout(new GridLayout(4,1)); p3.setLayout(new GridLayout(4,3)); for(int i=0;i<10;i++)

{ b[i].addActionListener(this);

p3.add(b[i]);

} p3.add(d); p3.add(eq);

d.addActionListener(this); eq.addActionListener(this); op1.addActionListener(this); op2.addActionListener(this); op3.addActionListener(this); op4.addActionListener(this); p2.add(op1); p2.add(op2); p2.add(op3); p2.add(op4); p1.add(p3); p1.add(p2); this.add(t1);

this.add(p1);

}

public void actionPerformed(ActionEvent ae)

{

String s1="",s2=""; flag=0; for(int i=0;i<10;i++)

{ if(ae.getSource()==b[i])

{ if(flag==0)

{ t1.setText(b[i].getLabel()); s1=t1.getText(); x=Integer.parseInt(s1.trim()); t1.setText(" ");

flag++; } else

{ t1.setText(b[i].getLabel()); s2=t1.getText();

y=Integer.parseInt(s2.trim());

}

break;

}

}

if(ae.getSource()==op1) ch='\*'; if(ae.getSource()==op2) ch='+'; if(ae.getSource()==op3) ch='-'; if(ae.getSource()==op4) ch='/'; if(ae.getSource()==d) t1.setText(""); if(ae.getSource()==eq)

{ switch(ch)

{

case '+': result=x+y; t1.setText(Integer.toString(result));

break;

case '\*': result=x\*y; t1.setText(Integer.toString(result));

break;

case '-': result=x-y; t1.setText(Integer.toString(result));

break;

case '/': result=x/y; t1.setText(Integer.toString(result));

break;

}

}

}

}

/\*<applet code=calcdemo.class width=800 height=800>

</applet>\*/

Slip - 13

Q1) Write a program to accept a file name from command prompt, if the file exits then display number of words and lines in that file.

→

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

public class fileinfo { public static void main(String[] args) throws Exception

{

File f = new File(args[0]); Scanner sc=new Scanner(f);

int wordCount = 0;

int lineCount = 0;

try

{

if(f.exists())

{

while(sc.hasNext())

{

wordCount++;

sc.next();

}

System.out.println("Total word count = "+ wordCount); Scanner sc1=new Scanner(f); while(sc1.hasNextLine())

{

sc1.nextLine();

lineCount++;

}

System.out.println("Total number of lines = "+ lineCount);

sc.close();

}

}catch(Exception e)

{System.out.println("File not Found" );}

}

}

Q2) Write a program to display the system date and time in various formats shown below: Current date is : 31/08/2021

Current date is : 08-31-2021

Current date is : Tuesday August 31 2021

Current date and time is : Fri August 31 15:25:59 IST 2021

Current date and time is : 31/08/21 15:25:59 PM +0530

→

*import java.util.\*; import java.text.SimpleDateFormat; public class datepattern*

*{*

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

*{*

*String x,pattern;*

*SimpleDateFormat sd;*

*pattern="dd-MM-yy"; sd=new SimpleDateFormat(pattern);*

*Date d=new Date(); x=sd.format(d); System.out.println(x);*

*pattern="MM/dd/yyyy"; sd=new SimpleDateFormat(pattern);*

*String x1=sd.format(d); System.out.println(x1);*

*pattern="EEE,d MMM yyyy HH:mm:ss Z"; sd=new SimpleDateFormat(pattern);*

*String x2=sd.format(d); System.out.println(x2);*

*pattern="HH:mm:ss"; sd=new SimpleDateFormat(pattern); x=sd.format(d); System.out.println(x);*

*pattern="EEEE,MMMM,dd,yyyy";*

*sd=new SimpleDateFormat(pattern); x=sd.format(d);*

*System.out.println(x);*

*}*

*}*

*Slip no 14*

Q1) Write a program to accept a number from the user, if number is zero then throw user defined exception “Number is 0” otherwise check whether no is prime or not (Use static keyword).

→

import java.io.\*; class NumberZeroException extends Exception

{

NumberZeroException()

{

System.out.println("Number is 0");

}

}

class PrimeNumber

{ int a;

BufferedReader br=new BufferedReader(new InputStreamReader(System.in));

PrimeNumber()

{

try

{

System.out.println("Enter any integer to check prime "); a=Integer.parseInt(br.readLine()); if(a==0)

throw new NumberZeroException();

}

catch(NumberZeroException ex)

{

System.out.println(ex);

}

catch(IOException ex1)

{

System.out.println("Enter proper number");

} }

public void prime()

{ int cnt=0; for(int i=2;i<=a/2;i++)

if(a%i==0)

{

cnt++; break;

} if(cnt==0)

System.out.println(a+" Number is prime");

else

System.out.println(a+" Number is not prime");

} public static void main(String args[])

{

PrimeNumber pn=new PrimeNumber();

pn.prime();

}

}

Q2) Write a Java program to create a Package “SY” which has a class SYMarks (members – ComputerTotal, MathsTotal, and ElectronicsTotal). Create another package TY which has a class TYMarks (members – Theory, Practicals). Create ‘n’ objects of Student class (having rollNumber, name, SYMarks and TYMarks). Add the marks of SY and TY computer subjects and calculate the Grade (‘A’ for >= 70, ‘B’ for >= 60 ‘C’ for >= 50, Pass Class for > =40 else‘FAIL’) and display the result of the student in proper format.

→

package sy; public class symarks

{ int ctotal; int mtotal; int etotal;

public symarks()

{ }

public symarks(int c,int m, int e)

{ ctotal=c;

mtotal=m;

etotal=e;

} public void display() {

System.out.println("Marks of comp sci:"+ctotal);

System.out.println("Marks of maths:"+mtotal);

System.out.println("Marks of electronics:"+etotal);

} public int sytotal()

{

return(ctotal+mtotal+etotal);

}

}

-------------------------------------------------------

package ty; public class tymarks

{ int theory; int practical;

public tymarks(){}

public tymarks(int t, int p)

{ theory=t;

practical=p;

} public int tytotal()

{

return(theory+practical);

} public void display()

{

System.out.println("Marks of Theory:"+theory);

System.out.println("Marks of Practical:"+practical);

}

}

-------------------------------------------------------------------------

import sy.\*; import ty.\*; import java.util.\*; class student

{ int roll;

String name; symarks o1; tymarks o2; student(){} student(int roll, String name)

{ this.roll=roll;

this.name=name;

Scanner sc=new Scanner(System.in); System.out.println("Enter ctotal:"); int c=sc.nextInt();

System.out.println("Enter mtotal:");

int m=sc.nextInt();

System.out.println("Enter etotal:");

int e=sc.nextInt();

o1=new symarks(c,m,e);

System.out.println("Enter Theory marks:");

int th=sc.nextInt();

System.out.println("Enter practical:"); int pc=sc.nextInt();

o2=new tymarks(th,pc);

} void grade()

{ int gt=o1.sytotal()+o2.tytotal(); System.out.println("Total Marks is:"+gt); double per=gt/5; if(per>=75)

System.out.println("Distinction with Percentage:"+per); else if(per>=60&&per<75)

System.out.println("First class with percentage:"+per); else if(per>=50&&per<60)

System.out.println("Second class with percentage:"+per);

else

System.out.println("You are Fail with percentage"+per);

} public static void main(String g[])

{ student s[]=new student[2]; Scanner sc1=new Scanner(System.in);

for(int i=0;i<2;i++)

{

System.out.println("Enter roll no.:"); int r=sc1.nextInt();

System.out.println("Enter Name:");

String nm=sc1.next(); s[i]=new student(r,nm);

s[i].o1.display(); s[i].o2.display();

s[i].grade();

}

}

}

Slip - 15

Q1) Accept the names of two files and copy the contents of the first to the second. First file having Book name and Author name in file.

→

import java.io.\*;

class setac {

public static void main(String[] args)

{

try {

FileReader fr = new FileReader("sample.txt");

FileWriter fw = new FileWriter("sampleout.txt"); String str = "";

int i;

while ((i = fr.read()) != -1)

{

str += (char)i;

}

System.out.println(str);

fw.write(str);

fr.close(); fw.close();

System.out.println(

"File reading and writing both done");

}

catch (IOException e) {

System.out.println("There are some IOException");

}

}

}

Q2) Write a program to define a class Account having members custname, accno. Define default and parameterized constructor. Create a subclass called

Savingwithdraw Account with member savingbal, minbal. Create a derived class AccountDetail that extends the class SavingAccount with members, depositamt and withdrawalamt. Write a appropriate method to display customer details.

→import java.util.\*; class account

{

String custnm; String accno; account()

{

custnm="Roy";

accno="ABOP123";

} account(String custnm,String accno)

{

this.custnm=custnm;

this.accno=accno;

}

} class savingacc extends account

{

long savingbal; int minbal=500; savingacc(String cnm,String ano,long savingbal)

{

super(cnm,ano);

this.savingbal=savingbal;

}

} class accdetails extends savingacc

{

double depositamt; double withdrawamt;

accdetails(String cnm,String ano,long savingbal)

{

super(cnm,ano,savingbal); this.depositamt=depositamt;

this.withdrawamt=withdrawamt;

}

public void deposit() { long amt;

Scanner sc = new Scanner(System.in);

System.out.println("Enter the amount you want to deposit: "); amt = sc.nextLong();

savingbal = savingbal + amt;

System.out.println("Balance after deposit: " + savingbal);

}

public void withdrawal() { long amt;

Scanner sc = new Scanner(System.in);

System.out.println("Enter the amount you want to withdraw: "); amt = sc.nextLong(); if (savingbal >= amt) { savingbal = savingbal - amt;

System.out.println("Balance after withdrawal: " + savingbal);

} else {

System.out.println("Your balance is less than " + amt + "\tTransaction failed...!!" );

}

} void display()

{

System.out.println("Customer Name:-\t"+custnm);

System.out.println("Customer Account No.:-\t"+accno);

System.out.println("Saving Balance:-\t"+savingbal);

System.out.println("Minimum Balance Should be:-\t"+minbal);

}

public static void main(String arg[])

{

Scanner sc = new Scanner(System.in);

System.out.print("Enter Account No: ");

String acno = sc.next();

System.out.print("Enter Customer Name: ");

String cusnm = sc.next();

System.out.print("Enter Balance: "); long balance = sc.nextLong();

accdetails obj=new accdetails(cusnm,acno,balance); obj.deposit();

obj.withdrawal();

}

}

Slip - 16

Q1) Write a program to find the Square of given number using function interface.

→

interface PrintNumber

{ public void print(int num1);

} public class PrintSquare

{ public static void main(String[] a)

{

PrintNumber p = n -> System.out.println("Square is: "+ n\*n); p.print(5);

}

}

Q2) Write a program to design a screen using Awt that,



→import java.awt.\*; import javax.swing.\*;

class MenuExample

{

MenuExample(){

JFrame f= new JFrame("AWT Example");

JMenuBar mb=new JMenuBar();

JMenu menu=new JMenu("File");

JMenu menu1=new JMenu("Edit");

JMenu menu2=new JMenu("About");

JMenuItem i1=new JMenuItem("NEW \t\t Ctrl+N");

JMenuItem i2=new JMenuItem("Open");

JMenuItem i3=new JMenuItem("Save");

JCheckBoxMenuItem cb=new JCheckBoxMenuItem ("Show About");

JMenuItem i5=new JMenuItem("Exit"); mb.add(menu); mb.add(menu1); mb.add(menu2); menu.add(i1); menu.add(i2); menu.add(i3); menu.addSeparator();

menu.add(cb); menu.addSeparator(); menu.add(i5);

f.setJMenuBar(mb);

f.setSize(400,400);

f.setLayout(null);

f.setVisible(true);

f.setDefaultCloseOperation(JFrame.EXIT\_ON\_CLOSE);

} public static void main(String args[])

{

new MenuExample();

}

}

Slip - 17

Q1) Design a Super class Customer (name, phone-number). Derive a class

Depositor(accno , balance) from Customer. Again, derive a class Borrower (loan-no, loan-amt) from Depositor. Write necessary member functions to read and display the details of ‘n’customers.

→

import java.util.\*;

class customer

{

String nm; long phno;

public void acceptc()

{

Scanner sc=new Scanner(System.in); System.out.println(" Enter Customer Name:"); nm=sc.next();

System.out.println(" Enter Customer Phone No. : ");

phno=sc.nextLong();

} void dispc()

{

System.out.println(" Details of Customer \n");

System.out.println("\n ----------------------------------------\n"); System.out.println("\n Customer Name : "+nm);

System.out.println("\n Customer Phone No. : "+phno);

} } class deposit extends customer

{ int accno, balance; public void acceptd()

{

Scanner sc=new Scanner(System.in);

System.out.println("\n Enter Customer A/c No : "); accno=sc.nextInt();

System.out.println("\n Enter Balance:");

balance=sc.nextInt();

} void dispd()

{

System.out.println("\n Customer A/c No : "+accno);

System.out.println("\n Balance : "+balance);

System.out.println("\n ----------------------------------------\n");

} } class borrow extends deposit

{ int loan\_no,loan\_amt; public void acceptb()

{

Scanner sc=new Scanner(System.in);

System.out.println("\n Enter Loan No : "); loan\_no=sc.nextInt();

System.out.println("\n Enter Loan Amount : "); loan\_amt=sc.nextInt();

System.out.println("\n ----------------------------------------\n");

} void dispb()

{

System.out.println("\n Loan No : "+loan\_no);

System.out.println("\n Loan Amount : "+loan\_amt);

System.out.println("\n ----------------------------------------\n");

}

public static void main(String a[])

{

Scanner sc=new Scanner(System.in);

System.out.println("\n Enter No. of Customer Details You Want : "); int n=sc.nextInt();

borrow b1[]=new borrow[n];

for(int i=0; i<b1.length; i++)

{

b1[i]= new borrow();

b1[i].acceptc();

b1[i].acceptd();

b1[i].acceptb();

} for(int i=0; i<b1.length; i++)

{

System.out.println("Inside for looop Array "); b1[i].dispc(); b1[i].dispd();

b1[i].dispb();

}

}

}

Q2) Write Java program to design three text boxes and two buttons using swing. Enter different strings in first and second textbox. On clicking the First command button, concatenation of two strings should be displayed in third text box and on clicking second command button, reverse of string should display in third text box

→

import java.awt.\*; import java.awt.event.\*; import javax.swing.\*; class stringGui extends JFrame implements ActionListener

{

JTextField t1,t2,t3;

JLabel l1,l2,l3; JButton b1,b2; stringGui()

{

l1=new JLabel("String 1:"); l2=new JLabel("String 2:"); l3=new JLabel("String 3:"); t1=new JTextField(20); t2=new JTextField(20); t3=new JTextField(20); b1=new JButton("Concat"); b2=new JButton("Reverse"); setLayout(new FlowLayout()); add(l1);add(t1); add(l2);add(t2); add(b1);add(b2); add(l3);add(t3); setVisible(true); setSize(600,600); setTitle("String Demo");

setDefaultCloseOperation(JFrame.EXIT\_ON\_CLOSE);

b1.addActionListener(this);

b2.addActionListener(this);

}

public void actionPerformed(ActionEvent ae)

{

if(ae.getSource()==b1)

{

String s1 = t1.getText();

String s2 = t2.getText(); String s3=s1+s2 ;

t3.setText(s3);

} if(ae.getSource()==b2)

{

String r="";

String s4=t1.getText()+t2.getText(); for (int i=0; i<s4.length(); i++)

{ char ch= s4.charAt(i);

r= ch+r;

}

t3.setText(r);

}

} public static void main(String a[])

{ new stringGui();

}

}

Slip - 18

Q1) Write a program to implement Border Layout Manager.

→

import java.awt.\*; import javax.swing.\*;

public class Border

{

JFrame f;

Border()

{

f = new JFrame();

JButton b1 = new JButton("MMCC");

JButton b2 = new JButton("Announcement");

JButton b3 = new JButton("BCA");

JButton b4 = new JButton("BSc"); JButton b5 = new JButton("SPPU");

f.add(b1, BorderLayout.NORTH);

f.add(b2, BorderLayout.SOUTH);

f.add(b3, BorderLayout.EAST);

f.add(b4, BorderLayout.WEST);

f.add(b5, BorderLayout.CENTER);

f.setSize(300, 300);

f.setVisible(true); } public static void main(String[] args) { new Border();

}

}

Q2) Define a class CricketPlayer (name,no\_of\_innings,no\_of\_times\_notout, totatruns, bat\_avg). Create an array of n player objects. Calculate the batting average for each player using static method avg(). Define a static sort method which sorts the array on the basis of average. Display the player details in sorted order.

→

import java.io.\*;

public class CricketPlayer

{

String name; int runs,no\_of\_innigs,no\_of\_times\_notout; float bat\_avg; public CricketPlayer()

{

} public static CricketPlayer[] sortPlayer(CricketPlayer [] player)

{

CricketPlayer temp;

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

{ for(int j=i+1;j<player.length;j++)

{ if(player[i].bat\_avg<player[i].bat\_avg)

{

temp=player[i]; player[i]=player[j];

player[j]=temp;

}

}

}

return player;

} public static CricketPlayer[] avg(CricketPlayer [] players)

{

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

{

players[i].bat\_avg=players[i].runs/players[i].no\_of\_innigs;

}

return players;

}

public void accept() throws Exception

{

BufferedReader br=new BufferedReader(new InputStreamReader(System.in)); System.out.println("Enter player Name:"); name=br.readLine();

System.out.println("Enter Number of Innings:\t"); no\_of\_innigs=Integer.parseInt(br.readLine());

if(no\_of\_innigs<=0)

System.out.println("Innigs cant be less than zero");

System.out.println("Enter Total Runs:\t"); runs=Integer.parseInt(br.readLine());

if(runs<0)

System.out.println("Runs cant be less than negative");

System.out.println("Enter Number of time notout:\t"); no\_of\_times\_notout=Integer.parseInt(br.readLine());

if(no\_of\_times\_notout>no\_of\_innigs)

System.out.println("Number of notout cant not be greater

than no of innigs");

}

public static void main(String[] args) throws Exception

{ BufferedReader br=new BufferedReader(new

InputStreamReader(System.in));

CricketPlayer players[];

System.out.println("How many players you want?:\t");

int n=Integer.parseInt(br.readLine()); if(n<0)

System.out.println("N cant not be negative");

else {

players=new CricketPlayer[n]; for(int i=0;i<n;i++)

{ players[i]=new CricketPlayer();

players[i].accept();

}

CricketPlayer.avg(players);

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

System.out.println("Name\t\tInnings\t\tRuns\t\tNotout\t\tBat Avg");

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

{

System.out.print(players[i].name+"\t\t"+players[i].no\_of\_innigs+"\t\t");

System.out.print(players[i].runs+"\t\t"+players[i].no\_of\_times\_notout+"\t\t");

System.out.println(players[i].bat\_avg);

}

CricketPlayer.sortPlayer(players);

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

System.out.println("Name\t\tInnings\t\tRuns\t\tNotout\t\tBat Avg");

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

{

System.out.print(players[i].name+"\t\t"+players[i].no\_of\_innigs+"\t\t");

System.out.print(players[i].runs+"\t\t"+players[i].no\_of\_times\_notout+"\t\t");

System.out.println(players[i].bat\_avg);

}

}

}

}

Slip - 19

Q1) Write a program to accept the two dimensional array from user and display sum of its diagonal elements. →import java.util.\*; import java.io.\*;

public class matrixeg{

static void Sum\_of\_Diagonals1(int[][] matrix,int N)

{

int Pd = 0, Sd = 0;

for (int k = 0; k < N; k++) {

for (int l = 0; l < N; l++) {

if (k == l)

Pd += matrix[k][l];

if ((k + l) == (N - 1))

Sd += matrix[k][l];

}

}

System.out.println("Sum of LEFT Diagonal:"+ Pd);

System.out.println("Sum of RIGHT Diagonal:"+ Sd);

} public static void main(String[] args)

{

Scanner s = new Scanner(System.in);

int a[][] = new int[4][4];

System.out.println("Enter all the elements of first matrix:");

for (int i = 0; i < 4; i++)

{ for (int j = 0; j < 4; j++)

{

a[i][j] = s.nextInt();

}

}

Sum\_of\_Diagonals1(a,4);

}

}

Q2) Write a program which shows the combo box which includes list of T.Y.B.Sc.(Comp. Sci) subjects. Display the selected subject in a text field.

→

import javax.swing.\*; import java.awt.\*; import java.awt.event.\*; public class comboboxeg extends JFrame implements ItemListener

{

JTextField t1;

JComboBox cb;

JLabel l1;

comboboxeg()

{

t1=new JTextField(20); l1=new JLabel("TYBSc(CS) Subjects");

String languages[]={"OS","DS","PHP","Java","Phython"}; cb=new JComboBox(languages);

cb.setBounds(50, 100,90,20); setLayout(new FlowLayout()); add(l1) ; add(cb); add(t1) ;

cb.addItemListener(this);

setSize(350,350);

setVisible(true);

setTitle("Combo Box");

setDefaultCloseOperation(JFrame.EXIT\_ON\_CLOSE);

}

public void itemStateChanged(ItemEvent e)

{

if (e.getStateChange() == ItemEvent.SELECTED)

{

String no=cb.getSelectedItem().toString();

t1.setText(no);

}

}

public static void main(String[] args)

{

new comboboxeg();

}

}

Slip – 20

Q1) Write a Program to illustrate multilevel Inheritance such that country is inherited from continent. State is inherited from country. Display the place, state, country and continent.

→

import java.util.\*; class continent {

String con;

Scanner sc=new Scanner(System.in); void cinput()

{

System.out.println("Enter Continent Name: "); con =sc.next();

} } class country extends continent

{

String cou; void couinput()

{

System.out.println("Enter Country Name: ");

cou = sc.next();

} } class state extends country

{

String sta; void stainput() {

System.out.println("Enter State Name: ");

sta = sc.next();

} } class place extends state

{ String pla; void pinput()

{

System.out.println("Enter Place Name : ");

pla = sc.next(); }

public static void main( String arg[] )

{ place s = new place();

s.cinput();

s.couinput();

s.stainput();

s.pinput();

System.out.println("Place :" + s.pla);

System.out.println("State: "+s.sta);

System.out.println("Country: "+s.cou);

System.out.println("Continent: "+s.con);

}

}

Q2) Write a package for Operation, which has two classes, Addition and

Maximum. Addition has two methods add () and subtract (), which are used to add two integers and subtract two, float values respectively. Maximum has a method max () to display the maximum of two integers

→

Slip - 21

Q1) Define a class MyDate(Day, Month, year) with methods to accept and display a MyDateobject. Accept date as dd,mm,yyyy. Throw user defined exception "InvalidDateException" if the date is invalid.

→

import java.util.\*;

class InvalidDateException extends Exception { public InvalidDateException(String g)

{ super(g);

} }

class mydateprg{ int flag=0; int dd,mm,yy; mydateprg(){ dd=0; mm=0; yy=0;

} void accept()

{ int d,m,y; Scanner sc=new Scanner(System.in); System.out.println("\n enter date in date,month,year format"); d=sc.nextInt(); m=sc.nextInt(); y=sc.nextInt(); try{ if(m>12)

throw new InvalidDateException("month can't be greater

than 12"); else

{ if(m==1 || m==3 || m==5 || m==7 || m==8 || m==10

||m==12)

{ if(d>31) throw new

InvalidDateException("date can't be greater than 31");

} else if(m==2) { if(y%4==0) { if(d>29) throw new InvalidDateException("in leap

year date can't be greater than 29");

} else { if(d>28) throw new

InvalidDateException("in february month date can't be greater than 28");

} } else { if(d>30) throw new InvalidDateException("date

can't be greater than 30 of "+m+"month");

}

}

if(flag==0){ dd=d; mm=m; yy=y;

}

}catch(InvalidDateException e){ flag=1; System.out.println(e);

}

} void display(){ if(flag==0)

System.out.println("date is :"+dd+"/"+mm+"/"+yy); else

System.out.println("date is invalid");

}

public static void main(String args[]){ mydateprg md=new mydateprg(); md.accept();

md.display(); }

}

Q2) Create an employee class(id,name,deptname,salary). Define a default and parameterized constructor. Use ‘this’ keyword to initialize instance variables. Keep a count of objects created. Create objects using parameterized constructor and display the object count after each object is created. (Use static member and method). Also display the contents of each object.

→

*import java.util.\*; class employee*

*{ int id;*

*String name,deptnm; float salary;*

*static int c; employee()*

*{*

*id=101; name="Virat"; deptnm="HR"; salary=80000;*

*c++;*

*} employee(int id, String name, String deptnm,float salary)*

*{ this.id=id;*

*this.name=name; this.deptnm=deptnm; this.salary=salary;*

*c++;*

*} public void display()*

*{*

*System.out.println("Employuee id is:"+id);*

*System.out.println("Employee name is:"+name);*

*System.out.println("Employee department is:"+deptnm);*

*System.out.println("Employee salary is:"+salary);*

*} static void displaycnt()*

*{*

*System.out.println("Object count is:"+c);*

*} public static void main(String g[])*

*{*

*Scanner sc= new Scanner(System.in); employee e1= new employee();*

*System.out.println("Enter employee id:"); int i=sc.nextInt();*

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

*String s=sc.next();*

*System.out.println("Enter deptname:");*

*String d=sc.next();*

*System.out.println("Enter employee salary:"); float sal=sc.nextFloat();*

*employee e2= new employee(i,s,d,sal); e1.display();*

*System.out.println(" "); e2.display(); employee.displaycnt();*

*}*

*}*

*Slip - 22*

Q1) Write a program to create an abstract class named Shape that contains two integers and an empty method named printArea(). Provide three classes named Rectangle, Triangle and Circle such that each one of the classes extends the class Shape. Each one of the classes contain only the method printArea() that prints the area of the given shape. (use method overriding).

→

import java.util.\*; abstract class shape

{

int x,y;

abstract void area(double x,double y);

} class rectangle extends shape

{ void area(double x,double y)

{

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

} } class circle extends shape

{ void area(double x,double y)

{

System.out.println("area of circle is :"+(3.14\*x\*x));

} } class triangle extends shape

{ void area(double x,double y)

{

System.out.println("area of triangle is :"+(0.5\*x\*y));

} }

public class shapedemo

{ public static void main(String[] args)

{ rectangle r=new rectangle();

r.area(2,5);

circle c=new circle();

c.area(5,5); triangle t=new triangle();

t.area(2,5);

}

}

Q2) Write a program that handles all mouse events and shows the event name at the center of the Window, red in color when a mouse event is fired. (Use adapter classes).

→

import java.awt.\*; import java.util.\*; import java.applet.\*; import javax.swing.\*; import java.awt.event.\*;

public class appleteventdemo extends Applet implements MouseListener,MouseMotionListener,KeyListener

{

String x="";

TextField t1=new TextField(25); public void init()

{ t1.addKeyListener(this); this.add(t1); this.addMouseListener(this);

this.addMouseMotionListener(this);

} public void paint(Graphics g) {

g.drawString(x,150,100); }

public void keyTyped(KeyEvent ke)

{

x="Key Typed";

t1.setText(""+ke.getKeyChar());

repaint(); }

public void keyPressed(KeyEvent ke)

{

x="Key pressed";

repaint(); }

public void keyReleased(KeyEvent ke)

{

x="Key Released";

repaint();

}

public void mouseMoved(MouseEvent me)

{

x="Mouse Moved";

repaint();

}

public void mouseDragged(MouseEvent me)

{

x="Mouse Gragged";

repaint();

}

public void mouseEntered(MouseEvent me)

{

x="Mouse Entered";

repaint();

}

public void mouseExited(MouseEvent me)

{

x="Mouse Exited";

repaint();

}

public void mousePressed(MouseEvent me)

{

x="Mouse Pressed";

repaint();

}

public void mouseClicked(MouseEvent me)

{

x="Mouse Clicked";

repaint();

}

public void mouseReleased(MouseEvent me)

{

x="Mouse Released";

repaint();

}

}

/\*<applet code=appleteventdemo.class width=600 height=600>

</applet>\*/

Slip - 23

Q1) Define a class MyNumber having one private int data member. Write a default constructor to initialize it to 0 and another constructor to initialize it to a value (Use this). Write methods isNegative, isPositive, isZero, isOdd, isEven. Create an object in main.Use command line arguments to pass a value to the Object.

→

*class mynum*

*{ private int a;*

*mynum()*

*{*

*a=0;*

*}*

*mynum(int a)*

*{ this.a=a;*

*} public void iseven()*

*{*

*if(a%2==0)*

*System.out.println("Number is even:"+a);*

*}*

*public void ispositive()*

*{ if(a>0)*

*System.out.println("Number is positive:"+a);*

*} public void isnegative()*

*{ if(a<0)*

*System.out.println("Number is negative:"+a);*

*} public void isodd()*

*{ if(a%2!=0)*

*System.out.println("Number is odd:"+a);*

*}*

*public void iszero()*

*{ if(a==0)*

*System.out.println("Number is zero:"+a);*

*} public static void main(String g[])*

*{ int x=Integer.parseInt(g[0]);*

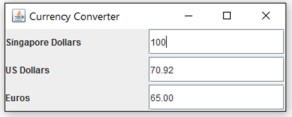
*mynum obj=new mynum(x);*

*obj.ispositive(); obj.isnegative(); obj.iseven(); obj.isodd(); obj.iszero();*

*}*

*}*

Q2) Write a simple currency converter, as shown in the figure. User can enter the amount of "Singapore Dollars", "US Dollars", or "Euros", in floating-point number. The converted values shall be displayed to 2 decimal places. Assume that 1 USD = 1.41 SGD, 1 USD = 0.92 Euro, 1 SGD = 0.65 Euro.



→

import javax.swing.\*; import java.awt.\*; import java.awt.event.\*; public class converter { converter()

{

JFrame f = new JFrame("CONVERTER");

JLabel l1, l2,l3;

JTextField t1, t2,t3; JButton b1;

l1 = new JLabel("Singapore Dollars:"); l1.setBounds(20, 40, 60, 30); l2 = new JLabel("US Dollars:"); l2.setBounds(170, 40, 60, 30); l3 = new JLabel("Euros:"); l3.setBounds(170, 40, 60, 30);

t1 = new JTextField(20); t1.setBounds(80, 40, 50, 30); t2 = new JTextField(20); t2.setBounds(240, 40, 50, 30); t3 = new JTextField(20); t3.setBounds(240, 40, 50, 30);

b1 = new JButton("Convert"); b1.setBounds(50, 80, 60, 15); b1.addActionListener(new ActionListener() { public void actionPerformed(ActionEvent e)

{

double d= Double.parseDouble(t1.getText());

double d1 = (d \*0.740); String str1 = String.valueOf(d1);

t2.setText(str1);

double d2=(d\*0.697);

String str2 = String.valueOf(d2);

t3.setText(str2);

}

});

f.add(l1);

f.add(t1);

f.add(l2);

f.add(t2);

f.add(l3);

f.add(t3);

f.add(b1);

f.setDefaultCloseOperation(JFrame.EXIT\_ON\_CLOSE);

f.setLayout(new GridLayout(4,2));

f.setSize(400, 300);

f.setVisible(true); }

public static void main(String args[])

{

new converter();

}

}