Java Assignment 4

Q1. Write a program in java to check whether the entered character is alphabet, digit or space character. If it is an alphabet then print whether it is capital or a small alphabet. Also change the alphabet into the reverse case.

import java.io.\*;

class Check

{

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

{

charch;

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

System.out.println("Enter Any Character:-");

ch=(char) inputstream.read();

if(Character.isDigit(ch))

System.out.println("Entered Character is Digit");

else if(Character.isUpperCase(ch))

{

System.out.println("Entered Character is Upper case");

System.out.println("Lower Case Character is:-"+Character.toLowerCase(ch));

}

else if(Character.isLowerCase(ch))

{

System.out.println("Entered Character is Lower Case");

System.out.println("Upper Case Character is:-"+Character.toUpperCase(ch));

}

else

System.out.println("Entered Character is special char");

}

}

Output:

Enter Any Character:-

J

Entered Character is Upper case

Lower Case Character is:-j

Q2. Write a program in java to check whether the entered character is alphabet, digit or space character. If it is an alphabet then print whether it is capital or a small alphabet. Also change the alphabet into the reverse case.

import java.io.\*;

classArrayDemo

{

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

{

intj,s,i,k=0,m=0,n;

int r[]=new int[10];

int prime[]=new int[10];

int perfect[]=new int[10];

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

System.out.println("Enter the Number of Elements:=>");

n=Integer.parseInt(br.readLine());

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

{

System.out.println("Enter the "+i+" Number of the Array:=>");

r[i]=Integer.parseInt(br.readLine());

}

for(j=0;j<10;j++)

{

int flag=0;

if(r[j]==0||r[j]==1)

{

continue;

}

else

{

for(i=2;i<r[j];i++)

{

if(r[j]%i==0)

flag=1;

}

}

if(flag==0)

{

prime[k]=r[j];

k++;

}

s=0;

for(i=1;i<r[j];i++)

{

n=r[j]%i;

if(n==0)

s=s+i;

}

if(s==r[j])

{

perfect[m]=r[j];

m++;

}

}

System.out.println("\nPerfect Numbers are");

for(i=0;i<m;i++)

{

System.out.println(perfect[i]+" ");

}

System.out.println("\nPrime Numbers are");

for(i=0;i<k;i++)

{

System.out.print(prime[i]+" ");

}

}

}

Output:

Enter the Number of Elements:=>

4

Enter the 0 Number of the Array:=>

2

Enter the 1 Number of the Array:=>

3

Enter the 2 Number of the Array:=>

6

Enter the 3 Number of the Array:=>

7

Perfect Numbers are

6

Prime Numbers are

2 3 7

Q3. Write a program in Java to accept name of cities from the user and sort them in ascending order (Use Command Line Argument

import java.io.\*;

classCitySort

{

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

{

int n,len,len1,c,i,j,k=0,m=0;

String str[]=new String[5];

String s1;

String s2,temp;

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

for(i=0;i<5;i++)

{

System.out.println("Enter the string "+i+" of the array:");

str[i]=br.readLine();

}

System.out.println("\nSorted Strings are:");

for(i=0;i<5;i++)

{

for(j=i+1;j<5;j++)

{

if(str[j].compareTo(str[i])<0)

{

temp=str[i];

str[i]=str[j];

str[j]=temp;

}

}

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

}

}

}

Output:

Enter the string 0 of the array:

2

Enter the string 1 of the array:

3

Enter the string 2 of the array:

6

Enter the string 3 of the array:

9

Enter the string 4 of the array:

1

Sorted Strings are:

1

2

3

6

9

Q4. Create an Interface Shape Derive three classes sphere, cone and cylinder from it. Calculate area and volume of all.

import java.util.\*;

abstract class Shape

{

double area, vol; void display()

{

System.out.println("Volume:\n"+vol);

System.out.println("Area :\n"+area);

}

abstract void calVol();

abstract void calArea();

}

class Cone extends Shape

{

float r,h;

Cone(float r,float h)

{

this.r=r; this.h=h;

}

void calVol()

{ vol=4/3\*3.14\*r\*r\*r;

}

void calArea()

{

area=4\*3.14\*r\*r\*h;

}

}

class Sphere extends Shape

{

float r;

Sphere(float r)

{

this.r=r;

}

void calVol()

{

vol=4/3\*3.14\*r\*r\*r;

}

void calArea()

{

area=4\*3.14\*r\*r\*h;

}

}

class Cylinder extends Shape

{

float r,h;

Cylinder(float r,float h)

{

this.r=r; this.h=h;

}

void calVol()

{

vol=3.14\*r\*r\*h;

}

void calArea() { area=(2\*3.14\*r\*h)+(2\*3.14\*r\*r);

}

}

class ShapeDemo

{

public static void main(String args[])

{

Scanner scan=new Scanner(System.in);

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

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

float r=scan.nextFloat();

float h=scan.nextFLoat();

Cone c=new Cone(r,h);

c.calArea();

c.calVol() c.display();

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

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

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

float r1=scan.nextFloat();

Sphere s=new Sphere(r1);

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

s.calArea();

s.calVol();

s.display();

}

}

Q5. Create an interface StringDemo which has a methods check (). Implement interface in StringCheck and implement the method of interface StringDemo. check () will check whether two string are equal or not

import java.util.\*;

interface StringDemo void Check();

{

}

class StringCheck implements StringDemo

{

String stri, str2;

StringCheck(String stri, String str2)

{

this.stri = stri;

this.str2 = str2;

public void Check()

{

if (str1.equals (str2))

System.out.print("Strings are equal");

else

System.out.print("Strings are not equal");

}

}

public class Main public static void main(String[] args)

{

Scanner scanner = new Scanner(System.in);

String str1, str2;

System.out.println("Enter first String");

str1 = scanner.nextLine();

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

str2 = scanner.nextLine();

StringCheck s1 = new String Check (str1, str2);

s1.Check();

}

Q6. Write a java program that reads lines of integers. Display each integer and also display sum of all integers

import java.io.\*;

classSumInt

{

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

{

long sum=0;

String i="";

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

System.out.println("Enter Line of Integer");

longstr=Long.parseLong(br.readLine());

while(str>0)

{

i=i+""+str%10;

sum=sum+str%10;

str=str/10;

}

System.out.println("All integer Digits :->" );

for(int k=i.length()-1;k>=0;k--)

{

System.out.println(i.charAt(k));

}

System.out.println("Summation of all integer Digits :-> "+sum);

}

}

Output:

Enter Line of Integer

12345

All integer Digits :->

1

2

3

4

5

Summation of all integer Digits :-> 15

Q7. Write a package for Games, which have two classes Indoor and Outdoor. Use a function display () to generate the list of players for the specific games. Use parameterized constructors

import game.Indoor;

import game.Outdoor;

import java.io.\*;

class Slip20

{

protected void finalize()

{

System.out.println("Finalized method is invoked");

}

public static void main(String a[]) throws IOException

{

String nm,gm;

int i;

BufferedReader br = new BufferedReader(new

InputStreamReader(System.in));

System.out.println("enter no indoor player");

int n= Integer.parseInt(br.readLine());

Indoor in[]=new Indoor[n];

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

{

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

nm=br.readLine();

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

gm=br.readLine();

in[i]=new Indoor(nm,gm);

}

System.out.println("enter no outdoor player");

int n1= Integer.parseInt(br.readLine());

Outdoor o[]=new Outdoor[n1];

for(i=0;i<n1;i++)

{

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

nm=br.readLine();

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

gm=br.readLine();

o[i]=new Outdoor(nm,gm);

}

System.out.println("enter game name for indoor : ");

gm=br.readLine();

System.out.println("Player name of specified indoor game : ");

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

{

in[i].display(gm);

}

System.out.println("enter game name for outdoor : ");

gm=br.readLine();

System.out.println("Player name of specified outdoor game: ");

for(i=0;i<n1;i++)

{

o[i].display(gm);

}

Slip20 ob = new Slip20();

ob.finalize();

}

}

Q8) Create a package MCA which will have 2 classes as class Mathematics with a methods to add two numbers, add three float numbers and class Maximum with a method to find maximum of three numbers

import java.io.\*;

class Mathematics

{

floata,b,c;

intp,q;

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

publicintadd\_i()

{

try

{

System.out.println("Enter two integers");

p=Integer.parseInt(br.readLine());

q=Integer.parseInt(br.readLine());

}

catch(Exception ex){}

return (p+q);

}

public float add\_f()

{

try

{

System.out.println("Enter three floats");

a=Float.parseFloat(br.readLine());

b=Float.parseFloat(br.readLine());

c=Float.parseFloat(br.readLine());

}

catch(Exception ex1){}

return ( a+b+c);

}

}

class Maximum

{

intx,y,z,max;

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

publicintcheck\_max()

{

try

{

System.out.println("Enter three integers");

x=Integer.parseInt(br.readLine());

y=Integer.parseInt(br.readLine());

z=Integer.parseInt(br.readLine());

if(x>y&&x>z)

max=x;

else if(y>z)

max=y;

else

max=z;

}

catch(Exception ex3){}

return max;

}

}

class Test

{

public static void main(String args[])

{

Mathematics m=new Mathematics();

System.out.println("Addition of two integers is "+m.add\_i());

System.out.println("Addition of three floats is "+m.add\_f());

Maximum m1=new Maximum();

System.out.println("Maximum amongst three integers is "+m1.check\_max());

}

}

Output:

Enter two integers

3

4

Addition of two integers is 7

Enter three floats

13.3

4.5

5.5

Addition of three floats is 23.3

Enter three integers

2

3

5

Maximum amongst three integers is 5

Q9. Create a class hierarchy of Employees,Manager and Sales Manager.They Should

have the following functionality.

Employee : Display the name,date of birth and id of the employee.

Manager :Display all above information with the salary drawn.

Sales Manager :Display all above information and commission if applicable.

Commission is applicable if sales >10000 and 10% of the sales.

class Employee

{

String name;

String dob;

int id;

Employee(String name, String dob, int id)

{

this.name=name;

this.dob=dob;

this.id=id;

}

void display()

{

System.out.pritnln("Name :\n"+name);

System.out.pritnln("Date of Birth :\n"+dob);

System.out.pritnln("ID :\n"+id);

System.out.pritnln("Salary :\n"+sal);

}

}

class Manager extends Employee

{

String name;

String dob;

int id;

Manager(String name,String dob,int id,float sal)

{

super(name,dob,id);

this.sal=sal;

}

}

class SalesManager extends Employee

{

float com;

int sales;

SalesManager(String name,String dob,int id,float sal, int sales)

{

super(name,dob,id,sal,sales);

this.sales=sales;

}

void calCom()

{

if(sales>10000)

{

com=sales/100;

}

System.out.println("Commission :\n"+com)

}

}

class Q9

{

public static void main(String args[])

{

Employee emp=new

Employee("Yashu","15-Apr-1999",101,20000.5f,20000);

emp.display();

emp.calCom();

}

}

Q10. create an abstract class Shape and also create Square, Circle and Rectangle

hierarchy to display the area of above shapes.

public abstract class Shape {

public abstract double area();

}

public class Rectangle extends Shape

{

private final double width, length; //sides

public Rectangle(double width, double length)

{

this.width = width;

this.length = length;

}

public double area() {

return width \* length;

}

public class Circle extends Shape

{

private final double radius;

final double pi = Math.PI;

public Circle(double radius)

{

this.radius = radius;

}

public double area()

{

return pi \* Math.pow(radius, 2);

}

}

public class Square extends Shape

{

private final double side;

public Square(double side)

{

this.side=side;

}

public double area()

{

return side\*side;

}

}

public class TestShape {

public static void main(String[] args) {

double width = 5, length = 7;

Shape rectangle = new Rectangle(width, length);

System.out.println("\n Area of rectange="+ rectange.area());

double radius = 5;

Shape circle = new Circle(radius);

System.out.println("\n Area of Circle="+circle.area());

double side=8;

Shape square=new Square(side);

System.out.println("\n Area of Square="+square.area());

}

Q11. Write an interface Stack with methods push and pop.Write a class StringStack which will implement a Stack interface where push will insert string in a string array and pop will remove a string.

Q12. Write a class StackInt which will implement a Stack interface where push will

insert integer value in a integer array and pop will remove a integer.

import java.util.\*;

public class Stack\_int

{

static void stack\_push(Stack<Integer> stack)

{

for(int i = 0; i < 5; i++)

{

int x =(Integer) stack.push(i);

System.out.println(x);

}

System.out.println("Elements Added Successfully....");

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

}

static void stack\_pop(Stack<Integer> stack)

{

System.out.println("Pop Operation:");

for(int i = 0; i < 5; i++)

{

int y = (Integer) stack.pop();

System.out.println(y);

}

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

}

public static void main (String[] args)

{

Stack<Integer> stack = new Stack<Integer>();

stack\_push(stack);

stack\_pop(stack);

}

}

Q13. Define a class Student with attribute rollno,name. Create a subclass Marks of

class Student with attribute marks of Java,DAA,ISA,OT&MC.Define a setMarks()

which set the marks of these subject.Define an interface GraceMarks with field max

grace marks set to 10 and a method displayResult() .Define a class Result which is a

subclass of Marks and implements GraceMarksinterface.displayResult() will provides

a gracemarks if the total marks obtained is less than 160 and if total marks is greater

than 160 then total marks=160.and calculate

percentage and grade accordingly(Assume marks out of 100).

package riza;

import java.util.Scanner;

interface GraceMarks

{

int max\_grace=10;

void displayResult();

}

class Student {

int rollno;

String name;

void set(int r,String n)

{

rollno=r;

name=n;

}

}

class Marks extends Student {

int java,daa,isa,ot,mc;

void setMarks(int j,int d,int i,int o,int m)

{

java=j;

daa=d;

isa=i;

ot=o;

mc=m;

}

}

public class Example extends Marks implements GraceMarks

{

int totalmarks;

int per;

public void displayResult() {

System.out.println(name);

System.out.println(rollno);

totalmarks=java+daa+isa+ot+mc;

if(totalmarks<160 && totalmarks>=150)

{

totalmarks=totalmarks+max\_grace;

if(totalmarks>160)

{

totalmarks=160;

}

System.out.println("Total marks(grace):"+totalmarks);

}

else

{

System.out.println("Total marks: "+totalmarks);

}

per=(totalmarks\*100)/500;

System.out.println("Percentage: "+per);

if(per>=32 && per<=50)

System.out.println("Grade: C");

else if(per>50 && per<=80)

System.out.println("Grade: B");

else if(per<32)

System.out.println("Grade: D ");

else

System.out.println("Grade: A");

}

public static void main(String args[])

{

Scanner sc=new Scanner(System.in);

System.out.println("Enter the total students: ");

int total=sc.nextInt();

Example s1[]=new Example[total];

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

{

s1[i]=new Example();

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

s1[i].name=sc.next();

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

s1[i].rollno=sc.nextInt();

System.out.println("Enter the daa marks: ");

s1[i].daa=sc.nextInt();

System.out.println("Enter the java marks: ");

s1[i].java=sc.nextInt();

System.out.println("Enter the isa marks: ");

s1[i].isa=sc.nextInt();

System.out.println("Enter the ot marks: ");

s1[i].ot=sc.nextInt();

System.out.println("Enter the mc marks: ");

s1[i].mc=sc.nextInt();

}

System.out.println("Result of all students: ");

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

{

s1[i].displayResult();

}

}

}

Q14. Define interface IArithematicOperation include add,subtract,division and

multiplication function.InIBinaryOperation include binaryOR,binaryAND,binaryXOR

and binaryNOTmethods.Define a class Operation which inherits both the interface and

implements interface methods.

import java.util.\*;

interface ArithematicOperation

{

void add();

void subtract();

void div();

void multi();

}

interface BinaryOperation

{

void binaryOR();

void binaryAND();

void binaryXOR();

void binaryNot();

}

class Operation implements ArithematicOperation

{

int n1,n2;

Operation(int n1,int n2)

{

this.n1 = n1;

this.n2 = n2;

}

public void add()

{

System.out.println("Addition : "+(n1+n2));

}

public void subtract()

{

System.out.println ("Subtraction : "+(n1-n2));

}

public void div()

{

System.out.println ("Division : "+(n1/n2));

}

public void multi()

{

System.out.println ("Multiplication: "+(n1\*n2));

}

public void binaryAND()

{

System.out.println("Binary AND: "+(n1&n2));

}

public void binaryOR()

{

System.out.println("binary OR : "+(n1|n2));

}

public void binaryXOR()

{

System.out.println("binary XOR : "+(n1^n2));

}

public void binaryNot()

{

System.out.println("binary Not of n1 : "+(~n1));

System.out.println("binary Not of n2: "+(~n2));

}

}

public class Main

{

public static void main(String[] args) {

Scanner sc = new Scanner (System.in);

System.out.println ("Enter 2 numbers :");

int n1 = sc.nextInt();

int n2 = sc.nextInt();

Operation obj = new Operation(n1,n2);

obj.add();

obj.subtract();

obj.div();

obj.multi();

obj.binaryAND();

obj.binaryOR();

obj.binaryXOR();

obj.binaryNot();

}

}

Q15. Define one dimensional array A and B which are sorted in ascending order.WAP

to merge them into a single sorted array C that contains every item from array A and B

in ascending order.

import java.util.Scanner;

class Example

{

public static void main(String[] args) {

Scanner sc=new Scanner(System.in);

System.out.println("Enter the size of array1:");

int a1=sc.nextInt();

System.out.println("Enter the size of array2:");

int a2=sc.nextInt();

int arr1[]=new int[a1];

System.out.println("Enter elements of array1:");

for(int i=0;i<a1;i++)

{

arr1[i]=sc.nextInt();

}

int arr2[]=new int[a2];

System.out.println("Enter elements of array2:");

for(int i=0;i<a2;i++)

{

arr2[i]=sc.nextInt();

}

int[] result=merge(arr1, arr2);

for (int j=0; j<result.length;j++)

System.out.print(result[j]+ " ");

}

static int[] merge(int[] arr1, int[] arr2)

{

int arr1\_len=arr1.length;

int arr2\_len=arr2.length;

int[] result = new int[arr1\_len + arr2\_len];

int i=0, j = 0;

for(int k = 0 ; k< (arr1\_len + arr2\_len);k++)

{

if ( i >= arr1\_len )

{

result[k] = arr2[j];

j ++;

}

else if ( j >= arr2\_len)

{

result[k] = arr1[i];

i ++;

}

else

{

if ( arr1[i] < arr2[j])

{

result[k] = arr1[i];

i ++;

}

else

{

result[k] = arr2[j];

j ++;

}

}

}

return result;

}

}

Q16. WAP to read a string and a character and count the occurrence if a particular

character in a string.

classJavaExample {

static void countEachChar(String str)

{

int counter[] = new int[256];

intlen = str.length();

for (inti = 0; i<len; i++)

counter[str.charAt(i)]++;

char array[] = new char[str.length()];

for (inti = 0; i<len; i++) {

array[i] = str.charAt(i);

int flag = 0;

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

if (str.charAt(i) == array[j])

flag++;

}

if (flag == 1)

System.out.println("Occurrence of char " + str.charAt(i)

+ " in the String is:" + counter[str.charAt(i)]);

}

}

public static void main(String[] args)

{

String str = "kausain";

countEachChar(str);

}

}

Output:

Occurrence of char k in the String is:1

Occurrence of char a in the String is:2

Occurrence of char u in the String is:1

Occurrence of char s in the String is:1

Occurrence of char i in the String is:1

Occurrence of char n in the String is:1

Q17. WAP which read a string and rewrite it in an alphabetical order.Eg.STRING = GINRST

import java.io.\*;

importjava.util.\*;

class Alphabetic

{

String alphaOrder(String str)

{

char[] charArray=str.toCharArray();

Arrays.sort(charArray);

String aString=new String(charArray);

returnaString;

}

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

{

System.out.println("Enter the String->");

BufferedReaderbr=new BufferedReader(new

InputStreamReader(System.in));

String inputString=br.readLine();

Alphabetic obj=new Alphabetic();

String alphaString=obj.alphaOrder(inputString);

System.out.println("String in the Alphabetic Order :" +alphaString);

}

}

Output:

Enter the String->

kausain

String in the Alphabetic Order :aaiknsu

Q18. The annual examination result of 50 student are tabulated as follow:Define a class

Student and make an array of objects.

Roll no Name Subject 1 Subject 2 Subject 3

1

2

3

……..

50

Write a program to read the data and determine the following:

• Total Marks obtained by each student

• The highest marks in each subject and the roll no of the student who secured it.

• The student who obtained the highest total marks.

package juveria;

import java.util.Scanner;

class Student {

int rollno;

String name;

void set(int r,String n)

{

rollno=r;

name=n;

}

}

class Marks extends Student {

int sub1,sub2,sub3;

void setMarks(int s1,int s2,int s3)

{

sub1=s1;

sub2=s2;

sub3=s3;

}

}

public class Example extends Marks

{

public static void main(String args[])

{

Scanner sc=new Scanner(System.in);

System.out.println("Enter the total students: ");

int total=sc.nextInt();

Example s1[]=new Example[total];

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

{

s1[i]=new Example();

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

s1[i].name=sc.next();

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

s1[i].rollno=sc.nextInt();

System.out.println("Enter marks of subject1: ");

s1[i].sub1=sc.nextInt();

System.out.println("Enter marks of subject2: ");

s1[i].sub2=sc.nextInt();

System.out.println("Enter marks of subject3: ");

s1[i].sub3=sc.nextInt();

}

System.out.println("Total marks obtained by each student: ");

int totalmarks=0,max=0;

Example maxmarks=new Example();

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

{

System.out.println("Name: "+s1[i].name);

totalmarks=s1[i].sub1+s1[i].sub2+s1[i].sub3;

if(totalmarks>max)

{

max=totalmarks;

maxmarks=s1[i];

}

System.out.println("Total marks of "+s1[i].name+":

"+totalmarks);

}

Example max1=new Example();

Example max2=new Example();

Example max3=new Example();

max1=s1[0];

max2=s1[0];

max3=s1[0];

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

{

if(s1[i].sub1>max1.sub1)

max1=s1[i];

if(s1[i].sub2>max2.sub2)

max2=s1[i];

if(s1[i].sub3>max1.sub3)

max3=s1[i];

}

System.out.println("Highest marks in subject1 :");

System.out.println("Roll no: "+max1.rollno +"\nSubject1 marks:

"+max1.sub1);

System.out.println("Highest marks in subject2 :");

System.out.println("Roll no: "+max2.rollno +"\nSubject2 marks:

"+max2.sub2);

System.out.println("Highest marks in subject3 :");

System.out.println("Roll no: "+max3.rollno +"\nSubject3 marks:

"+max3.sub3);

System.out.println("student who obtained highest totalmarks: ");

System.out.println("Name: "+maxmarks.name+" \nRoll no:

"+maxmarks.rollno+"\nTotal Marks: "+max);

}

}

Q19. Create a superclass, Student, and two subclasses, Undergrad and Grad. The superclass Student should have the following data members: name, ID, grade, age, and address. The superclass, Student should have at least one method: boolean isPassed (double grade) The purpose of the isPassed method is to take one parameter, grade (value between 0 and 100) and check whether the grade has passed the requirement for passing a course. In the Student class this method should be empty as an abstract method. The two subclasses, Grad and Undergrad, will inherit all data members of the Student class and override the method isPassed. For the UnderGrad class, if the grade is above 70.0, then isPassed returns true, otherwise it returns false. For the Grad class, if the grade is above 80.0, then isPassed returns true, otherwise returns false. Create a test class for your three classes. In the test class, create one Grad object and one Undergrad object. For each object, provide a grade and display the results of the isPassed method.

importjava.util.Scanner;

class Grade extends Student

{

booleanisPassed(double grade) {

if(grade>80)

return true;

else

return false;

}

}

public class Program19 {

public static void main(String[] args) {

Scanner scan=new Scanner(System.in);

Grade g=new Grade();

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

System.out.println("Enter id, name, city, age :-");

g.setData(scan.nextInt(), scan.next(), scan.next(), scan.nextInt());System.out.println("Enter Grade :-");

if((g.isPassed(scan.nextDouble()))==true)

System.out.println("You passed the exam.");

else

System.out.println("You have not passed the exam.");

UnderGradeug=new UnderGrade();

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

System.out.println("Enter id, name, city, age :-");

ug.setData(scan.nextInt(), scan.next(), scan.next(), scan.nextInt());

System.out.println("Enter Grade :-");

if((ug.isPassed(scan.nextDouble()))==true)

System.out.println("You passed the exam.");

else

System.out.println("You have not passed the exam.");

}

}

abstract class Student

{

private String name,address;

private double grade;

privateintid,age;

voidsetData(intid,Stringname,Stringaddress,int age){

this.id=id;

this.name=name;

this.age=age;

this.address=address;

}

abstractbooleanisPassed(double grade);

}

classUnderGrade extends Student

{

booleanisPassed(double grade) {

if(grade>70)

return true;

else

return false;

}

}

Output:

Enter id, name, city, age :-

101

kausain

pune

21

Enter Grade :-

70

You have not passed the exam.

===========Undergrade============

Enter id, name, city, age :-

102

shifu

mumbai

22

Enter Grade :-

80

You passed the exam.