

Q) Write a program to create a class student with members UN, name, marks (6 subjects). Include methods to accept student details and marks. Also include a method to calculate the percentage and display appropriate details.

Code :-

```

import java.util.Scanner;
class Student {
    private int un;
    private String name;
    private int marks[6];
    Student(int un, String name)
    {
        this.un = un;
        this.name = name;
        this.marks = new int[6];
    }
    public void accept()
    {
        Scanner input = new Scanner(System.in);
        System.out.println("Enter the name");
        this.name = input.next();
        System.out.println("Enter the un");
        this.un = input.nextInt();
        System.out.println("Enter marks of all subjects");
        for (int i = 0; i < 6; i++)
            this.marks[i] = input.nextInt();
    }
    void calculate()
    {
        int sum = 0;
        for (int i = 0; i < 6; i++)
            sum += this.marks[i];
        float percentage = (sum / 600) * 100;
        System.out.println("Percentage is " + percentage);
    }
}

```

```
for (int i=0; i<6; i++)
```

```
{
```

System.out.println("Enter the marks of " + "i'th "+
"subject") ;

```
this.mark[i] = input.nextInt();
```

```
}
```

```
}
```

```
public double calcPercentage()
```

```
{
```

```
int totalmarks = 0;
```

```
for (int i=0; i<6; i++)
```

```
{
```

```
totalmarks = totalmarks + mark[i];
```

```
}
```

```
double percentage = totalmarks / 6;
```

```
return percentage;
```

```
}
```

```
void display()
```

```
{
```

```
System.out.println("The name is " + this.name)
```

```
System.out.println("The mark is " + this.mark);
```

```
b.
```

```
}
```

class main

{

 public static void main(String args[])

{

 System.out.println("no. of students");

 Scanner input = new Scanner(System.in);

 int num = input.nextInt();

 Student [] s = new Student [num];

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

 {

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

 s[i] = new Student (0, " ");

 s[i].accept();

 }

 System.out.println("the details are");

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

 {

 s[i].display();

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

}

Output

Enter the number of students
2

Enter the details of 1 student
after the name

name

UVA

1

marks

30

60

60

60

60

60

Enter the details of 2 student

name

hang

UVA

2

marks

70

70

70

To

The details of student are

The name is Ganesh

The class is 1

The percentage is 55.0%

The name is hanu

The class is 2

The percentage is 70.0%

- ⑤ Create a class Book that contains four members: name, author, price, num-page. Include a constructor to set the values for the members. Include methods to set and get the details of the object. Include a toString() method that could display the complete details of the book.

Code :-

```
import java.util.Scanner
```

```
class Book
```

```
{
```

```
    String name;
```

```
    String author;
```

```
    int price;
```

```
    int num-page;
```

Book() { }

Book(String name, String author, int price, int pages)

{

this.name = name;

this.author = author;

this.price = price;

this.numPages = numPages;

}

void set()

{

Scanner input = new Scanner(System.in);
System.out.print("Enter the name : ")

this.name = input.nextLine();

System.out.print("author : ");

this.author = input.nextLine();

System.out.print("no. of pages : ")

this.numPages = input.nextInt();

System.out.print(" price : ");

this.price = input.nextInt();

}

public String toString()

{

String name, author, price, numPages;

name = "Bookname is " + this.name + "\n";

author = "author name is " + this.author + "\n";
 price = " price is " + this.price + "\n";
 num_page = "no of pages is " + this.num_page +
 "\n";

return name + author + price + num_page;

}

}

class main

{

RVVM(String args[])

{

Scanner input = new Scanner(System.in);
 SDP("No. of books");

int num = input.nextInt();

Book[] b = new Book[num];

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

{

b[i] = new Book();

b[i] = HB();

}

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

{ pop("book " + i + b[i]); }

}

}

Output

Enter the number of books

2

Enter the name of the book

ghost book

author

Gana

number of page

200

price

500

Name

the book allowed

author

hana

page

900

price

30000

book 1 : Book name is the book

author name is gana

price is 500

number of page is 200

book I: in the book closed
author name is hanu,
price is 3000.
number of pages is 000.

Quadratic equation

Solve Quadratic

C

double a, b, c;

~~d = a * b * c;~~

~~if d == 0;~~

Quadratic (a, b, c)

else a == 0

else b == 0

else c == 0

$d = b * b - 4 * a * c;$

void calc()

~~double d = b * b - 4 * a * c;~~

~~if (d >= 0.0)~~

double r1 = (-b + Math.sqrt(d)) / 2.0 * a,

)

else if (d == 0.0)

C double r1 = b1 (woia);
J top (The roof is r1" + r1);

cls.

SOP(The roots are natural)

2

3

clou main

6

asym (string angi[])

Quadratic root - new quadratic WO⁴
root, calc();

1

~~oelbret~~

The roots are not real

4) Shape:

```
import java.util.Scanner;  
abstract class Shape  
{  
    protected int dimension1;  
    protected int dimension2;  
    public Shape (int dimension1, int dimension2)  
    {  
        this.dimension1 = dimension1;  
        this.dimension2 = dimension2;  
    }  
    public abstract void printArea();  
}
```

```
class Rectangle extends Shape  
{  
    public Rectangle (int length, int width)  
    {  
        super (length, width);  
    }  
}
```

```
public void printArea()  
{  
    int area = dimension1 * dimension2;  
    System.out.println ("Area is " + area);  
}
```

{}

class Triangle extends Shape {

 printArea() {

 System.out.println("Area = " + (base * height / 2));

class Circle extends Shape {

 printArea() {

 System.out.println("Area = " + (Math.PI * radius * radius));

class Main {

 public static void main(String[] args) {

 Rectangle rectangle = new Rectangle(4, 5);
 rectangle.printArea();

 Triangle triangle = new Triangle(3, 5);

 triangle.printArea();

 Circle circle = new Circle(7);

 circle.printArea();

Output

Area of Rectangle = 20

Area of Triangle = 9

Area of Circle = 153.94.

5) Bank Java

```

import java.util.Scanner;

public class Bank {
    public static void main(String[] args) {
        Scanner newScanner = new Scanner(System.in);
        System.out.println("Welcome to Bank");
        System.out.print("Name");
        String Name = newScanner.nextLine();
        System.out.print("Enter the account number");
        String accountNumber = newScanner.nextLine();
        SavingAccount savingAccount = new SavingAccount(accountNumber);
        System.out.print("Enter the number of current account");
        CurrentAccount currentAccount = new CurrentAccount(Name,
        accountNumber);
        System.out.print("Deposit amount");
        double depositAmount = newScanner.nextDouble();
        savingAccount.deposit(depositAmount);
        System.out.print("Withdraw amount");
        double withdrawAmount = newScanner.nextDouble();
        savingAccount.withdraw(withdrawAmount);
        System.out.println("Current balance");
        System.out.println(savingAccount.getBalance());
    }
}

```

double currentDeposit (Current Deposit amount),
current Account. display balance);

current Account. display balance();
SOP (Amount to withdraw);

double withdraw = nextDouble();

(curr. display balance));
(cannot. close());

}

}

class Account {

String customerName;

String accountNumber;

double balance;

public void deposit (double amount) {

balance += amount.

}

public void displayBalance()

SOP (balance)

}

class Savings Account extends Account

(if balance > amount)

balance -= amount)

)

```

class CurrentAccount extends Account {
    private double minimumBalance = 1000;
    void withdraw (double amount) {
        if (balance - amount >= minimumBalance)
            balance -= amount;
        else
            System.out.println("Insufficient Balance");
    }
    void imposeServiceCharge() {
        double serviceCharge = 20;
        balance -= serviceCharge;
        System.out.println("Service Charge");
    }
}

```

Output:-

Welcome to the Bank!

Enter name for Account (Savings / current): Gaurav

Enter account number : 1001

current account no.: 1001

performing operation on savings
Account:

Enter amount to deposit into savings
Account: 5987.0

successful

Ent. the amount to withdraw: 347
successful

Balance: 5645.

performing operation on current,
deposit into current account: 55990

successful

withdraw: 5679.0

successful

Balance: 51312.0

88
85
72
23

Two	Page No.
Date:	1 / 1

Package

student Create a package CIE which has two class - student and internal . The class student has members like name, roll no. The class internal derived from student has an array that stores internal mark scored in five courses of the current semester . If the student . Create another package SIZ which has class student .

Student

```
package cie;
public class Student {
    public String name;
    public String roll;
    public int sem;
```

Internals

```
package cie;
public class Internal extends Student {
    public int[] marks = new int[5];
```

Student

```
package sc;
import cc. shidat;
public class External extends Shidat {
    public int[] remarks = new int[5];
}
```

Main

```
import cc. Internals;
import cc. External;
import java. util. Scanner;
```

```
public class Main{
```

```
DSVM (String[] args)
```

```
SOR("Enter the number of students")
```

```
Scanner input = new Scanner(System. in);
int n = input. nextInt();
```

```
Internals [] s1 = new Internals [n];
```

```
Externals [] s2 = new Externals [n];
```

```
int [] finalcc = new int [n];
```

```
int [] finalee = new int [n];
```

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

```
    s1[i] = new Internals();
```

```
    s2[i] = new Externals();
```

```
    System. SOR ("name")
```

```

    SOP( USN )
    SOP( JPM )
    SOP( Marks 15 subjects )
    for ( int i=0; i<5; i++ )
    {
        S(i).marks = input.readInt();
        final circ[i] += S(i).marks(j);
    }
}

for ( int i=0; i<n; i++ )
    JL[i] = new Employee()
SOP( name )
SOP( DOB )
SOP( sex )
SOP( marks )
for ( int j=0; j<5; j++ )
{
    JL[i].sec_marks(j) = input.readInt();
    final ar[i] = S(i).sec_marks(j);
}
}

for ( int i=0; i<n; i++ )
    SOP( name + JL[i].name + " " + marks(i) );

```

rent (i).rem);
JOP ("Internal marks" & finance (i));
JOP ("External marks" & finance (i));
MNC "Total marks & (Maths + Eng)

)
)

Output

Enter the number of subjects

Enter the name

or

Enter the 011
or 3

Enter the 123

Enter the marks of 5 subjects

10

10

10

10

10

Enter the name

ra

Other Reason

x3

Enter the score

3

math

20

20

20

20

20

Name : aa USN : 23 sem 2 3

internal mark : 50

External mark : 20 2

Total mark : 352

- Q) Write a program which creates two threads and thread displaying BMS college of Engineering once every 1 sec and other every 10 second.

Code:

```
class BMSCE extends Thread {
    public void run() {
        while(true) {
            System.out.println("BMS college of Engineering");
            try {
                Thread.sleep(1000);
            }
        }
    }
}
```

```
catch (UncaughtException e) {
    e.printStackTrace();
}
}

class CSLE extends Thread {
    public void run() {
        while (true) {
            synchronized ("CSE") {
                System.out.println("Thread - " + Thread.currentThread().getName());
            }
        }
    }
}

catch (InterruptedException e) {
    e.printStackTrace();
}
}
}
```

```
class CSLE {
    public void run() {
        String args[] = {"new CSLE(.)"};
        CSLE c = new CSLE();
        c.start();
        c.start();
    }
}
```

Output
MU College of Engineers

CSLE

CSLE

CSLE

CSK

BMS College of Engineering

Q) Exception handling

```
class WrongAgeException extends RuntimeException {
    public WrongAgeException(String message) {
        super(message);
    }
}
```

class Father {

```
private int age;
public Father(int age) throws WrongAgeException {
    validateAge(age);
    this.age = age;
}
```

private void validateAge(int age) throws

WrongAgeException {

if (age < 0) {

throw new WrongAgeException("Age

" + age + "

) ;

) ;

```
class Son extends Father {  
    int sonAge;  
    public Son(int fatherAge, int sonAge) throws  
        WrongAgeException {  
        super(fatherAge);  
        validateSonAge(sonAge);  
        try { sonAge = sonAge; }  
    }  
}
```

```
void validateSonAge(int sonAge) throws  
WrongAgeException {  
    if (sonAge == getAge())  
        throw new WrongAgeException();  
}
```

```
public int getSonAge() {  
    return sonAge;  
}
```

```
public static void main(String[] args) {  
    try {  
        Father father = new Father(-5);  
        catch (Exception e) {  
            System.out.println(e);  
        }  
    }
```

try 5

son son. new son (20, 35);

}

catch (WrongAge KXceptional) {

 JDT(* ());

}

}

}

Output:

age cannot be negative

child age cannot be greater than or
equal to father's age.

Done
9/2/24

1) Create label, button and Textfield in a frame using AWT.

```
import java.awt.*;
```

```
import java.awt.event.*;
```

```
public class AWTExample extends
```

```
WindowAdapter
```

```
Frame f;
```

```
AWTExample() {
```

```
f = new Frame();
```

```
f.addWindowListener(this);
```

```
label d = new label("Employee id");
```

```
Button b = new Button("Submit");
```

```
Textfield t = new Textfield();
```

```
d.setBounds(20, 80, 80, 30);
```

```
b.setBounds(20, 100, 20, 30);
```

```
t.setBounds(100, 100, 200, 30);
```

```
f.add(d);
```

```
f.add(b);
```

```
f.add(t);
```

```
f.setSize(400, 300);
```

```
f.setTitle("Employee Info");
```

```
f.setLayout(null);
```

```
f.setVisible(true);
```

}

```

public void windowClosing(WindowEvent e) {
    System.exit(0);
}

public static void main(String[] args) {
    AWTExample awtobj = new AWTExample();
}

```

- 2) Create a button and add a action listener for Mouse click.

```

import java.awt.*;
import java.awt.event.*;
public class EventHandling extends Event
WindowAdapter implements ActionListener {
Frame f;
TextField tf;
EventHandling() {
f = new Frame();
f.addWindowListener(this);
tf = new TextField();
tf.setBounds (60, 50, 170, 10);
Button b = new Button("click me");
b.setBounds (100, 120, 80, 30);
b.addActionListener(this);
}
}
```

```
f.add(8);  
f.add(11);  
f.setLayout(null);  
f.setvisible(true);  
  
public void actionperformed(ActionEvent e)  
{  
    System.out("Welcome");  
}  
  
public void windowClosing(WindowEvent e)  
{  
    System.exit(0);  
}
```

```
psVM(SAP){  
    new EventHandling();  
}
```

Program on ZO

1) Example 1.



PREMIUM
KING SIZE

```

import java.io.*;
public class ByteArrayInput {
    PSVM(JC) This will IO operations
    byte[] buf = {35, 36, 37, 38};
    Byte ArrayInputStream buf = new Byte Array
    Input Stream(buf);
    int k > 0
    while (k = byte.read() != -1) {
        char ch = (char) k;
        System.out.println("ASCII value of character is "
            + k); // Special character is "ch";
    }
}

```

Output

ASCII value of character is 35;
 special character is #.
 ASCII value of character is 36;
 special character is \$.
 ASCII value of character is 37;
 special character is %.
 ASCII value of character is 38
 special character is & f

Example :-

```
import java.io.FileInputStream;
import java.io.IOException;
```

```
public class FileIO {
    public static void main(String[] args) throws IOException {
        FileInputStream fin = new FileInputStream("Example.txt");
        byte[] bytes = new byte[20];
        int i;
        char c;
        i = fin.read(bytes);
        System.out.println("Number of bytes read: " + i);
        System.out.print("Bytes read: ");
        for (byte b : bytes) {
            c = (char) b;
            System.out.print(c);
        }
    }
}
```



PREMIUM
KING SIZE

Output

Number of bytes read : 30

Bytes read : Hey this is a new file.

CRS
11/3/24