

```
class addCalc {
```

```
    int double x, y, z;
```

```
    class add() {
```

```
        x = 20;
```

```
        y = 50;
```

```
        z = x+y;
```

```
        sout ("The sum is " + z);
```

```
}
```

```
class sub() {
```

```
    x = 20;
```

```
    y = 50;
```

```
    z = x-y;
```

```
    sout ("the subtraction is " + z);
```

```
class mult() {
```

```
    x = 20;
```

```
    y = 50;
```

```
    z = x*y;
```

```
    sout ("the product is " + z);
```

```
class div() {
```

```
    x = 20;
```

```
    y = 50;
```

```
    z = x/y;
```

```
    sout ("The quotient is " + z);
```

```
class Run()
```

```
    psvm (String args[]) {
```

```
        Calc c1 = new Calc();
```

```
        c1.add();
```

```
        c1.sub();
```

```
        c1.mult();
```

```
        c1.div();
```

O/p \Rightarrow the sum is 70

the sub is -30

the product is 1,000.

the quotient is 0.4.

(2) Fibonacci series.

class Ex {

psvm (String args[]) {

int n=10, term1=0, term2=1;

sout ("Fibonacci series") titl ("no. of terms");

for (int i=1; i<n+1; i++) {

sout (term + " ");

int term3 = term1 + term2;

term1 = term2;

term2 = term3;

}

O/p \Rightarrow Fibonacci series : fill 10 terms

0, 1, 1, 2, 3, 5, 8, 13, 21, 36.

(3) Prime Number

class Prime {

psvm (String args[]) {

int n=20, flag=0;

for (int i=2; i<=n/2; i++) {

if (n%i==0) {

flag=1;

break;

} if (flag>0) {

sout (n + " is a prime no."); }

```
else {
```

```
    sout(m + " is not a prime number");
```

```
}
```

```
}
```

Output: 29 is a prime number.

Week 2

④

Roots of Quadratic Equation II

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

```
class calc {
```

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

```
    double a, b, c;
```

```
    sout("Enter coefficients of equation");
```

```
    Scanner s = new Scanner(System.in);
```

```
    a = s.nextInt();
```

```
    b = s.nextInt();
```

```
    c = s.nextInt();
```

```
    double root1, root2;
```

```
    double det = b * b - 4 * a * c;
```

```
if (det > 0) {
```

```
    root1 = (-b + Math.sqrt(det)) / (2 * a);
```

```
    root2 = (-b - Math.sqrt(det)) / (2 * a);
```

```
    sout("root1 = " + root1 + "root2 = " + root2);
```

```
} else if (det == 0) {
```

```
    root1 = root2 = -b / (2 * a);
```

```
    sout("root1 = root2 = " + root1);
```

} else {

double real = -b / (2*a);

double img = Math.sqrt(-det) / (2*a);

sout (" root1 = " + real + " + " + imaginary + "i");

sout (" root2 = " + real + " - " + imaginary + "i");

op>> Enter co-efficients of the equation

10, 12, 4

root1 = -0.6 + 0.86i

root2 = -0.6 - 0.86i

(5)

Grocery class Grocery {

double q_dal; pulses, sugar;

Grocery () {

dal = 1;

pulses = 2;

Sugar = 3;

double cal_amount () {

double totalAmount = (dal * 50) + (pulses * 80) + (Sugar * 100);

sout (" Total Amount " + totalAmount);

return totalAmount;

class grocery Run {

param (String arg0) {

Grocery q1 = new Grocery ();

double v = q1.cal_amount ();

sout ("The total amount is: " + v);

Q/p: Total amount = 460.

Solve
22/12/23

12/01/2024

Import java.util.*;

```
class student {  
    int n;  
    String USN;  
    String Name;  
    String stud[] = new String [n];  
    double marks [] = new double [6];
```

```
void input() {  
    Scanner sc = new Scanner (System.in);  
    System.out.println ("Enter USN");  
    USN = sc.nextLine();  
    System.out.println ("Enter your Name");  
    Name = sc.nextLine();  
    System.out.println ("Enter marks of 6 subjects");  
    for (int i = 0; i < 6; i++) {  
        marks [i] = sc.nextInt();  
    }  
}
```

```
double percentage() {  
    double per, sum = 0;  
    int i;  
    for (i = 0; i < 6; i++) {  
        sum = sum + marks[i];  
    }
```

```
per = (sum / 600) * 100;  
System.out.println("sum");  
return per;
```

```
}
```

```
class Run {
```

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

```
int n;
```

```
sout ("Enter no. of students");
```

```
Scanner sc = new Scanner (System.in);
```

```
n = sc.nextInt();
```

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

```
Student si = new Student();
```

```
si.input();
```

```
double x = si.percentage();
```

```
sout ("Percentage : " + x);
```

```
}
```

```
}
```

O/P: Enter no. of students : |

Enter u/n : 328

Enter Name : 28 Karan

Enter marks : 98 97 98 99 98 98

percentage : 97.7

Ques
Date : 12/11/24

b) Book Display

```
import java.util.*;  
class Books {  
    String name;  
    String author;  
    int price;  
    int numPages;  
    Books();  
    Books (String name, String author, int price, int numPages)  
    {  
        this.name = name;  
        this.author = author;  
        this.price = price;  
        this.numPages = numPages;  
    }  
    public String toString()  
    {  
        String name, author, price, numPages;  
        name = "Book Name" + this.name + "\n";  
        author = "Author Name:" + this.author + "\n";  
        price = "Price :" + this.price + "\n";  
        numPages = "No. of pages :" + this.numPages + "\n";  
        return name + author + price + numPages;  
    }  
}
```

class Main

{
public static void main (String args[])

{

Scanner s = new Scanner (System.in);

int n, price, numpages;

String name, author;

cout ("Enter number of books");

n = s.nextInt();

Books b[];

b = new Books [n];

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

cout ("Books " + (i+1) + ": ");

cout ("Enter name of book: ");

name = s.next();

cout ("Enter author: ");

author = s.next();

cout ("Enter price: ");

price = s.nextInt();

b[i] = new Books (name, author, price, numpages);

}

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

cout ("Books " + (i+1) + ": " + b[i]);

}

}

(20) Registration of a company - Super User

(20) User can login & logout from system



Op:

Enter no. of books : 1

Enter the name of book : (i)

Book 1:

Enter name of book : Peace

Enter Author : Christopher Nolan

Enter price : 450

Enter number of pages : 222

Book 1:

Book name : Peace & Author name : Christopher Nolan

Author name : Christopher Nolan

Price : 450

Enter number of pages : 22.

~~3.2nd edition published 2002~~

~~3rd edition published~~

~~(3rd edition for expression of intent) due~~

~~3rd edition due on 3rd week~~

~~3rd week from 3rd week~~

~~3rd week from 3rd week~~

~~3rd week due on 3rd week~~

~~3rd week due~~

~~(3rd week 3rd week) due~~

~~3rd week last week~~

~~3rd week last week~~

~~3rd week last week~~



19/01/24

① Area using Abstract.

» `import java.util.Scanner;`

`class Input {`

`Scanner s = new Scanner(System.in);`

`}`

`abstract class Shape extends Input {`

`double x,y;`

`void display();`

`}`

`class Rectangle extends Shape {`

`double Rectangle() {`

`sout ("Enter the dimensions of Rectangle h & w");`

`double x = s.nextDouble();`

`y = s.nextDouble();`

`return x*y;`

`}`

`}`

`class Triangle extends Shape {`

`double Triangle() {`

`sout ("Enter the height & base (in)");`

`x = s.nextDouble();`

`y = s.nextDouble();`

`return 0.5*x*y;`



```
class Circle extends Shape {  
    double circle() {  
        cout << "Enter the radius: ";  
        cin >> r;  
        return 3.14 * r * r;  
    }  
}
```

```
class Run {  
public:  
    sum (string args) {
```

```
        Rectangle R1 = new Rectangle();  
        Triangle t1 = new Triangle();  
        Circle c1 = new Circle();  
  
        double re = R1.rectangle();  
        cout << "The area of rectangle is: " + re; //  
  
        double tr = t1.triangle();  
        cout << "The area of triangle is: " + tr; //  
  
        double ci = c1.circle();  
        cout << "The area of circle is: " + ci; //
```

```
} }
```

O/P \Rightarrow Enter height and width of rectangle: 10 20
the area is 200

Enter height and base of triangle: 10 12
the area is 60

Enter radius of circle: 3

the area is 28.27



② Bank Details

public class Account {

String cname;

int aceno;

String acctype;

double balance;

public Account (String cname, int aceno, String acctype, double balance)

{

this.cname = cname; this.aceno = aceno; this.acctype = acctype;

this.balance = balance;

}

void deposit (double amt)

{

balance += amt;

sout ("Updated balance = " + balance);

}

void balance()

{

sout ("Account balance: " + balance);

}

void withdraw (double amt)

{ if (balance >= amt) {

balance -= amt;

sout ("withdraw completed, updated balance: " +

```

    else
        sout ("Insufficient balance");
    }
}

class Current extends Account {
    private double minBal = 1000;
    private double serviceCharge = 50;

    public Current (String name, int acNo, double balance) {
        super (name, acNo, acType: "current", balance);
    }

    public void checkMinBal() {
        if (balance < minBal) {
            balance -= serviceCharge;
            sout ("Service charge imposed. balance" + balance);
        }
        else
            sout ("Balance above min balance");
    }
}

class Savacct extends Account {
    double rate = 0;

    Savacct (String name, int acNo, double balance) {
        super (name, acNo, acType: "savings", balance);
    }
}

```

void depositInterest()

double interest = balance * (rate/100);

balance += interest;

set ("Interest deposited, balance:" + balance);

}

class Main

PSVM () {

SavAct SA = new SavAct ("John Doe", 1234, 1000);

CurAct CA = new CurAct ("Jane Smith", 18910, 2000.0);

SA. deposit (500);

SA. balance ();

SA. depositInterest ();

SA. withdraw (200);

CA. deposit (2000);

CA. balance ();

CA. withdraw (1500);

CA. checkMBal ();

CA. withdraw (100);

CA. checkMBal ();

}

O/P \Rightarrow

Updated balance: 800.0

Account balance: 800.0

820.0

~~828.0~~

Updated balance: 6000.0

Account balance: 6000

Insufficient Balance funds

Balance above min balance

withdrawal completed updated bal: 900

Service charge imposed balance: 800

~~Service charge imposed
19/1/24~~

Lab 6

Packages

GDB Package.

```
package GDB;
public class personal {
    String USN;
    String name;
    int sem;
    public personal (String USN, String name, int sem) {
        this.USN = USN;
        this.name = name;
        this.sem = sem;
    }
}
```

```
import java.util.Arrays;
```

```
public class Internals {
    public int[] internal_marks;
    this.internal_marks = internal_marks;
}
```

```
package SBE
```

```
import GDB.personal;
```

```
public class External extends personal {
    public int[] see_marks;
```

```
public External (String USN, String name, int sem,  
int[] seemarks)
```

```
super (USN, name, sem);
```

```
this.sem = sem;
```

```
}
```

Package Main

```
import java.util.Arrays;
```

```
import DB.Internals;
```

```
import SEB.Personal;
```

```
public class main {
```

```
psvm (String args[]) {
```

```
int n = 3;
```

```
student [] students = new student [n];
```

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

```
{ int internalmarks [] = { 80, 75, 90, 85, 95 };
```

```
int seemarks [] = { 70, 80, 85, 85, 90 };
```

```
student p1 = new student ( new personal ("USN" +
```

```
"Student " + i, ? ),
```

```
new Internals ( internalmarks ) );
```

```
students [] . see = new External ( "USN" + i, "Student " + i, seemarks )
```

```
}.
```

for (int i=0; i < student.length; i++)
Student student1 = student[i];
cout ("Student " + student1.name + " Internal " + ArrayToString
(student1.Internal_marks) + " SB " + ArrayToString
(student1.Sem_marks));

} }
Static class Student{

public personal personal;
public Internal Internal;
public External External;

public Student (personal p1, Internal I1);

personal = p1;
Internal = I1;

} }

Output :

Student : student Q

Internal : {80, 70, 90, 85, 98}

Sem : {70, 80, 85, 85, 90}

Student : student J

Internal = {80, 70, 90, 84, 96}

SB : {70, 90, 85, 64, 98}



Threads

class Mthread extends Thread

```
String m;
int i;
```

```
Mthread (String m, int i)
```

```
{  
    m = m;  
    i = i;
```

```
public void run () {
```

```
try {
```

```
while (true) {
```

```
System.out.println (message);
```

```
Thread.sleep (interval);
```

```
} catch (InterruptedException e) {
```

```
e.printStackTrace ();
```

```
class ThreadDemo {
```

```
psvm (String args) {
```

```
Mthread t1 = new Mthread ("BMS", 10000);
```

```
Mthread t2 = new Mthread ("CSB", 1000);
```

```
t1.start ();
```

```
t2.start ();
```

o/p →

BMS

CSB

CSB

CSB

CSB

CSB

BMS.

Exceptions

class wrongAge extends Exception {
 wrongAge (String s) {
 super(s);
 }
}
}
class Father {
 int age;
 Father (int age) throws wrongAge {
 if (age < 0)
 throw new wrongAge ("Age is less");
 this.age = age;
 }
}
class Son extends Father {
 int sage;
 son (int fage, int sage) throws wrongAge {
 super(fage);
 this.sage = sage;
 if (sage >= fage)
 throw new wrongAge ("Son's age is more than father");
 if (sage < 0)
 throw new wrongAge ("Age is invalid");
 }
}

public class Person {

 String name;

 toy f;

 father f = new Father(00);

 son s = new Son(60, 6);

 sout (or Rather : " + age + "son" + sage);

 catch (wrong age e) {

 sout(e);

}

if \Rightarrow wrong age : age < lesser share

if son age > 10

wrong age : son age is more

check
check
check

(age, age) greater than

(old, old) less than

(old) less than

(new) all right

{

 } + (new) greater than old

3) Creating label, button and textfield in a frame using AWT.

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

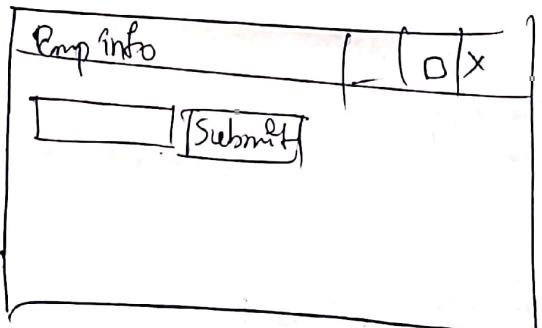
```
public class AWTEx extends WindowAdapter {
    Frame f;
    AWTEx () {
        f = new Frame();
        f.addWindowListener (this);
        Label l = new Label ("Employee Id:");
        Button b = new Button ("Submit");
        Textfield t = new Textfield();
        l.setBounds (20, 80, 80, 30);
        t.setBounds (20, 100, 80, 30);
        b.setBounds (100, 100, 80, 30);
        f.add (b);
        f.add (l);
        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){
```

```
    AWTEx awtObj = new AWTEx ();
```

```
}
```

```
}
```



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

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

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

```
public class Event extends WindowAdapter implements ActionListener
```

```
Frame f;
```

```
Textfield tf;
```

```
Event () {
```

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

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

```
tf = new TextField();
```

```
tf.setBounds (60,50,170,20);
```

```
Button b = new Button ("click me");
```

```
b.setBounds (200,120,80,30);
```

```
b.addActionListener (this);
```

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

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

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

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

```
}
```

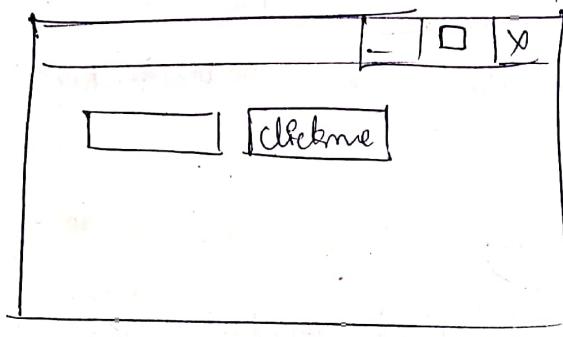
```

public void actionPerformed (ActionEvent e) {
    if (e.getSource() == tf) {
        tf.setText ("welcome");
    }
}

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

public static void main (String args[]) {
    new EventHandling ();
}

```



Programs on IO

Example 1:

```

import java.io.*;
public class ByteArrayInput {
    public void main () throws IOException {
        byte [] buf = {35, 36, 37, 38};
    }
}

```

ByteArrayInputStream byt = new ~~ByteArrayInputStream~~^(buf)

int k = 0;

while (k <= byt.read ()) { }

char ch = (char) k;

System.out.println ("ASCII value of char is: " + k);

System.out.println ("Special char is: " + ch);

}

}

}

Example ③

```
public class FileEx3 {  
    public void psvm() throws IOException {  
        fileInputStream fin = new FileInputStream  
            ("Example.txt");  
        int content;  
  
        System.out.println("Remaining bytes that can be read: " + fin.  
            available());  
        content = fin.read();  
        System.out.println((char) content + " ");  
        System.out.println(content + " ");  
        System.out.println("Remaining bytes that can be read: " +  
            fin.available());  
    }  
}
```

Ex 4

```
import java.io.FileInputStream;  
import java.io.IOException;  
  
public class FileEx4 {  
    public void psvm() throws IOException {  
        fileInputStream fin = new FileInputStream("Example.txt");  
        byte bbytes = new byte[20];  
        int i;  
        char c;  
        i = fin.read(bbytes);  
        System.out.println("Number of bytes read: " + i);  
        System.out.println("Bytes read");  
        for (byte b : bbytes) {  
            c = (char) b;  
            System.out.print(c);  
        }  
    }  
}
```

Example 2

```
import java.io.*;
public class ByteArrayEx {
    private() throws IOException {
        FileOutputStream fout1 = new FileOutputStream("Example.txt");
        FileOutputStream fout2 = new FileOutputStream("Example2.txt");
        ByteArrayOutputStream bout = new ByteArrayOutputStream();
        bout.write(65);
        bout.writeTo(fout1);
        bout.writeTo(fout2);
        bout.flush();
        bout.close();
        sout("success.");
    }
}
```

Output

- Ex \Rightarrow 1 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: &.

Ex(3) \Rightarrow R file is closed!

Remaining bytes that can be read : 11

FILE WORD !

Remaining bytes that can be read 10

Ex(4) \Rightarrow Number of bytes read : 1
Bytes read : A.

Ex(5) \Rightarrow ofp

Success . . .

~~Done~~
23/2/24