

## Program 1

```
Class Grocery {
    String c-name;
    String c-phone;
    int Total;

    Grocery (String Name; String Phone)
    {
        c-name = Name;
        c-phone = Phone;
    }

    void total (int dal, int pulses, int sugar)
    {
        Total = dal * 100 + pulses * 80 + sugar * 50;
    }

    void display ()
    {
        System.out.println ("Name = " + c-name + "\nPhone - Number = "
            + c-phone + "\nTotal amount = " + Total);
        System.out.println ();
    }
}
```

## Class Run {

```
public static void main (String ar[])
{
    Grocery G1 = new Grocery ("Varsha", "12347688");
    Grocery G2 = new Grocery ("Sneha", "12347688");
    Grocery G3 = new Grocery ("Siri", "12347688");

    G1.total (1, 2, 3);
    G1.display ();
    G2.total (2, 3, 4);
    G2.display ();
    G3.total (3, 4, 5);
    G3.display ();
}
```

## Output

Name : Varsha

Phone - Number : 123456789

Total - amount : 410

Name : Sneha

Phone - Number : 123456789

Total - amount : 640

Name : Siri

Phone - Number : 123456789

Total - amount : 870

## \* Program 2.

Class Overload {

void number (int n)  
{

int sum = 0;

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

System.out.println ("Sum = " + sum);

void number (int n, int m)

{

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

int flag = 0;

for (int j=2; j<=i; j++)  
{

if (i % j == 0)

```
{  
    flag = 1;  
    private;  
}  
  
{  
    if (flag == 0)  
        System.out.println(i);  
}  
  
{  
class sum &  
public static void main (String ar[])  
{
```

Output:  
Sum = 10  
1, 3, 5, 7

### \* Program 3

```
import java.util.Scanner;  
class Quad {  
    int a, b, c;  
    double root1, root2, d;  
    Scanner s = new Scanner (System.in);  
    void input()  
    {  
        System.out.println("Quadratic Equation");  
        System.out.print("Enter a = ");  
        a = s.nextInt();  
        System.out.print("Enter b = ");  
        b = s.nextInt();  
    }
```

```
System.out.print("Enter a:");  
a = sc.nextInt();
```

{

```
void discriminant(){
```

$$d = (b^2) - (4 * a * c);$$

{

```
void calculateRoots(){
```

```
if (d > 0)
```

{

```
System.out.println("Roots are real and unequal");
```

$$\text{root1} = (-b + \sqrt{d}) / (2 * a);$$

$$\text{root2} = (-b - \sqrt{d}) / (2 * a);$$

```
System.out.println("First root is " + root1);
```

```
System.out.println("Second root is " + root2);
```

```
else if (d == 0)
```

{

```
System.out.println("Roots are real and equal");
```

$$\text{root1} = (-b + \sqrt{d}) / (2 * a);$$

```
System.out.println("Root " + root1);
```

{

```
else
```

{

~~```
System.out.println("No real solutions. Roots  
are imaginary");
```~~

~~$$\text{double real} = -b / (2 * a);$$~~

~~$$\text{double imaginary} = \sqrt{-d} / (2 * a);$$~~

~~```
System.out.println("The Eq has 2 complex roots")
```~~

{

{

```
Class Main {
```

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

```
Quad q = new Quad();  
q = input();  
q = discriminant();  
q = calculateRoots();  
q.
```

Output

Enter variable i  
-4

-4

Roots are Equal

$$x = 2.00$$

~~$$\sqrt{\frac{-b}{2a} \pm \sqrt{\frac{b^2 - 4ac}{4a^2}}}$$~~

12/1/24

Date \_\_\_\_\_  
Page \_\_\_\_\_

Q. 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 Name:");  
Name = sc.nextLine();
```

```
System.out.println("Enter marks of 6 subjects:");  
for (int i = 0; i < 6; i++) {
```

```
marks[i] = sc.nextInt();
```

```
}
```

}

```
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 Runs

```
public static void main (String[] args) {  
    int n;  
    System.out.println ("Enter the 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 ();  
        System.out.println ("percentage of "+ si.name + " with  
        USN "+ si.USN + " is  
        "+ x);  
    }  
}
```

### Book data base

Output :

Enter the no of students 2

Name : Varsha

USN : IBM22

Marks : 100 100 100 100 100 100

percentage of Varsha with IBM22 is 100.0

### 1. Book data base

### Class Books

{

String name;

String author;

int piece;

int numpages;

Book(name, author, price)

{  
    this.name = name;  
    this.author = author;  
    this.price = price;  
}

public String toString()

{  
    String name, author, price;

    name = "Book name: " + this.name;

    author = "Author name: " + this.author;

    price = "Price: " + this.price + "\n";

    numPages = "no of pages: " + this.numPages;  
    return name + author + price + numPages;  
}

1  
Class Main

{  
    public static void main (String args)

    Scanner s = new Scanner (System.in);  
    int n;

    String name;

    String author;

    int price;

    System.out.print ("Enter the no of books: ");

    n = s.nextInt();

    Books b[];

    b = new Books[n];

    for (int i=0; i<n; i++)

{

+ (i+1)

    System.out.println ("Book: ");

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

```
name = s.next();
System.out.print("Enter author : ");
author = s.next();
System.out.print("Enter price : ");
price = s.nextInt();
System.out.print("Enter no of pages : ");
numPages = s.nextInt();
b[i] = new Books(name, author, price, numPages);
}
}
```

8/11/2014

0 →

19/1/24

Lab 3

### Area Calculation

1. abstract class Shape {  
double length, breadth;  
Shape (double l, double b)  
{  
long a = l;  
b = b;  
}}

Shape (double a)  
{  
a = a;

abstract void printArea();  
}

Class Rectangle extends Shape  
{

Rectangle (double l, double b)  
{

super (l, b);  
}

void printArea()  
{

double area;

area = a \* b;

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

}

Class Triangle extends Shape  
{

Triangle (double b, double h)

```
{  
    super(b, h);  
}  
void printArea()  
{  
    double area;  
    area = (a + b) / 2;  
    System.out.println("Area of triangle = " + area);  
}
```

Class Circle - extends Shape

```
{  
    Circle(double r)  
}
```

```
{  
    super(r);  
}
```

```
void printArea()  
{
```

```
    double area;  
    area = 3.14 * r * r;  
    System.out.println("Area of Circle = " + area);  
}
```

Class run {

    public static void main (String[] args)  
    {

        Rectangle r = new Rectangle(2, 4);  
        Triangle t = new Triangle(3.1, 2.3);  
        Circle c = new Circle(2.2);  
        Shape s;

        s = r;

        s.printArea();

        s = t;

```
s.println();
```

```
s = c;
```

```
s.println();  
}  
}
```

Output

Area of Rectangle = 23.3

Area of triangle = 42.3

Area of Circle = 23.3

## Bank Account

2.

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

```
Class Account {
```

```
String CustomerName;
```

```
long accno;
```

```
String accountType;
```

```
double balance;
```

```
Account (String CustomerName, long accno)
```

~~this.CustomerName = CustomerName;~~

~~this.accno = accno;~~

~~this.accountType = accountType;~~

~~this.balance = 0.0;~~

```
}
```

~~void displayBalance()~~

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

```
System.out.println ("Customer Name: " + customerName);
```

```
System.out.println ("Account Type: " + accountType);
```

```
System.out.println ("Balance $ " + balance);
```

Class Current extends Account {  
double minBalance;  
double serviceCharge;  
Customer (String customerName, long accno)  
{

super (CustomerName, accno, "Current");  
this.minBalance = 500.0;  
this.serviceCharge = 50.0;  
}

void withdraw (double amount) {  
if (balance - amount >= minBalance)

balance -= amount;

System.out.println ("Withdrawal successful.  
Current Balance: \$" + balance);

}

else

{

System.out.println ("Insuff funds");

}

accountType) void imposeServiceCharge() {

if (balance < minBalance) {

balance = serviceCharge;

SOP ("Service charge \$" + balance);

}

}

}

Class SavAcct extends Account {

double interestRate;

SavAcct (String customerName, long accno)

{

super (CustomerName, accno, "Savings");

this.interestRate = 0.05;

}

```
void displayInterest() {  
    double interest = balance * interestRate;  
    balance += interest;  
    SOP("Interest displayed");  
}
```

}

class Bank2

```
{ public static void main(String[] args) {  
    Scanner scanner = new Scanner(System.in);  
    SOP("Choose account type:");  
    SOP("1. Current");  
    SOP("2. Savings");  
    SOP("Enter choice (1 or 2):");  
    int choice = scanner.nextInt();  
    System.out.print("Enter customer name: ");  
    String CustomerName = scanner.next();  
    SOP("Enter Account number:");  
    long acno = scanner.nextInt();  
    if (choice == 1) {
```

~~CustAcct~~ = new CustAcct(CustomerName, 0);  
SOP("Enter initial balance: \$");  
double initialBalance = scanner.nextDouble();

~~CustAcct~~.balance = initialBalance;  
SOP("Enter withdraw amount \$");  
}

else if (choice == 2)

~~favAcct~~ SavAccount = new SavAcct(CustomerName, 0);  
SOP("Enter initial balance");  
double initialBalance = scanner.nextDouble();

SOP("Enter withdraw amount \$");  
SOP("Withdrew amount \$");

SOP("Withdraw Successful")  
System.out.print("Enter Interest Rate: ")  
double interestRate = scanner.nextDouble();  
Scanner account = displayBalance();  
System.out.print("Enter term (in years) ")  
int term = scanner.nextInt();  
Savaccout compoundInterest(initialBalance, term);  
Savaccout.displayBalance();  
}  
else  
SOP("Invalid choice");  
}

## Output

Choose account type:

1. current

2. savings

Enter choice (1 or 2) : 1

Enter customer name: Vanha

Enter account number: 123

Enter initial balance: \$2000

Enter withdrawal amount: \$300

Withdrawal successful. Current Balance: \$1700

*Debit  
19/11/24*

16/2/24  
Lab 4

1. Write a program which creates two threads  
① BMS College of Engineering,  
② "CSE" once every two seconds

Class Thread1 Extends Thread {

```
public void run()
{
    try {
        for (i=0; i<10; i++)
        {
            SOP("BMS college of Engineering");
            Thread.sleep(10000);
        }
    } catch (InterruptedException e) {
        SOP("Exception occurred");
    }
}
```

Class Thread2 Extends Thread {

```
public void run()
{
    try {
        for (i=0; i<30; i++)
        {
            SOP("CSE");
            Thread.sleep(2000);
        }
    } catch (InterruptedException e) {
        SOP("Exception 2 occurred");
    }
}
```

Class Call {

public static void main(String[] args)  
{

Thread t1 = new Thread1();

Thread t2 = new Thread2();

t1.start();

t2.start();

}

}

- Q. Create Base class called Father and derived class called son. Use defined Exception WrongAge(); throws Exception when input age < 0 (father class) and son's age >= father's age (son class).

Class Main

{

public static void main(String[] args)

{

try {

Son s = new Son(60, 41);

},

catch (WrongAgeException e)

{

System.out.print(e);

}

}

}

Class WrongAgeException extends Exception {

public WrongAgeException(String str) {

super(str);

}

,

class Father {

int fatherAge;

Father(int Age) throws WrongAgeException

{

if (Age < 0)

throw new WrongAgeException("Invalid age - Father")

else

fatherAge = Age;

}

}.

Class Son extends Father {

int SonAge;

Son(int sAge, int fAge) throws WrongAgeException

{

super(fAge);

if (sAge >= fatherAge || sAge < 0)

throw new WrongAgeException("Invalid age - Son")

else {

SonAge = sAge;

SOP("Father's Age: " + fatherAge + " Son's Age: " + SonAge)

}

}.

### 3. Package Program

package CIE

public class student {

public String CN;

public String name;

public int sem;

public student (String USN, String name, int sem)  
{

    this.USN = USN;

    this.name = name;

    this.sem = sem;

}

public class Internals {

    public int[] marks;

    public Internals (int[] marks) {

        this.marks = marks;

}

}

package SEE;

import CIE.student;

public class External extends Student {

    public int[] marks;

    public External (String USN, String name, int sem  
                  int[] marks) {

        super(USN, name, sem);

        this.marks = marks;

}

}

import CIE.student;

import SEE.External;

public class Main {

    public class Main {

        public static void main (String[] args)

            Student s1 = new Student ("320", "Var", 3)

            int[] marks = {96, 92, 95, 90, 91};

            External s2 = new External ("IBN12", "Var", 3)

            marks

SOP("student1");  
SOP("USN" + S1.USN);  
SOP("Name" + S1.name);  
SOP("student 2.");  
SOP("USN" + S2.USN);  
SOP("Name" + S2.name);  
SOP("Sem" + S2.Sem);

3

2

## Output of Program 1

1) BMS college of Engineering

CSE

CSE

CSE

CSE

BMS college of Engineering  
by it will continue.

2) Father's Age : 60 Son's Age 41

3) Student 1

USN : 320

Name : Varsha

Student 2

USN : 321

Name : Var

*Sister*  
16 Jan 14

10

23/2/24

Date / /  
Page

## Program 1

Creating label, button and TextField in 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 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) {  
    AWTExample awtObj = new AWTExample();
```

```
}
```

```
}
```

## Output

Employee ID  
[ ] Submit

2. import java.awt.\*;  
import java.awt.event.\*;  
public class EventHandling extends WindowAdapter  
frame f;  
TextField tf;  
EventHandling(){  
f = new Frame();  
f.addWindowListener(this);  
tf = new TextField();  
tf.setBounds(60, 50, 170, 20);  
Button b = new Button("click me");  
b.setBounds(100, 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){  
tf.setText("Welcome");

3.

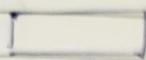
public void windowClosing(WindowEvent){  
System.exit(0);

3.

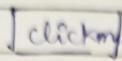
public static void main(String args[]){  
new EventHandling();  
}

}

### Output



Welcome



clickme



3. import java.io.\*;  
public class ByteArrayInputs  
public static void main (String [ ] args) throws  
IOException {

byte [ ] buf = { 35, 36, 37 };

ByteArrayInputStream bft = new ByteArrayInputStream  
int k=0

while ( (k = bft.read ()) != -1 ) {

char ch = (char) k

System.out.println ("Ascii " + k + " ch ");

}

}

### Output

→ Ascii 35, #

→ Ascii 36, \$

→ Ascii 37, %

4. public class FileEx {

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

FileInputStream fin = new FileInputStream ("Exam.txt")  
int content;

System.out.println ("Remaining bytes : " + fin.available ())  
content = fin.read ();

System.out.println ((char) content + " ");

SOP ( content + " ");

SOP (" Remaining bytes : " + fin.available ())

SOP ( " Remaining bytes : " + fin.available ())

? ?

Output

A  
65

Remaining Bytes : 1

Remaining Bytes that can be read : 0

```
5 import java.io.*.FileInputStream;
import java.io.IOException;
public class FileEx2 {
    public static void main(String args) throws
        FileInputStream, IOException {
        byte[] bytes = new byte[20];
        Stream("Example.txt");
        int i;
        char c;
        i = fin.read(bytes);
        SOP("No of bytes read: " + i);
        SOP("Bytes read");
        for (byte b : bytes) {
            c = (char) b;
            SOP(c);
        }
    }
}
```

No of bytes read: 20  
Bytes read: Hello we are god.

~~Solve~~  
23/24