

OOJ Lab

To execute Java programs -

- 1) open a notepad file, write program & then save the file as filename.java
- 2) save the java program under all files category in ; `<c:\jdk-8\bin>`
- 3) go to Dosprompt. set the path where jdk-8 is installed `<c:\users\document>`
set path = "c:programfiles\Java\Jdk-8\bin"
- 4) `<c:\users\bmesel\documents> Java`
- 5) `<c:\users\bmesel\documents> Java Hello.java`

dir *.java

dir

Cmd prompt

① `cd\`

② `cd location of file (saved path)`
`&under.java`

③ `javac file name` ← same as main class name & save under all files.

④ `java main class name`

Quadratic Equation

```
import java.io.*;
import java.util.*;
class quadratic {
    public static void main (String args[])
    {
        Scanner s = new Scanner (System.in);
        System.out.println ("Enter a,b,c");
        double a = s.nextDouble();
        double b = s.nextDouble();
        double c = s.nextDouble();
        double d = b*b - 4*a*c;
        if (d>0)
        {
            double r1 = (-b + Math.sqrt(d))/ (2*a);
            double r2 = (-b - Math.sqrt(d))/ (2*a);
            System.out.println ("Roots are " + r1 + " " + r2);
        }
        else if (d==0)
        {
            double r = -b / (2*a);
            System.out.println ("Root is " + r);
        }
        else
            System.out.println ("Roots are Imaginary");
    }
}
```

O/P - Enter the values of a,b,c : 1 5 6
Roots are : -2 -3

X O/P: -2 -3
28/11/20

SGPA

```
import java.util.Scanner;  
class Student  
{
```

```
    String USN;  
    String name;  
    int [] credits = new int [20];  
    int [] marks = new int [20];  
    public void input (int n)  
{
```

```
    Scanner s = new  
    Scanner (System.in);  
    System.out.print("Enter student USN: ");  
    USN = s.nextLine();  
    System.out.print ("Enter student name: ");  
    name = s.nextLine();  
    for (int i=0; i<n; i++)
```

~~System.out.print ("Enter the subject "+(i+1)+"
marks and credits respectively! ");~~

~~marks[i] = s.nextInt();
credits[i] = s.nextInt();~~

```
{  
    public float calculate (int n)
```

```
{  
    int sum_of_credits = 0;  
    float result = 0.0f;  
    for (int i=0; i<n; i++)
```

{

```
sum - of - credits + = credits[i];
if (calculate-grade-point(marks[i]) == -1)
    return -1.0f;
else
    {
        result = result + (float) calculate-grade-
point(marks[i]) * credits[i];
    }
    return (result / sum - of - credits);
}

public int calculate-grade-point(int marks)
{
    if (marks >= 90)
        return 10;
    else if ((marks >= 80) && (marks < 90))
        return 9;
    else if ((marks >= 70) && (marks < 80))
        return 8;
    else if ((marks >= 60) && (marks < 70))
        return 7;
    else if ((marks >= 50) && (marks < 60))
        return 6;
    else if ((marks >= 40) && (marks < 50))
        return 5;
    return 1;
}
```

```
public void display (int n, float result)
```

{

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

```
    System.out.println (" Students Details");
```

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

```
    System.out.println (" Students USN :" + USN);
```

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

```
    System.out.println (" Students Marks and  
                        credits");
```

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

{

```
    System.out.println (" Subject " + i + " Marks :  
                        " + marks[i] + " credits : " + credits[i]);
```

{

```
    System.out.println ("SGPA : " + result);
```

{

public class ~~student~~ SGPA

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

{

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

```
    student s1 = new student ();
```

```
    System.out.print ("Enter the number of  
                      subjects : ");
```

```
    int n = s.nextInt();
```

```
    s1.input (n);
```

```
    float result = s1.calculate (n);
```

if (result == 1.0f)

3

System.out.println();

System.out.println("The student has failed in a subject. SGPA cannot be calculated!");

System.exit(0);

3

sl.display(n, result);

3

(*) Design

Y
21/12/2022
01pm

3

Design (desire) defining two methods

3

(*) Design (design) defining the interface

3

M02

- 50 days - 100 students

3

What needs to be done?

Book Details

```
import java.io.*;
import java.util.*;

class Book {
    String title; author;
    double price;
    int numPages;

    Book() {
        title = "Default";
        author = "Default";
        price = 0.0;
        numPages = 0;
    }

    void setTitle (String t) {
        title = t;
    }

    void setAuthor (String a) {
        author = a;
    }

    void setPrice (double p) {
        price = p;
    }

    void setPages (int np) {
        numPages = np;
    }

    public String toString () {
        return title + "\t" + author + "\t" + price + "\t" +
               numPages + "\n";
    }
}
```

```
class BookDetails {
    public static void main (String args[])
    {
        String t,a;
        double p;
        int np,n;
        Scanner s = new Scanner (System.in);
        System.out.println ("Enter the number of Books");
        n = s.nextInt();
        Book b[] = new Book[n];
        for (int i=0; i<n; i++)
        {
            System.out.println ("Enter the Title of the Books");
            t = s.next();
            System.out.println ("Enter the Author of the Books");
            a = s.next();
            System.out.println ("Enter the Price of the Books");
            p = s.nextDouble();
            System.out.println ("Enter number of pages of Book");
            np = s.nextInt();
            b[i] = new Book();
            b[i].setTitle (t);
            b[i].setAuthor (a);
            b[i].setPrice (p);
            b[i].setPages (np);
        }
    }
}
```

```
System.out.println("Title\t Author\t Price\t  
Price In");
```

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

```
System.out.println(b[i]);
```

3

3

✓
9/12/2021

Open

Abstract Demo

```
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 Abstract Demo

{

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);

}

{

~~System.out.println ("After setting dimensions,~~

~~area of rectangle is : 10.0~~

~~area of circle is : 78.5~~

~~area of triangle is : 5.0~~

OUTPUT-

area of rectangle is : 10.0

area of circle is : 78.5

area of triangle is : 5.0

Bank

```
import java.util.Scanner  
class account {  
  
    String customer-name  
    long acc-no;  
    float bal;  
  
    Scanner s = new  
    Scanner (System.in);  
    public void input()  
    {  
        System.out.println ("Enter customer name!");  
        customer-name = s.nextLine();  
        System.out.println ("Enter account number!");  
        acc-no = s.nextLong();  
        System.out.println ("Enter the starting amount");  
        (minimum.amount = 5000);  
        bal = s.nextFloat();  
        if (bal < 5000)  
    }  
  
    System.out.println ("Account balance can't be  
    less than 5000.0");  
    System.exit(0);  
}
```

3

```
public void display()
{
    System.out.println ("Customer Name: "
                        + customer-name);
    System.out.println ("Account Number: " + acc-no);
    System.out.println ("Amount: " + bal);
}

class Savings extends Account
{
    Scanner s = new Scanner (System.in);
    float deposit, withdraw, interest;
    public void deposit()
    {
        System.out.println ("Enter the amount to
                            be deposited:");
        deposit = s.nextFloat();
        bal += deposit;
        System.out.println ("Balance: " + bal);
    }

    public void withdraw()
    {
        System.out.println ("Enter the amount to be
                            withdrawn:");
        withdraw = s.nextFloat();
        if (bal < 5000)
            System.out.println ("Insufficient balance");
    }
}
```

else

3

bal -= withdraw;

```
System.out.println("Amount withdrawn: " + withdrawal);  
System.out.println("Balance: " + bal);
```

3

3

```
public void check_Bal()
```

Σ

If (Bal < 5000)

3

```
System.out.println ("Insufficient Balance\nBalance : "+ bal);
```

3

else

3

```
System.out.println ("Balance: " + bal);
```

3

3

public void interest ()

3

$$\text{interest} = (\text{bal} * 6) / 100;$$

bal + = interest

```
bal + = interest;  
System.out.println (" Interest Credited : " + interest  
" \n Balance : " + bal);
```

3

3

class Current extends Account

{

float deposit, withdraw, penalty;
public void deposit()

{

System.out.println("Enter amount to be deposited!");

deposit = s.nextFloat();

bal += deposit;

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

}

public void checkBal()

{

if (bal < 5000)

{

penalty = (0.1 * bal);

System.out.println("Initial Account Balance:
" + bal);

bal = bal - penalty;

System.out.println("Low Balance! \nPenalty
Amount: " + penalty + " \nAccount
balance: " + bal);

{

else

{

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

{

{

```
public boolean check_Bal_part_2()
```

```
{
```

```
    if (bal < 5000)
```

```
{
```

```
    penalty = (0.1 * bal);
```

```
System.out.println ("Initial Account Balance:" + bal);
```

```
bal = bal - penalty;
```

```
System.out.println ("Low Balance! In Penalty  
Amount :" + penalty + " In Account  
balance :" + bal);
```

```
return false;
```

```
{
```

```
return true;
```

```
{
```

```
public void withdraw ()
```

```
{
```

```
System.out.println ("Enter Amount to withdraw:");
```

```
withdraw = s.nextFloat();
```

```
if (check_Bal_part_2 ()) {
```

```
{
```

```
    bal -= withdraw;
```

```
System.out.println ("Amount withdrawn:
```

```
" + withdraw + " In Balance:" + bal);
```

```
{
```

```
{
```

```
public void chequebook()
```

```
{  
    System.out.println("cheque Book has been  
    issued!");}
```

```
}
```

```
public class Bank
```

```
{  
    public static void main (String [3] args)
```

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

```
    String ch;
```

```
    int n;
```

```
    Current c = new Current();
```

```
    Savings sa = new Savings();
```

~~System.out.println("Enter the account
type (s for savings, c for current):");
ch = s.next();~~

```
switch (ch.toLower Case()) {
```

```
{
```

```
    case "s": sa.input();
```

```
        do
```

```
{
```

~~System.out.println("1. Deposit\n2. withdrawal
3. check balance\n4. check interest\n5. check details\n6. Exit transaction
nEnter your choice:");
n = s.nextInt();~~

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

switch (n)

{

case 1: sa.deposit();
break;

case 2: sa.withdraw();
break;

case 3: sa.checkBal();
break;

case 4: sa.interest();
break;

case 5: sa.display();
break;

case 6: System.out.println("Existing
Transaction!");
System.exit(0);
break;

default: System.out.println("Invalid
operation");

}

} while(true);

case "c": c.input();
do {

System.out.print("1. Deposit\n2. withdrawal
3. check balance\n4. Issue Cheque Book"
"5. Show Account Details\n6. Exit
Transaction\nEnter your choice: ");
n = s.nextInt();

switch (n)

{

case 1: c. deposit ();
break;

case 2: c. withdrawal ();
break;

case 3: c. check_Bal ();
break;

case 4: c. interest ();
break;

case 5: c. display ();
break;

case 6: System.out.println ("Exiting
transaction!");

System.exit (0);

break;

default: System.out.println
("Invalid operation");

while (true);

default: System.out.println ("Invalid
choice");

break;

}

}

}

Output:

Enter account type: S

Enter customer name: Pr

Enter account number: 123

Enter the starting amount (min. amt=500) : 7000

~~Pulseen~~
01 Pulseen
30/12/22

8/12/2022

```
import java.util.Scanner;
import java.util.Arrays;
public class Anchor
{
    public static void main (String [] args)
    {
        Scanner s = new Scanner (System.in);
        int choice;
        System.out.println ("1. Under Graduate Student\n"
                            "2. Graduate Student\n"
                            "Enter your choice? ");
        choice = s.nextInt();
        switch (choice)
        {
            case 1:
                System.out.println ("Enter the student name:");
                UnderGraduate u = new UnderGraduate (s.next());
                System.out.println ("Enter subject number &\n"
                                    "marks of 4 subjects");
                for (int i=0; i<4; i++)
                {
                    u.setTestScore (s.nextInt(), s.nextInt());
                }
            break;
        }
    }
}
```

```
    u. set Test Result();  
    u. display();  
}  
break;
```

case 2:

```
{
```

```
System.out.println("Enter new student name:");  
Graduate g = new Graduate(s.nextInt());  
System.out.println("Enter subject number &  
marks of 4 subjects");  
for (int i=0; i<4; i++) {  
    System.out.println("Enter mark " + i + ":");  
    g.setTestScore(s.nextInt(), s.nextInt());  
}  
g.setTestResult();  
g.display();  
}  
break;
```

default : "System.out.println("Invalid choice!")";

interface A

```
{  
    public String getName();  
    public void setTestScore (int no, int marks);  
    public int [] getTestScore();  
    public void setTestResult();  
    public int getTestResult();  
}
```

```
public void display();
```

```
{
```

```
abstract class Student implements A
```

```
{
```

```
String name;
```

```
int [ ] test = new int[4];
```

```
int sum;
```

```
abstract public void generateResult();
```

```
Student()
```

```
{
```

```
student( String name)
```

```
{
```

```
this.name = name;
```

```
{
```

```
public String getName()
```

```
{
```

```
return this.name;
```

```
{
```

```
public void setTestScore( int no, int marks)
```

```
{
```

```
test [no] = marks;
```

```
{
```

```
public int getTestScore()
```

```
{
```

```
return test;
```

```
{
```

```
public void setTestResult()
```

```
{
```

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

```
{
```

```
sum = sum + test[i];
```

```
}
```

```
sum /= 4
```

```
{
```

```
public int getTestResult()
```

```
{
```

```
return sum;
```

```
{
```

```
public void display()
```

```
{
```

```
System.out.println("Student Name: " + getName());
```

```
System.out.println("Student Marks: " +
```

```
+ arrays.toString(getTestScore()));
```

```
System.out.println("Result: ");
```

```
generateResult();
```

```
{
```

```
{
```

```
class UnderGraduate extends Student
```

```
{
```

```
UnderGraduate()
```

```
{}
```

```
UnderGraduate(String name)
```

```
{
```

```
this.name = name;
```

```
{
```

```
public void generateResult()
```

```
{
```

```
if (get TestResult () >= 60)
    system.out.println("Pass");
else
    system.out.println("Fail");
```

{

```
class graduate extends Student
```

{

```
(graduate)
```

{}

```
Graduate (String name)
```

{

```
this.name = name;
```

{

```
public void generateResult()
```

{

```
if (get TestResult () >= 70)
```

```
    system.out.println ("Pass");
```

```
else
```

```
    system.out.println ("Fail");
```

{

{}

OUTPUT

✓ 1) Under Graduate Student

✓ 2) Graduate Student

Enter your choice: 1

Enter student name: P

Enter subject number & marks of 4 subjects

1 33 3 55
2 44 4 66

student name: P

student marks: [33, 44, 55, 66]

result: fail

6/1/2023

```
import java.util.Scanner;
class WrongAge extends Exception
{
    public String getMessage()
    {
        return "Age can't be negative";
    }
}
class InvalidAge extends Exception
{
    public String getMessage()
    {
        return "Son's age can't be greater than father's!";
    }
}
class Father
{
    Scanner s = new Scanner(System.in);
    int f;
    Father() throws WrongAge
    {
        System.out.println("Enter father's Age:");
        f = s.nextInt();
        try
        {
            if (f < 0)
                throw new WrongAge();
        }
        catch (WrongAge e)
        {
            System.out.println(e.getMessage());
        }
    }
}
```

catch (WrongAge e1)

{

```
    System.out.println (e1.getMessage());  
    System.exit (0);
```

{

}

}

class son extends Father

{

```
    int son';
```

son () throws WrongAge, InvalidAge

{

```
    super();
```

```
    System.out.print ("Enter the son's age:");
```

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

try

{

```
    if (son < 0)
```

```
        throw new WrongAge();
```

}

catch (WrongAge e2)

{

```
    System.out.println (e2.getMessage());
```

{

try (son > j)

```
    throw new InvalidAge();
```

{

catch (InvalidAge e3)

{

```
    System.out.println (e3.getMessage());
```

```
    System.exit (0);
```

```
    } system.out.println("Ages are appropriate");  
}  
public class Try-1  
{  
    public static void main (String [] args)  
        throws wrongAge, InvalidAge  
    {  
        new Son();  
    }
```

output

Enter Father's Age: 40
Enter Son's Age: 60
Son's Age can't be greater than Father's!

Enter Father's Age: 40

Enter Son's Age: -16

Age can't be negative

Enter Father's Age: 48

Enter Son's Age: 18

Ages are appropriate.

✓ 1/1/23
B. B. P. S.
B. B. P. S.

13/11/2023

class Thread-1 extends Thread

{

public void run()

{

int i=0;

while (i<100)

{

try

{

Thread.sleep(10000);

System.out.println("BMSCE");

}

catch (Exception e)

{

System.out.println("Exception: "+e);

}

i++;

}

}

class Thread-2 extends Thread

{

public void run()

{

int i=0;

while (i<100)

{

try

{

Thread.sleep(2000);

System.out.println("CSE");

{

catch (Exception e)

{

```
System.out.println ("Exception"+e);
```

3

10

itt

3

3

public class Try

3

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

3

```
Thread t1 = new Thread_1();
```

Thread t2 = new Thread(2);

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

t2.start()

3

(3) *the following 3 numbers*

OUTPUT

CSE
CSE
CSE
CSE
~~BMSCE~~
CSE
CSE
CSE
CSE
CSE
CSE
BMSCE