

11/12/23

classmate

Date \_\_\_\_\_

Page \_\_\_\_\_

### 1. Point

- class Example {

public static void main (String args []) {

System.out.print ("Hello") ;

}

}

O/P: "Hello"

(1) 2020 print () will be static method

### 2. Calculate sum, diff (+ product), quotient

- class calc {

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

int a = 5;

int b = 1;

int sum = a + b;

int diff = a - b;

int product = a \* b;

int quotient = a / b;

System.out.print ("Sum: " + sum + " DIFF = " + diff + " Product = " + product + " Quotient = " + quotient);

}

}

O/P: Sum = 6 DIFF = 4 Product = 5 Quotient = 5

### 3. Check prime: (non prime) will be static method

class Prime {

public static void main (String args []) {

int a = 6; } (1 = prime 2 = prime 3 = prime)

if .. (a % 2 == 0) { "Not prime" ; }

System.out.print ("Not prime"); }

else { }

System.out.print ("Prime"); } (8 : prime)

O/P: Not Prime

#### 4. Fibonacci Series

\* class Fib {

public static void main (String args [] ) {

```
int n1 = 0, n2 = 1, n3, i; count = 10;
System.out.print (n1 + " " + n2);
for (i = 2; i < count; i++) {
    n3 = n1 + n2;
    System.out.print (" " + n3);
    n1 = n2;
    n2 = n3;
}
```

}

}

= fib() O P: 0 + 1 + 2 + 3 + 5 + 8 + 13 + 21 + 34 + 55

#### 3. Prime

\* class Prime {

public static void main (String args []) {

int i, m = 0, flag = 0;

int n = 3;

m = (n / a) \* prime(); linear time complexity

if (n == 0 || n == 1) {

System.out.println (n + " is not prime no. " );

}

else {

for (i = 2; i <= m; i++) { flag = 0;

If  $(n \neq 0) \{$

```
System.out.println(n + " is not prime no.");
```

$$\text{Flag} = \text{LJ}$$

bæk j)

3

2021-06-19 21:56:31b 31a5b

if (Flag == 0)

f

```
System.out.println(n + " is prime no.") ;
```

3

1

3

1

( $\rho$  3) such) flagship

O/P : 3 is prime no.

(~~33dub~~ (~~33dub~~ p 33dub) posse

$i \cdot p = 126$

id: 19849

$$D = \sup_{\theta} \varphi$$

(p. 1000sp.) 100200sp

Chap. 10

2009-07-20 12:10

$$\text{input} \cdot p = \text{output}$$

( ) 2003-2004 bjp

18/12/23

CLASSMATE

Date \_\_\_\_\_

Page \_\_\_\_\_

## Grocery

import java.util.Scanner;

class grocery

{

double dal = pulses = sugars;

grocery ()

{

dal = 1; + 0.5 \* 1000;

pulses = 1;

sugars = 0.5;

}

grocery (double a)

{

dal = pulses = sugars = a;

{

grocery ( double a, double b, double c )

{

dal = a;

pulses = b;

sugars = c;

{

grocery ( grocery g )

{

dal = g.dal;

pulses = g.pulses;

sugars = g.sugars;

{

void amtCalc()

{

System.out.println ("Total = " + (dal\*50 + pulses\*50 + sugars\*60));

{ }

class main

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

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

grocery gl = new grocery();  
gl.amt-calc();

System.out.println("Enter the value");

double q = sc.nextDouble();

grocery gd = new grocery(10);  
gd.amt-calc();

grocery g3 = new grocery(10, 30);

g3.amt-calc();

grocery g4 = new grocery(10, 30);

g4.amt-calc();

}

(Output part 2.00)

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

int n1 = sc.nextInt() + " " + "n1";

int n2 = sc.nextInt() + " " + "n2";

int n3 = sc.nextInt() + " " + "n3";

int n4 = sc.nextInt() + " " + "n4";

(Output part 2.00 + 0.00 + 0.00 + 0.00 = 0.00)

main 0.0

(Epson font) main 0.00 0.00

8/01/24

Date  
Page

## Book - database

Page No.

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

```
class books {
```

```
{
```

```
String name;
```

```
String author;
```

```
int price;
```

```
int numpages;
```

```
books() {
```

```
books(String name, String author, int price, int numpage,
```

```
{
```

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

```
this.price = price; this.numpages = numpages;
```

```
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 = "number 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;
```

```
String name;
```

```
String author;
```

```
int price;
```

```
int npages;
```

```
System.out.print("enter the number of books : ");
```

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

```
books b[];
```

```
b = new books[n];
```

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

```
{
```

```
System.out.println("book" + (i + 1) + ":");
```

```
System.out.print(" enter 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: ");
```

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

```
b[i] = new books(name, author, price, npages);
```

```
}
```

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

```
System.out.print("book" + (i + 1) + ":" + n + " " + b[i]);
```

```
}
```

```
?
```

O/P:

enter the number of books : 2

book1 : enter name of book : ABC

enter author : XYZ

enter price: 1200

enter number of pages: 300

book2: enter name of book: WASD

enter author: ab

enter price: 700

enter number of pages: 230

(i) d book

(ii) book wasd

(iii) ab 700 230

(iv) 1200 + (1 + 5%) \* 1200 = 1260 (Ans)

(v) 700 + (1 + 5%) \* 700 = 735 (Ans)

(vi) 1260 - 735 = 525 (Ans)

(vii) 1260 / 1200 = 1.05 (Ans)

(viii) 735 / 700 = 1.05 (Ans)

(ix) 1260 - 735 = 525 (Ans)

(x) 1260 / 1200 = 1.05 (Ans)

(xi) 735 / 700 = 1.05 (Ans)

(xii) 1260 - 735 = 525 (Ans)

(xiii) 1260 / 1200 = 1.05 (Ans)

(xiv) 735 / 700 = 1.05 (Ans)

## Quadratic Equation

(0 < b) if 38/9

```
import java.util.Scanner; if (d == 0) {
```

```
class Quadratic { if (d != 0) {
```

```
    int a, b, c;
```

```
    double r1, r2, d;
```

```
    void getd()
```

```
{ if (a == 0) { System.out.println("Not a quadratic equation"); return; }
```

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

```
System.out.print("Enter the coefficients of ax^2 + bx + c: ");
```

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

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

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

```
void compute()
```

```
while (a == 0)
```

```
{ System.out.println("Not a quadratic equation"); return; }
```

```
System.out.print("Entered a non zero value for a: ");
```

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

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

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

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

```
r1 = (-b) / (2 * a);
```

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

```
System.out.print("Root1 = Root2 = " + r1);
```

```
else if (d > 0) {
```

128808, 51

else if ( $d > 0$ )

{

$$r1 = ((-b) + (\text{Math.sqrt}(d))) / (\text{double})(a+e);$$

$$r2 = ((-b) - (\text{Math.sqrt}(d))) / (\text{double})(a+e);$$

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

System.out.println("Root1 = " + r1 + " Root2 = " + r2);

}

else if ( $d < 0$ )

{

System.out.println("Roots are Imaginary");

$$r1 = (-b) / (a+e);$$

$$r2 = \text{Math.sqrt}(-d) / (a+e);$$

System.out.println("Root1 = " + r1 + " + i " + r2);

System.out.println("Root2 = " + r1 + " - i " + r2);

}

} (j) 10.2.3.2

}

(2) imaginary roots

class QuadraticMain

{

public static void main (String arg[])

{ Roots are imaginary if  $d < 0$

(3) Roots are real & equal if  $d = 0$

Quadratic q = new Quadratic();

q.getd();

q.compute();

}

$i^2 + b^2 - d^2 = 0$

$(a+b)^2 = 0$

}

}

O/P :

i[0][0] = 0

(3) Enter the coefficients of quadratic eqn. 5 2

Roots are real and distinct

Root1 = -0.4384471871911647 and Root2 = -4.5615520

12480883 |

**Student**

import java.util.\*;

class Student {

String usn; // student id or usn

String name; // student name

int marks[] = new int[6];

float percentage = 0;

void acceptData(int i) {

Scanner sc = new Scanner(System.in);

System.out.println("Enter student - " + i + " name");

name = sc.nextLine();

System.out.println("Enter student - " + i + " usn");

usn = sc.nextLine();

System.out.println("Enter student - " + i + " marks");

for (int j = 0; j < 6; j++) {

System.out.println("marks of subject " + j + ": ");

marks[j] = sc.nextInt();

percentage += marks[j];

}

S.P.

}

: 6 subjects to print

void calculatePercentage(int i) {

percentage = (percentage / 6);

System.out.println("Percentage of student - " + i + " = ");

percentage += (10 / 6);

}

S.P.

}

: 2 subjects to print

S.P.

public static void main(String args[]) {

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

Scanner sc = new Scanner(System.in);

int n = sc.nextInt();



21/1/24

classmate

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

Package A

(Student.java) date:

import java.util.\*; import

java.util.List; import

Package CIE;

private Student student;

public class Student {

String name;

String USN;

int sem; } (class print name, USN, student object)

(constructor) creates object of the class

public Student (String name, String USN, int sem)

{

this.name = name; this.USN = USN;

this.sem = sem; } (constructor) creates object of the class

this.name = name; this.USN = USN;

this.sem = sem; } (constructor) creates object of the class

}

}

package CIE;

public class Internals { extends (CIE.Student); }

{ (constructor) creates object of the class

public int Internals[] { } (constructor) creates object of the class

public Internals (String name, String USN, int sem, int Internals)

{ super (name, USN, sem); }

this.Internals[] = Internals[]; }

{ }

{ }

package SEE;

import CIE.Student;

public class Externals extends (CIE.Student); }

{ public int Externals[] { } (constructor) creates object of the class

public Externals (String name, String USN, int sem, int Externals)

{ super (name, USN, sem); }

this.Externals[] = Externals[]; }

{ }

{ }

```

import CIE.Student;
import CIE.Internal;
import STE.External;
import java.util.Scanner;
}

public class FinalMarks {
    public static void main(String args[]) {
        Scanner s1 = new Scanner(System.in);
        System.out.println("Enter no. of students");
        int n = s1.nextInt();
        String name[] = new String[n];
        String un[] = new String[n];
        int sem[] = new int[n];
        int internalMarks[][] = new int[n][5];
        int externalMarks[][] = new int[n][5];
        for (int i = 0; i < n; i++) {
            System.out.println("Enter details of student " + (i + 1));
            System.out.print("Name: ");
            name[i] = s1.nextLine();
            System.out.print("Un: ");
            un[i] = s1.nextLine();
            System.out.print("Semester: ");
            sem[i] = s1.nextInt();
            System.out.println("Enter internal marks for 5 courses");
            for (int j = 0; j < 5; j++) {
                System.out.print("In courses " + (j + 1) + ": ");
                internalMarks[i][j] = s1.nextInt();
            }
        }
    }
}

```

```
System.out.println("Enter SEE marks for 5 courses");
for (int j=0; j<5; j++) {
    System.out.print("In course " + (j+1) + ": ");
    Seemarks[i][j] = sc.nextInt();
}
}
```

```
System.out.println("In course " + (j+1) + ": ");
Seemarks[i][j] = sc.nextInt();
}
```

```
int Finalmarks[5] = new int[n][5];
for (int i=0; i<n; i++) {
    System.out.println("Final marks for " + name[i] + " is " + Finalmarks[i]);
}
}
```

```
InternalMarks[5] = new InternalMarks(name[1], USN[1], sem[1]),
    System.out.println("Internal marks for " + name[1] + " is " + InternalMarks[1]);
}
```

```
ExternalMarks[5] = new ExternalMarks(name[1], USN[1], sem[1], Seemarks[1]),
    System.out.println("External marks for " + name[1] + " is " + ExternalMarks[1]);
}
```

```
System.out.println("Final marks for " + name[1] + " student in 5 courses");
for (int k=0; k<n; k++) {
    System.out.println(" " + Finalmarks[k]);
}
}
```

```
System.out.println("Name[" + k + "] : ");
```

```
System.out.println("CIE marks: " + CIEmarks);
    System.out.println(" " + CIEmarks);
    System.out.println(" " + CIEmarks);
    System.out.println(" " + CIEmarks);
    System.out.println(" " + CIEmarks);
}
```

```
for (int j=0; j<5; j++) {
    System.out.println("course " + (j+1) + " : " + InternalMarks[j]);
}
}
```

```
System.out.println("SEE marks: ");
    System.out.println(" " + SEEmarks);
    System.out.println(" " + SEEmarks);
    System.out.println(" " + SEEmarks);
    System.out.println(" " + SEEmarks);
    System.out.println(" " + SEEmarks);
}
```

```
for (int j=0; j<5; j++) {
    System.out.println("course " + (j+1) + " : " + Seemarks[1][j]);
}
}
```

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

```
sc.close();
```

```
}
```

```
}
```

```
}
```

~~STP :  $x = 10, 20, 30, 40, 50$  and  $y = 1, 2, 3, 4, 5$~~

$x : 10 \ 20 \ 30 \ 40 \ 50$  and  $y : 1 \ 2 \ 3 \ 4 \ 5$

~~STP :  $x = 10, 20, 30, 40, 50$  and  $y = 1, 2, 3, 4, 5$~~

$x : 10 \ 20 \ 30 \ 40 \ 50$  and  $y : 1 \ 2 \ 3 \ 4 \ 5$

~~OP : Definition of STP~~

Final marks for 5 students in 5 courses.

~~$x_1, x_2, x_3, x_4, x_5$  (CIE marks) and  $y_1, y_2, y_3, y_4, y_5$  (SEE marks)~~

~~CIE marks~~      ~~SEE marks~~

~~Course 1 : 10      Course 2 : 20~~

~~2 : 20~~

~~2 : 2~~

~~3 : 30~~

~~4 : 40~~

~~5 : 50~~

~~3 : 3~~

~~5 : 5~~

~~SEE marks~~      ~~SEE marks~~

~~Course 1 : 10      Course 2 : 20~~

~~3 : 30~~

~~3 : 3~~

~~4 : 40~~

~~5 : 50~~

~~4 : 4~~

~~5 : 5~~

~~Q1 W1 M1~~

~~(1, 2, 3)~~

1/24

1.

class WrongAgeException extends Exception {

    WrongAgeException (String message) {

        super (message);

}

}

class Father {

    private int age;

    public Father (int age) throws WrongAgeException {

        if (age < 0) {

            throw new WrongAgeException ("age cannot be -ve");

}

        this.age = age;

    }

    return age;

}

class Son extends Father {

    private int sonAge;

    public Son (int fatherAge) throws WrongAgeException {

        throws wrong AgeException {

            super (fatherAge);

        if (sonAge >= fatherAge) {

            throw new WrongAgeException ("Son's age should be less than  
                father");

        this.sonAge = sonAge;

    }

```
public class InheritanceExceptionDemo {
```

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

```
        try {
```

```
            Father father = new Father(40);
```

```
            System.out.println("Father's age: " + father.getAge());
```

```
Son son = new Son(20);
```

```
        System.out.println("Son's age: " + son.getSonAge());
```

```
    } catch (WrongAgeException e) {
```

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

```
}
```

```
}
```

Output:

Father age: 40

Son's age: 20

Output shows no exception

instead it is giving

(exception to wrongage age) no? sir?

20

```
class DisplayThread extends Thread {
```

```
    private String message;
```

```
    private int intervalMillis;
```

```
    public DisplayThread (String message, int intervalMillis) {
```

```
        this.message = message;
```

```
        this.intervalMillis = intervalMillis;
```

```
}
```

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

```
        while (true) {
```

```
            try {
```

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

```
                Thread.sleep(intervalMillis);
```

```
}
```

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

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

```
}
```

```
}
```

```
}
```

```
public class Main {
```

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

```
        DisplayThread thread1 = new DisplayThread ("BMSCE", 1000);
```

```
        DisplayThread thread2 = new DisplayThread ("CS", 2000);
```

```
        thread1.start();
```

```
        thread2.start();
```

```
}
```

```
}
```

OPP:

BMSCE

CSE

3 basic human building blocks

CSE

opposite photo shapes

CSE

University of Delhi

CSE

BMSCE

CSE

3 types - application

SS &  
DBMS

application & utility based

3 basic building blocks

functionality

3 C's

if suddenly stop working

all other applications will break

3 components required

interrelated

Java

1/12

1. Creating Label, button and textField in a frame using AWT.

```
import java.awt.*;           // for frame, label, button
import java.awt.event.*;      // for windowListener
```

public class AWTExample extends WindowAdapter {

Frame f;

AWTExample() {

frame f = new Frame();

f.addWindowListener(this);

Label l = new Label("Employee id");

Button b = new Button("Submit");

TextField t = new TextField();

(This line) means we add message

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 aut = new AWTExample();

}

Q. Create a button and add a action listeners for mouse click.

```
import javax.swing.*; // import swing package
import java.awt.event.*; // import event package
public class EventHandling extends WindowAdapter implements ActionListener {
```

Frame f;

TextField tf;

EventHandling() {

F = new Frame();

F.addwindowListener(this);

tF = new JTextField();

tF.setBounds(60, 50, 120, 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 e) {

tf.setText("Welcome");

public void actionPerformed(ActionEvent e) {

System.exit(0);

}

public static void main(String args[])

{

new EventHandling();

}

}

it is very useful

to implement this

O/P: (W3C) and now click on it

increased and it will

Employee ID	welcome	Employee Name
		John Smith
	(Click)	Job = Sales Manager

is (cong) = 100 ms

and it is different when we click on the button

O/P: (No +100 ms increase) 100 ms

i)

Employee Info

Employee ID

Submit

on submit

(final)

but when I click on submit to continue then

100 ms

100 ms

100 ms

Programs on I/O: reading file input

Ex 1

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

```
ByteInputStream b = new ByteArrayInputStream(buf);
int k = 0;
while ((k = b.read()) != -1) {
    System.out.println("ASCII value character is " + k);
    System.out.println("Special character is " + (char) k);
}
```

char ch = (char) k;

System.out.println("ASCII value character is " + k);  
Special character is " + ch);

}

Output:

13 14 15 16

Employee info

Employee ID

Submit

ASCII values of character is : 35 ; special character is

" 36 ; " "

" 37 ; " "

" 38 ; " "

## Ex - 3

```
public class FileEx {
```

```
    public static void main (String args[]) 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.print ((char) content + " ")

System.out.println (content);

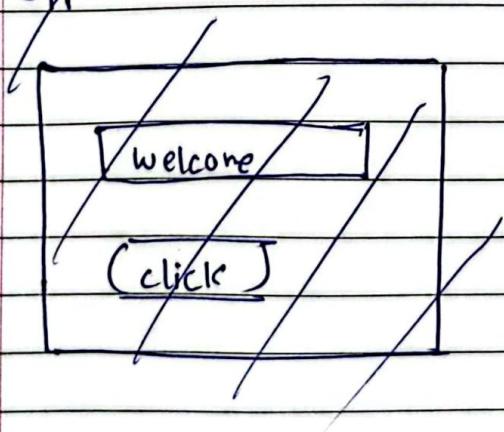
System.out.println ("Remaining bytes that can be read : " +

fin.available());

"File Example has been read" is being displayed.

System.out.println ("Remaining bytes that can be read : "  
 fin.available());

O/P:



O/P:

Remaining bytes that can read: 5

H 7a e101 L 108 0111

Remaining byte that can read: 0.

Ex-4

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

```
public class FileEx2 {
```

```
    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("no.of bytes read" + i);  
        System.out.print("Byte read");  
    }
```

```
for (byte b : bytes) {
```

```
    c = (char) b;
```

```
    System.out.print(c);
```

```
}
```

```
}
```

O/P:

Number of byte read: 5

Bytes read: HELLO.

Shri 02.06.  
26.02.24