

VISVESVARAYA TECHNOLOGICAL UNIVERSITY

“JnanaSangama”, Belgaum -590014, Karnataka.



LAB REPORT

on

JAVA LAB (21CS3PCOOJ)

Submitted by

Archit Mehrotra (1BM21CS031)

in partial fulfillment for the award of the degree of
BACHELOR OF ENGINEERING
in
COMPUTER SCIENCE AND ENGINEERING



B.M.S. COLLEGE OF ENGINEERING

(Autonomous Institution under VTU)

BENGALURU-560019

October-2022 to Feb-2023

**B. M. S. College of Engineering,
Bull Temple Road, Bangalore 560019**
(Affiliated To Visvesvaraya Technological University, Belgaum)
Department of Computer Science and Engineering



CERTIFICATE

This is to certify that the Lab work entitled “Java Lab 21CS3PCOOJ” carried out by **Archit Mehrotra (1BM21CS031)**, who is bonafide student of **B. M. S. College of Engineering**. It is in partial fulfillment for the award of **Bachelor of Engineering in Computer Science and Engineering** of the Visvesvaraya Technological University, Belgaum during the year 2022. The Lab report has been approved as it satisfies the academic requirements in respect of a Database Management Systems (22CS3PCDBM) work prescribed for the said degree.

Name of the Lab-Incharge
Designation
Department of CSE
BMSCE, Bengaluru

Dr. Jyothi S Nayak
Professor and Head
Department of CSE
BMSCE, Bengaluru

Index

Sl. No.	Experiment Title	Page No.
1	LAB 1	4
2	LAB 2	8
3	LAB 3	13
4	LAB 4	22
5	LAB 5	26
6	LAB 7	42
7	LAB 8	50
8	LAB 9	

LAB 1

Question

Develop a Java program that prints all real solutions to the quadratic equation $ax^2+bx+c = 0$. Read in a, b, c and use the quadratic formula. If the discriminant $b^2 - 4ac$ is negative, display a message stating that there are no real solutions.

Code

The image shows handwritten code on lined paper. At the top, it says "Java Weeks". Below that, the code starts with a comment: "Q1) WAP to write a Java program to print all real solution to a quadratic equation". The code then begins with the import statement: "import java.util.Scanner;". It defines a class named "Equation" with a main method: "public static void main (String args[])". Inside the main method, it prints a message: "Enter the coefficients a, b, c:". It then creates a Scanner object "sc" and reads three integers from the user: "double a = sc.nextInt();", "double b = sc.nextInt();", and "double c = sc.nextInt();". It checks if "a" is zero: "if (a==0) {". If "a" is zero, it prints "a cannot be zero!!" and exits. Otherwise, it calculates the discriminant: "double D = b*b - 4*a*c;". Finally, it creates an instance of the "EquationClass" class: "EquationClass e = new EquationClass();".

```
Q1) WAP to write a Java program to print all real solution to a quadratic equation
import java.util.Scanner;
class Equation {
    public static void main (String args[])
    {
        Scanner sc = new Scanner(System.in);
        double a = sc.nextInt();
        double b = sc.nextInt();
        double c = sc.nextInt();
        if (a==0) {
            System.out.println ("a cannot be zero!!");
        }
        else {
            double D = b*b - 4*a*c;
            EquationClass e = new EquationClass();
        }
    }
}
```

if (2 > 0)

System.out.println ("There are no
real solutions");

} else if (2 == 0)

{

System.out.println ("One solution
(unique) exists");

realRoot = -b / a;

ob.check (a, b, c);

ob.display ();

}

System.out.println ("Two solutions
exist (real and distinct)!");

ob.check (a, b, c);

ob.display ();

}

5

5

5

5

5

5

5

5

5

5

5

5

5

5

5

5

5

5

5

5

5

5

5

5

5

5

5

5

5

5

5

5

5

5

5

5

5

5

5

5

5

5

5

5

5

5

5

5

5

5

5

5

5

5

5

5

5

5

5

5

5

5

5

5

5

5

5

5

5

5

5

5

5

5

5

5

5

5

5

5

5

5

5

5

5

5

5

5

5

5

5

5

5

5

5

5

5

5

5

5

5

5

5

5

5

5

5

5

5

5

5

5

5

5

5

5

5

5

5

5

5

5

5

5

5

5

5

5

5

5

5

5

5

5

5

5

5

5

5

5

5

5

5

5

5

5

5

5

5

5

5

5

5

5

5

5

5

5

5

5

5

5

5

5

5

5

5

5

5

5

5

5

5

5

5

5

5

5

5

5

5

5

5

5

5

5

5

5

5

5

5

5

5

5

5

5

5

5

5

5

5

5

5

5

5

5

5

5

5

5

5

5

5

5

5

5

5

5

5

5

5

5

5

5

5

5

5

5

5

5

5

5

5

5

5

5

5

5

5

5

5

5

5

5

5

5

5

5

5

5

5

5

5

5

5

5

5

5

5

5

5

5

5

5

5

5

5

5

5

5

5

5

5

5

5

5

5

5

5

5

5

5

5

5

5

5

5

5

5

5

5

5

5

5

5

5

5

5

5

5

5

5

5

Date _____
Page _____

void check (double a, double b, double c)
{
 this.a = a;
 this.b = b;
 this.c = c;
 double d = Math.pow(b * b - 4 * a * c,
 0.5);
 x1 = (-b - d) / (2 * a);
 x2 = (-b + d) / (2 * a);
}

void display()
{
 System.out.println(x1);
 System.out.println(x2);
}

~~Off~~

ex: Enter the coefficient a,b,c:
0 1 2
a cannot be zero

~~Ex: Enter the coefficient a,b,c:
-3
1~~

Solutions are real and distinct!
2.10688572
-0.36885721

Output

```
C:\Users\archi\OneDrive\Desktop>java Equation.java
enter the coefficients a,b,c:
5
9
4
Solutions are real and distinct!
-1.0
-0.8

C:\Users\archi\OneDrive\Desktop>
```

LAB 2

question

Develop a Java program to create a class Student with members usn, name, an array credits and an array marks. Include methods to accept and display details and a method to calculate SGPA of a student.

code

Week 2

Date _____
Page _____

- Q2) PWT to create a student with
members usn, name, an array ~~marks~~
marks. & include methods to accept
and display

```
import java.util.Scanner;  
  
class Student  
{  
    String usn;  
    String name;  
}
```

```
int credits[] = new int[5];  
int marks[] = new int[5];
```

```
void entrInfo()
```

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

~~```
System.out.println("Enter the usn");
this.usn = sc.nextLine();
```~~~~```
System.out.println("Enter the  
Name : ");  
this.name = sc.nextLine();
```~~

```
System.out.println("Enter the  
credits : ");
```

for (int i=0; i<9; i++)

{

this.credit[i] = sc.nextInt();

System.out.println("Enter your marks:");

for (int i=0; i<9; i++)

{

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

}

void displayInfo()

{

System.out.println("In Below o The
Student Information :- ");

System.out.println("USN: " + this.usn);

System.out.println("NAME: " + this.name);

System.out.println("CREDITS: ");

for (int i=0; i<9; i++)

{

System.out.println(this.marks[i] + " ");

}

float calculate SGPA ()

{
float sgpa;
float totalcredits = 0;

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

{
totalcredits += this.credit[i];

float gp = 0;

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

this.marks[i] * (this.tu
marks[i]) / 100 + f1);

}
sgpa = gp / totalcredits;

return sgpa;

public class Calc

{
public static void main (String args[])

student s1 = new student();
s1.enrollInfo();
s1.displayInfo();

Output

```
C:\Users\archi\OneDrive\Desktop>java calc
enter the USN
1BM21CS031
Enter the Name:
Archit
Enter the credits:
3 4 1 3 1 3 1
enter your marks:
50 62 86 68 62 75 56 78 79
Below is the Student Information:-
USN: 1BM21CS031
NAME: Archit
CREDITS:
50
62
86
68
62
75
56
78
79
SGPA: 7.25
C:\Users\archi\OneDrive\Desktop>
```

LAB 3

Question

Create a class Book which contains four members: name, author, price, num_pages. Include a constructor to set the values for the members. Include methods to set and get the details of the objects. Include a `toString()` method that could display the complete details of the book. Develop a Java program to create n book objects.

Code

Wppn 3

Date / /
Page / /

- Q) Create a class Book which contains 4 members: name, author, price, num-pages. Include constructor, ToString() method. Include a ToString() method. Create n book objects.

```
import java.util.Scanner;  
  
class Book  
{  
    String author;  
    String name;  
    int price;  
    int numPages;  
  
    book(String name, String author, int  
          price, int pages)  
    {  
        name = name;  
        author = author;  
        price = price;  
        numPages = pages;  
    }  
}
```

book()

{

}

to

void SetDetails()

{

Date / /
Page / /

```
float sgpa = sl.calculateSGPA();
```

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

off

enter the usn

1BM21CS071

enter the Name:

Archit

Enter the credits:

3 9 1 31 3131

enter your marks

50 62 86 68 62 75 30 78 79

B Below is the student Information:-

USN :- 1BM21CS071

NAME: Archit

CREDITS:

50

6 L

86

68

62

75

56

78

29

SGPA: 7.25

20/01/2023

Scanner sc = new Scanner (System.in);
System.out.print ("entr name
of book ");
name = sc.nextLine();

System.out.print ("entr author's name : ");
author = sc.nextLine();
System.out.print ("entr price of a book : ");
price = sc.nextInt();
System.out.print ("entr num of
pages : ");
pages = sc.nextInt();

System.out.print ("Penter num of
Pages : ");
num_pages = sc.nextInt();
System.out.println (" DETAILS SET
SUCCESSFULLY :)");

void getDetails () {
System.out.println ("NAME : " +
name);
System.out.println ("AUTHOR : "
+ author);
System.out.println ("PRICE : " +
price);

PAGE NO.: ..

Date _____
Page _____

System.out.println() is set using
constructor In2: Set using method "j";

int ch = sc.nextInt();

if (ch == 2)

{ b1[i] = new book(1);

b1[i].setDetails(2);

}

else

{

b1[i] = new
book ("FINKLE", "ANONYMOUS", 20,

100),

}

for (int j = 0; j < n; j++)

{

System.out.println("PRINTING
Book DETAILS");

new (System.in),

System.out.println("PRINTING
Book DETAILS");

book[i],

System.out.println("It displays

; j++)

using method In2: display
using 6 strings ");

+ ln ("Enter Details")); int ch = sc.nextInt();

System.out.println ("NUM_PAGES:
+ num_pages);

}

public ~~sting~~ String () {

{

return ("NAME : " + name +
"AUTHOR : " + author + "PRICE : "
+ price + "NUM_PAGES : "
+ num_pages);

}

class book {

public static void main (String
args []) {

Scanner s = new Scanner (System.in);

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

Scanner sr = new Scanner (System.in);
n = sr.nextInt();

book b1 [] = new book [n];

for (int j=0; j < n; j++)

System.out.println ("Enter DF:

```
Page _____  
if (ch == 1)  
    b1[i].getdetails();  
    System.out.println();  
}  
else  
    String details = b1[i].toString();  
    System.out.println(details);  
}  
as "ZOMOZIONA + 10017" book  
}
```

OR

Enter the number of books:

ENTER 2001 : 1

Enter DETAILS

1. Set using constructor

2. Set using method :-

2

Enter name of book : Inka

Enter author's name : RPT

Enter price of book : 22

Enter num of pages : 22

DETAILS SET SUCCESSFULLY :)

ENTER DETAILS

1. Set using constructor
2. Set using method
3. Set using properties

PRINTING Book DETAILS

1. display using method
2. display using ToString
- 3.

NAME : lmao

AUTHOR : rfgs

PRICE : 22

NUM PAGES : 22

PRINTING Book DETAILS

1. display using * method
2. display using ToString

NAME : TINKLE AUTHOR : ANONYMOUS

PRICE : 20 NUM PAGES : 100

Output

```
ENTER DETAILS
1:set using constructor
2:set using method
2
enter name of book: lmao
enter author's name: rfgg
enter price of book: 22
enter num of pages: 22
DETAILS SET SUCCESSFULLY :)
ENTER DETAILS
1:set using constructor
2:set using method
2
enter name of book: lmao
enter author's name: rfgg
enter price of book: 22
enter num of pages: 22
DETAILS SET SUCCESSFULLY :)

PRINTING BOOK DETAILS
1:display using method
2:display using toString
3
NAME: lmao AUTHOR: rfgg PRICE: 22 NUM_PAGES: 22

PRINTING BOOK DETAILS
1:display using method
2:display using toString
1
NAME: lmao
```

```
enter price of book: 22
enter num of pages: 22
DETAILS SET SUCCESSFULLY :)
ENTER DETAILS
1:set using constructor
2:set using method
2
enter name of book: lmao
enter author's name: rfgg
enter price of book: 22
enter num of pages: 22
DETAILS SET SUCCESSFULLY :)

PRINTING BOOK DETAILS
1:display using method
2:display using toString
3
NAME: lmao AUTHOR: rfgg PRICE: 22 NUM_PAGES: 22

PRINTING BOOK DETAILS
1:display using method
2:display using toString
1
NAME: lmao
AUTHOR: rfgg
PRICE: 22
NUM_PAGES: 22
```

```
C:\Users\archi\OneDrive\Desktop\java>
```

LAB 4

Question

Develop a Java program to create an abstract class named Shape that contains two integers and an empty method named printArea(). Provide three classes named Rectangle, Triangle and Circle such that each one of the classes extends the class Shape. Each one of the classes contain only the method printArea() that prints the area of the given shape

Code

Week 7

Date _____
Page _____

Q) WAP To create a class Figure with an instance variable - no_perim and abstract methods calc_area() and calc_perim(). Extend class ^{same name} Create two objects R1 each of the classes and calc area and disp play area and perimeter.

import java.util.Scanner;

abstract class shape

{ int a, b; }

shape (int x, int y)

{ a=x; b=y; }

a=x;
b=y;

? X

shape (int x)

{
 a=x;
}

X

public void printarea();

class rectangle extends shape

public void printArea()
9

System.out.println("area of
circle is " + (a * a * 3.14))

3
7

public class G33R {
 ^(write start)

public static void main (String args)
9

triangle t1 = new triangle (50,
rectangle r1 = new rectangle

(10, 20),

circle c1 = new circle (5);

t1.printArea();

r1.printArea();

c1.printArea();

3

~~Output~~

area of triangle is : 600.0

area of rectangle is : 200

area of circle is : 78.5

Output

```
C:\Users\archi\OneDrive\Desktop\java>javac absrt.java
C:\Users\archi\OneDrive\Desktop\java>java absrt.java
area of triangle is: 600.0
area of rectangle is: 200
area of cirle is: 78.5
C:\Users\archi\OneDrive\Desktop\java>
```

LAB 5

Question

a Java program to create a class Bank that maintains two kinds of account for its customers, one called savings account and the other current account. The savings account provides compound interest and withdrawal facilities but no cheque book facility. The current account provides cheque book facility but no interest. Current account holders should also maintain a minimum balance and if the balance falls below this level, a service charge is imposed.

Create a class Account that stores customer name, account number and type of account. From this derive the classes Cur-acct and Sav-acct to make them more specific to their requirements. Include the necessary methods in order to achieve the following tasks:

a) Accept deposit from customer and update the balance.

b) Display the balance.

c) Compute and deposit interest

d) Permit withdrawal and update the balance

Check for the minimum balance, impose penalty if necessary and update the balance.

Code

Week 5

Q) Develop a WAP to create a Class Bank that maintains two kinds of account for its customers, one called Savings account and the other current account. The Savings account provides CT and withdrawal facility but no cheque book facility but no interest. Current account holders should also maintain a minimum balance and if balance falls below please impose a

- a) Accept deposit from customer and update the balance and interest.
- b) Display the balance.
- c) Compute and display interest.
- d) Permit withdrawal and update the balance.

Check for the minimum balance, impose penalty if necessary and update the balance.

```
import java.util.Scanner;  
class Account {  
    String name; int account_num;  
    String acc_type; double balance;  
    Account(String n, int acn, String atyp, double b) {  
        name = n; account_num = acn;  
        acc_type = atyp; balance = b; }  
}
```

```
name = n;  
account_num = acn;  
acc_type = atyp;  
balance = b;
```

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

```
void display_bal()  
{
```

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

```
void deposit_interest()  
{
```

```
double int_rate = 0.03;
```

wire
and
red.
calcu
-b Br
me

Date _____
Page _____

rectangle (int x, int y),

{
super (x, y);

public void printarea(),

System.out.println ("area of
rectangle is : " + (a * b));

(class) triangle extends shape

{

triangle (int x, int y)

{
super (x, y);

public void printarea(),

public void printarea()

System.out.println ("area of triangle
is : " + (a * b * 0.5));

(class) circle extends shape

{

circle (int x)

{
super (x);

{

double time = 0;

System.out.println("Enter the
time period");

time = sc.nextInt();
double amount;

amount = balance * Math.pow((1 +
int_rate), time);

balance = amount;

void withdraw(int val);

void withdraw();

void (val > balance)

System.out.println("Amount of
funds withdrawn less");

balance -= val;

System.out.println("Withdrawal
successful");

System.out.print("New balance: " +
balance);

Date _____
Page _____

void check_min()

Double min_bal = 1000.00;

Double penalty = 100.00;

if (balance < min_bal)

{

System.out.println ("balance less
than minimum balance, penalty
imposed");

}

balance -= penalty;

void check_min()

{

Double min_bal = 1000.00;

Double penalty = 100.00;

if (balance < min_bal)

{

System.out.println ("balance
less than minimum
balance, penalty
imposed");

balance -= penalty;

else

void depositInterest()

{

System.out.println("current
balance does not provide any interest");

}

void withdraw(int val)

{

System.out.println("current
balance does not provide withdraw
facility");

(4) Solt, 2012

VIII
9

S

Class

void checkMin()

{

Double minBal = 1000.00;

Double penalty = 100.00;

if (balance < minBal)

{

System.out.println("balance
less than minimum
balance, penalty imposed");

balance -= penalty;

}

else {

{

System.out.println("balance
higher than minimum balance");

}

}

void cheqWithdrawl(int n)

balance = val;

System.out.println ("withdraw
success");

System.out.println ("new balance",
balance);

class bank

public static void main (String
args [])

Scanner sc = new Scanner
(System.in);

System.out.println ("Enter your
name, account number, to count
type (savings / current), balance");

String name = sc.nextLine();

int account_no = sc.nextInt();

String acc_type = sc.nextLine();

double balance = sc.nextDouble();

ic (acc_type equals ("Savings"))
9

Sav_acc = new Sav_acc
(name, account_no, acc_type,
balance);

int choice = 0;

while (choice != 6)

{

System.out.print("1. depo

1n 2. display balance in 3. compute and

deposit interest in 4. withdraw in 5.

Check for minimum balance in 6. exit

choice = sc.nextInt();

switch (choice)

case 1: System.out.print("1

2. deposit : Enter the value to

deposit"),

int val = sc.nextInt();

1. deposit .(val);

break;

case 2: a1. display_bal();

break;

System.out.println ("balance is higher
than minimum balance");

3

8 (balance is less than minimum balance)

7

(less (current account extends account

2

double balance;

curr_acct (String n, int ac, String
ac_type, Double bl)

2

name = n;

account_num = ac;

ac_type = ac_type;

balance = bl;

3

(balance > min_balance)

void deposit (int val)

2

balance += val;

3

void display_bal ()

2

System.out.println ("Balance is :
" + balance);

3

case 2: a1.depositInterest();
break;

case 3: System.out.println("enter
the value to withdraw");
int withd = sc.nextInt();
a1.withdraw(withd);
break;

case 5: a1.checkMin();
break;

1. deposit
2.
3.
4.
5.
6. exit();

case 6: System.out.println("exited");
break;

default: System.out.println("enter a
valid choice");
break;

else {

(~~if (cur_acct > a1.getBalance())~~)
choice, accountnum, acc_type,
balance);

int choice = 0;

while (choice != 6)

System.out.println("1. deposit in

2. display balance in 3. compute acc deposit
interest in 4. withdraw using cheque
in 5. check for minimum balance in 6. fix interest

choice = Scanner.nextInt();

switch (choice)

{

case 1: System.out.println("Enter the value to deposit")

int val = Scanner.nextInt();

al.deposit(val);

break;

case 2: al.displayBal();

break;

case 3: al.depositInterest();

break;

case 4: System.out.println("Enter the value to withdraw")

int withdraw = Scanner.nextInt();

al.withdrawal(withdraw);

break;

1
Enter valy to deposit

1000

1. deposit

2. display balance

3. compute & deposit interest

4. withdraw

5. check for minimum balance

6. exit

2

Balance is : 11000.0

1. deposit

2. display balance

3

entering the time period say that

2

1. deposit

:

:

2

Balance is : 12127.3

length of deposit say 1 month

1. deposit

2. display balance say 1 month

3

Enter valy to withdraw

1000

withdraw successfull.

Output

```
Command Prompt + X
enter your name, account number, account type(savings/current), balance
Archit
10000001
savings
10000
1.deposit
2.display balance
3.compute and deposit interest
4.withdraw
5.check for minimum balance
6.exit
1
enter the value to deposit
1000
1.deposit
2.display balance
3.compute and deposit interest
4.withdraw
5.check for minimum balance
6.exit
2
Balance is: 11000.0
1.deposit
2.display balance
3.compute and deposit interest
4.withdraw
5.check for minimum balance
6.exit
3
enter the time period

Command Prompt + X
6.exit
3
enter the time period
2
1.deposit
2.display balance
3.compute and deposit interest
4.withdraw
5.check for minimum balance
6.exit
2
Balance is: 12127.5
1.deposit
2.display balance
3.compute and deposit interest
4.withdraw
5.check for minimum balance
6.exit
4
enter the value to withdraw
1000
withdrawal successful
new balance: 11127.5
1.deposit
2.display balance
3.compute and deposit interest
4.withdraw
5.check for minimum balance
6.exit
5
```

```
Command Prompt + × - □ ×

1.deposit
2.display balance
3.compute and deposit interest
4.withdraw
5.check for minimum balance
6.exit
4
enter the value to withdraw
1000
withdrawal successful
new balance: 11127.5
1.deposit
2.display balance
3.compute and deposit interest
4.withdraw
5.check for minimum balance
6.exit
5
balance higher than minimum balance
1.deposit
2.display balance
3.compute and deposit interest
4.withdraw
5.check for minimum balance
6.exit
6
exited

C:\Users\archi\OneDrive\Desktop>java bank.java
enter your name, account number, aacount type(savings/current), balance
```

```
Command Prompt + × - □ ×

C:\Users\archi\OneDrive\Desktop>java bank.java
enter your name, account number, aacount type(savings/current), balance
Archit
10000000
current
100000
1.deposit
2.display balance
3.compute and deposit interest
4.withdraw using cheque
5.check for minimum balance
6.exit
2
Balance is: 100000.0
1.deposit
2.display balance
3.compute and deposit interest
4.withdraw using cheque
5.check for minimum balance
6.exit
3
Current account doesnt provide any interest
1.deposit
2.display balance
3.compute and deposit interest
4.withdraw using cheque
5.check for minimum balance
6.exit
6
```

LAB 7

Question

Write

a program that demonstrates handling of exceptions in inheritance tree. Create a base class called "Father" and derived class called "Son" which extends the base class. In Father class, implement a constructor which takes the age and throws the exception WrongAge() when the input age<0. In Son class, implement a constructor that cases both father and son's age and throws an exception if son's age is >=father's age.

Code

Topic

Date _____
Page _____

- Q) WAP to create a base class called "father" and derived class called "son" which extends the base class. In father class, implement a constructor which takes the age and throws the exception "Wrong age" when the input age is < 0. In son class, implement a constructor that takes both father and son's age and throws an exception if son's age is >= father's age.

import java.util.Scanner;

class Father

{

 int fatherAge;

 public Father (int fa)

{

 try {

 fatherAge = fa;

 if (fatherAge < 0)

{

 throw new Exception();

}

 ("Error! Age is less than 0")

}

is called
"son"
In Father
which takes
in Wray my
class, implemented
other and
exception like

else {

Father-age = Fa;

}

catch (Exception e)

{

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

}
}

Class Son extends Father

{

int son-age;

public Son (int Fa, int Sa)

super (Fa);

try :

Son-age = Sa;

if exception
less than

; if (son-age < 0)

2

throw new Exception ("Error !

Son's age is less than 0");

3

else if (son.age >= father.age)
{
 throw new Exception ("Error!
Son's age cannot be more than
the father's age");
}

else
{
 son.age = sa;
}

catch (Exception e)

System.out.println ("Caught : " + e)

void display()

System.out.println ("Father's age = "
+ father.age);

System.out.println ("Son's age = " + son.age);

System.out.println ("Son's age = " + son.age);

Date _____
Page _____

class Inheritance tree & extends
Exception &

① → public static void main (String args)

Scanner ss = new Scanner (System.in);

int a, b;
System.out.println ("Enter the
father's age");

a = ss.nextInt();
System.out.println ("Enter the son's
age");

b = ss.nextInt();

System.out.println ("Enter the son's
age");

b = ss.nextInt();

age = "

Son ob1 = new Son(a, b);

ob1.display();

" + son.age);

5

Date / /
Page _____

deposit
charge
In 6. Exit();

case 5:
al. check_min();
break;

case 6:
System.out.println("exit()");
break;

default:
System.out.println("enter a valid
choice");
break;

5

7

7

Output - savings acc

Enter your name, account number, account
type (savings / current), balance
Amount
1000 000

Savings
1000

1. deposit
2. display balance
3. compute and deposit interest
4. Withdraw
5. check for minimum balance
6. Exit

off

Entr father's age
25

Entr son's age

23

Father's age = 25

Son's age = 23

Entr father's age

25

Entr son's age

26

Catch.. java.lang.Exception: Error! Son's

age cannot be more than father's age.

Father's age = 25

Son's age = 26

Entr father's age

0

Entr son's age

0
Error! Son's age

less than 0

caught
java.lang.Exception: Error!
Son's age
cannot be more than father's age.
Son's age = 0

Output

```
Command Prompt + X - ×

C:\Users\archi\OneDrive\Desktop>javac father.java
C:\Users\archi\OneDrive\Desktop>java father.java
Enter the father's age
25
Enter the son's age
23
Father's age = 25
Son's age = 23

C:\Users\archi\OneDrive\Desktop>java father.java
Enter the father's age
25
Enter the son's age
26
Caught : java.lang.Exception: Error! Son's age cannot be more than the Father's age
Father's age = 25
Son's age = 26

C:\Users\archi\OneDrive\Desktop>java father.java
Enter the father's age
0
Enter the son's age
-1
Caught : java.lang.Exception: Error! Son's age is less than 0
Father's age = 0
Son's age = -1

C:\Users\archi\OneDrive\Desktop>
```

LAB 8

Question

Write a

program which creates two threads, one thread displaying “BMS College of Engineering” once every ten seconds and another displaying “CSE” once every two seconds.

Code

Week 8

Q) WAP : Create two threads , one thread displaying "BMS college of Engineering" once every ten seconds and another displaying "(SEI Once every two seconds,

class bms implements Runnable

{

 Thread t1;

 bms();

{

 bms();

{

 t1 = new

 Thread (t1, "bms");

 son)

 public void run()

 try :

 for (int i=5; i>0; i--)

 System.out.println ("BMS college
 of Engineering ");

 Thread.sleep (10000);

}

} catch (InterruptedException e)

{

catch (InterruptedException e)

} System.out.println ("SE interrupted
in ");

System.out.println ("Existing : "
+ t2);

Class thread prg

{ public static void main (String
args [])

bms Obj1 = new bms ();
cse Obj2 = new cse ();
Obj1.t1.start ();
Obj2.t2.start ();

System.out.println ("CSE");

Thread.sleep (2000);

Output

main (String args [])

CSE

LSE

CSE

LSE

BMS college of Engineering
CSE

BMS college of Engineers

BMS college of Engineers

BMS college of Engineers

BMS college of Engineers

Date _____
Page _____

System.out.println("BMS interrupted by")

System.out.println("Exiting "+t)

class CSE implements Runnable

{

 Thread t;

 CSE();

 {}

 t = new Thread(this, "CSE");

 public void run()

 {

 try {

 for (int i=5; i>0; i--)

 {

System.out.println("CSE")

Thread.sleep(2000);

 catch (InterruptedException e) {

 public static void main

 {

CS

CS

CS

CS

BS

CS

Output

```
BMS College of Engineering
CSE
CSE
CSE
CSE
CSE
BMS College of Engineering
Exiting: Thread[#22,cse,5,main]
BMS College of Engineering
BMS College of Engineering
BMS College of Engineering
Exiting: Thread[#21,bms,5,main]
```

LAB 9

Question

Create a package **CIE** which has two classes- **Student** and **Internals- a subclass of Student**. The class **Student** has members like usn, name, sem. The class **internals** has an array that stores the internal marks scored in five courses of the current semester of the student.

Create another package **SEE** which has the class **External** which is a derived class of **Internals**. This class has an array that stores the **SEE** marks scored in five courses of the current semester of the student.
Import the two packages in a file that declares the final marks of n students in all five courses.

Code

Q1 Create package CIE with two classes Student and Internals - a sub-class of Grade
 In Class Student has members like USN (name), Sem, The class Internals has an array that stores the internal marks of student
 in 5 courses of the current sem by
 of students Create another package SECE
 which has the class External which is
 derived (by) of Internals. This plays as
 an array but stores the marks of student
 in five courses of the current sem of
 the student. import the two packages in a file
 and declare the final marks of student in
 all 5 courses.

(a)

```
package CIE;
import java.util.Scanner;
public class Internals {
    public int Imarks[5] = new int[5];
    public void gotmi() {
```

```
Scanner ss = new Scanner(System.in)
```

```
System.out.println("Enter marks")
```

```
for (int i=0; i<5; i++)
```

```
Imarks[i] = ss.nextInt();
```

student
is an
object
of class
Student

```
public void disp()
{
    System.out.println ("Internal Marks");
    for (int i=0; i<=4; i++)
        System.out.println ("Subject" + i + "="
                            + Imarks[i]);
}
```

```
package com.luminar.han;
import java.util.Scanner;
public class student
```

```
public String usn, name;
public int sem;
public void getd()
```

```
Scanner s = new Scanner (System.in);
System.out.println ("Enter USN, Name & Sem");
usn = s.nextLine();
name = s.nextLine();
sem = s.nextInt();
```

```
public void disp()
```

```
System.out.println ("\n Student Details : \n")
```

Usw : " + usn + "\nName!" + name +
"\n SPM" + sem);

S

import CIE.*;
import SEB.*;

class p-main
{

public static void main(string args[])
{

Student s1 = new Student();

s1.getid() ; s1.dispt;

External el = new External();

el.getmi() = el.display(); el.getur();
el.getdismt(); el.printcal();

J

O/P

Entr usn, Nam & SPM

1BM21CS071

NAMe: Archit

SPM :>

Entr marks scores in 5 courses :

95

78

92

91

40 (Total marks of 5 courses)

INTERNAL MARKS

Subject 0 = 93

Subject 1 = 48

Subject 2 = 42

Subject 3 = 41

Subject 4 = 40

Enter external marks Scout & S Comc.

49

48

47

42

45

EXTERNAL MARKS

44

41

Eligentur:

FINAL MARKS

Subject 0 : 69

Subject 1 : 72

Subject 2 : 65

Subject 3 : 64

Subject 4 : 62

①

package SEE;

import java.util.Scanner;

import *;

public class External extends Cls implements

{

int Smarks[3] = new int[3];

public void getm()

{

Scanner ss = new Scanner(System.in);

System.out.println("Enter External marks

Scored in 5 courses");

for (int i=0; i<5; i++)

{

Smarks[i] = ss.nextInt();

}

}

public void disp()

{

System.out.println("External Marks");

for (int i=0; i<5; i++)

{

System.out.println("Subject " + i +
" = " + Smarks[i]);

}

}

```
public void finalM1()
{
    int finalCJ = new int[5];
    for (int j=0; j<5; j++)
    {
        final Cj3 = marks[j] + (smarks[j]/2);
    }
    System.out.println ("final Marks");
    for (int i=0; i<5; i++)
    {
        System.out.println ("Subject" + i + "="
                           + final [i]);
    }
}
```

Output

```
Microsoft Windows [Version 10.0.22621.1105]
(c) Microsoft Corporation. All rights reserved.

C:\Users\archi>cd C:\Users\archi\OneDrive\Desktop\java report\week9
C:\Users\archi\OneDrive\Desktop\java report\week9>set path="C:\Program Files\Java\jdk-19\bin"
C:\Users\archi\OneDrive\Desktop\java report\week9>javac CIE\Internals.java
C:\Users\archi\OneDrive\Desktop\java report\week9>javac CIE\Student.java
C:\Users\archi\OneDrive\Desktop\java report\week9>javac SEE\External.java
C:\Users\archi\OneDrive\Desktop\java report\week9>javac P_main.java

C:\Users\archi\OneDrive\Desktop\java report\week9>java P_main.java
Enter USN, NAME & SEM
1BM21CS031
Archit
3

Student Details:
USN:1BM21CS031
NAME:Archit
SEM:3
Enter marks scored in 5 courses:
45
48
42
41
```

```
48
42
41
40
INTERNAL MARKS
Subject0=45
Subject1=48
Subject2=42
Subject3=41
Subject4=40
Enter external marks scored in 5 courses:
49
48
47
46
45
EXTERNAL MARKS
Subject0=49
Subject1=48
Subject2=47
Subject3=46
Subject4=45
FINAL MARKS
Subject0=69
Subject1=72
Subject2=65
Subject3=64
Subject4=62

C:\Users\archi\OneDrive\Desktop\java report\week9>java P_main.java
```