VISVESVARAYA TECHNOLOGICAL UNIVERSITY

"JnanaSangama", Belgaum -590014, Karnataka.

LAB REPORT
on

Object Oriented Java Programming

Submitted by

SUSHANTH(1BM21CS227)

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)

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 "**Object Oriented Java Programming**" carried out by **SUSHANTH(1BM21CS227),** 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 2022-23. The Lab report has been approved as it satisfies the academic requirements in respect of Object Oriented Java Lab **- (21CS3PCOOJ)** work prescribed for the said degree.

Dr.Nandini Vineeth   **Dr. Jyothi S Nayak**
Assistant professor   Professor and Head
Department of CSE   Department of CSE
BMSCE, Bengaluru   BMSCE, Bengaluru

`

Index Sheet

	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.	
6	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=father's age.	29
7	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.	33
8	Create a package CIE which has two classes-Student and Internals. The class Personal has members like usn, name, sem. The class internals has an array that stores the internal marks scored	37

	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 Student. 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	
9	Demonstrate Inter process Communication and deadlock	43
10		

Course Outcome

CO1	Apply the knowledge of Java concepts to find the solution for a given problem.
CO2	Analyse the given Java application for correctness/functionalities..
CO3	Develop Java programs / applications for a given requirement.

CO4	Conduct practical experiments for demonstrating features of Java

LAB PROGRAM 1:

Develop a Java program that prints all real solutions to the quadratic equation ax2+bx+c = 0. Read in a, b, c and use the quadratic formula. If the discriminate b2 -4ac is negative, display a message stating that there are no real solutions.

```java
import java.util.Scanner;
import java.lang.Math;
class quadratic
{
double d,val;
void find(double a,double b,double c)
{
if(a==0)
{
System.out.println("a cannot be zero");
return;
}
else
{
   d=(b*b)-(4*a*c);
}
if(d>0)
{
System.out.println("The roots are real and unique");
System.out.println((-b+Math.sqrt(d))/(2*a)+"\n"+(-b-Math.sqrt(d))/(2*a));
}
else if(d==0)
{
```

```java
System.out.println("the roots are real and equal");
System.out.println(-b/(2*a));
}
else
{
System.out.println("there is no real root");
double img=Math.sqrt(-d)/(2*a);
double realpart=-b/(2*a);
System.out.println("real part is "+realpart+" imaginary part is "+img);
}
}
}
class Main
{
public static void main(String args[])
{
quadratic ob1=new quadratic();
System.out.println("enter the value of a,b and c");
Scanner sc=new Scanner(System.in);
double a1=sc.nextDouble();
double b1=sc.nextDouble();
double c1=sc.nextDouble();
ob1.find(a1,b1,c1);
}
}
```
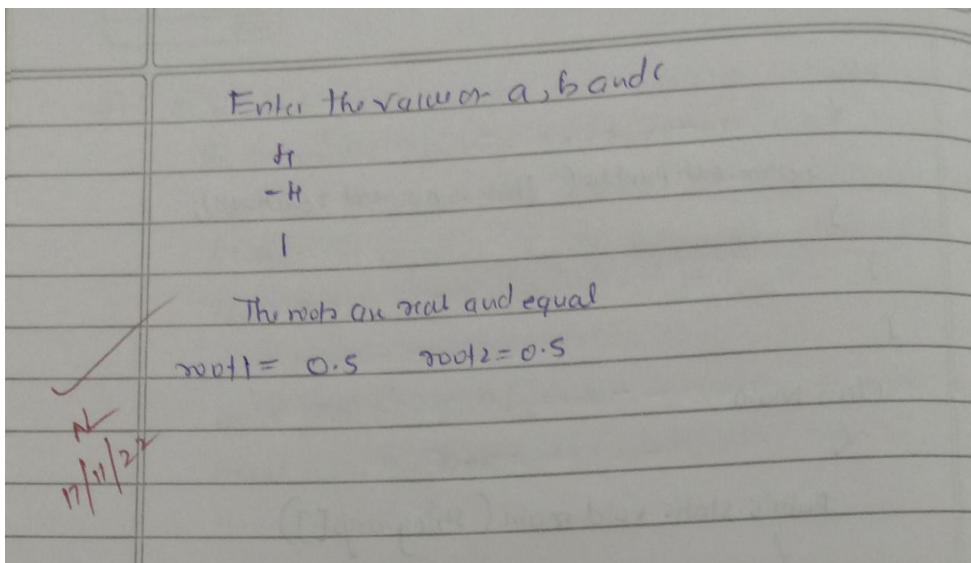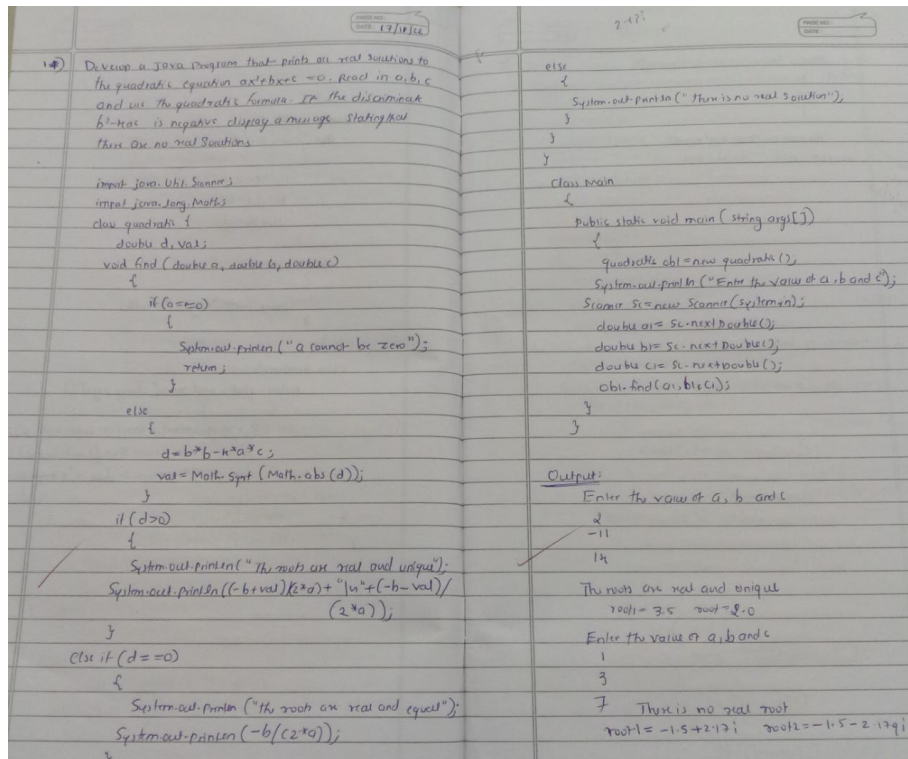
Observation:

(1) 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.

```java
import java.util.Scanner;
import java.lang.Math;
class quadratic {
    double d, val;
    void find (double a, double b, double c)
    {
        if (a==0)
        {
            System.out.println ("a cannot be zero");
            return;
        }
        else
        {
            d = b*b - 4*a*c;
            val = Math.sqrt (Math.abs (d));
        }
        if (d>0)
        {
            System.out.println (" The roots are real and unique");
            System.out.println ((-b+val)/(2*a) + "\n" + (-b-val)/
                                 (2*a));
        }
        else if (d == 0)
        {
            System.out.println ("the roots are real and equal");
            System.out.println (-b/(2*a));
        }
        else
        {
            System.out.println (" There is no real solution");
        }
    }
}
class main
{
    public static void main (String args[])
    {
        quadratic obj = new quadratic();
        System.out.println ("Enter the value of a, b and c");
        Scanner sc = new Scanner (System.in);
        double a = sc.nextDouble();
        double b = sc.nextDouble();
        double c = sc.nextDouble();
        obj.find (a, b, c);
    }
}
```

Output:
```
Enter the value of a, b and c
2
-11
14
The roots are real and unique
root1 = 3.5   root2 = 2.0
Enter the value of a, b and c
1
3
7   There is no real root
root1 = -1.5 + 2.17i   root2 = -1.5 - 2.17i
```



```
Enter the value of a, b and c
4
-4
1
The roots are real and equal
root1 = 0.5   root2 = 0.5
```

Output:

```
Select Command Prompt

C:\Users\BMSCECSE\Desktop>java Main
enter the value of a,b and c
2
-11
14
The roots are real and unique
3.5
2.0

C:\Users\BMSCECSE\Desktop>java Main
enter the value of a,b and c
1
3
7
there is no real root
real part is -1.5 imaginary part is 2.179449471770337

C:\Users\BMSCECSE\Desktop>java Main
enter the value of a,b and c
4
-4
1
the roots are real and equal
0.5

C:\Users\BMSCECSE\Desktop>_
```

LAB PROGRAM 2:

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.

```java
import java.util.Scanner; class student
{float total;
   Scanner sc=new Scanner(System.in); int marks[]=new int[10]; int credit[]=new
int[10]; void display()
{ int i;
float sumofcredit=0;
System.out.println("Enter the name of the student:");
String name= sc.nextLine();
System.out.println("Enter the usn of the student:");
String usn=sc.nextLine();
System.out.println("Enter the credits for 5 subjects sequentially:");
for(i=0;i<5;i++)
  {
    credit[i]=sc.nextInt();
  }
System.out.println("Enter the marks for 5 subjects sequentially:");
for(i=0;i<5;i++)
  {
  marks[i]=sc.nextInt();
  }
for(i=0;i<5;i++)
{
total=total+credit[i];
}
for(i=0;i<5;i++)
{
if(marks[i]==100)
  {
    sumofcredit+=(10*credit[i]);
  }
else sumofcredit+=((marks[i]/10)+1)*credit[i];

}
```

```java
System.out.println("name="+name);
System.out.println("usn="+usn);
System.out.println("the sgpa of student is "+(sumofcredit/total));
}
}
 class Main
{
public static void main(String args[])
{


student st=new student();


st.display();
}
}
```

Observation:

2) Develop a java program to create a class student with number USN, name an array credits. Calculate SGPA.

```java
import java.util.Scanner;

class Student
{ float total;
    Scanner sc = new Scanner(System.in);
    int marks[] = new int[10];
    int credit[] = new int[10];

    void display()
    {
        int sumofcredits = 0, i;
        System.out.println("Enter the name of the student:");
        String name = sc.nextLine();
        System.out.println("Enter the USN of student:");
        String usn = sc.nextLine();
        System.out.println("Enter credits for 5 subjects:");
        for (i=0; i<5; i++)
        {
            credit[i] = sc.nextInt();
        }
        System.out.println("Enter marks for 5 subjects:");
        for (i=0; i<5; i++)
        {
            marks[i] = sc.nextInt();
        }
        for(i=0; i<5; i++)
        {
            total = total + credit[i];
        }
        for (i=0; i<5; i++)
        {
            if (marks[i] ==100)
            {
                sumofcredit = sumofcredit + (10 * credit[i]);
            }
            else
                sumof-credit += ((marks[i]/10)+1) * credit[i];
        }
        System.out.println("name = "+ name+ " "+ "USN = "+ USN+
            " The SGPA = "+ (sumofcredit / total));
    }
}

class Main
{
    public static void main(String args[])
    {   Student st = new Student();
        st.display();
    }
}
```

output:

```
Enter the name of student
Sushanth
Enter the USN
1BM21CS227
Enter the marks for 5 subject
93 84 80 90 80
Enter the credits for 5 subject
4 3 4 3 2 1
name: Sushanth
USN: 1BM21CS227
SGPA: 9.44
```

Output:

```
C:\Users\Admin\Desktop>java Main
Enter the name of the student:
sushanth
Enter the usn of the student:
227
Enter the credits for 5 subjects sequentially:
3
4
4
3
1
Enter the marks for 5 subjects sequentially:
90
89
90
90
80
name=sushanth
usn=227
the sgpa of student is 9.666667

C:\Users\Admin\Desktop>
```

LAB PRGRAM-3:

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.

```java
import java.util.Scanner;
 class books {
int k,l;
String name,author;
Scanner sc=new Scanner(System.in);
int price;
int numpages;


books()
{
price=0;
numpages=0;
}

void get()
{
System.out.println("Enter the name of the book:");
name=sc.nextLine();
System.out.println("Enter the name of the author:");
author=sc.nextLine();
System.out.println("Enter the price of the book:");
price=sc.nextInt();
System.out.println("Enter the no of pages of the book:");
numpages=sc.nextInt();


}
```

```java
public String toString()
{

return name+"\n"+author + "\n"+price+"\n"+numpages+"\n";


/*System.out.println(" name of the boook:");
System.out.println(name);
System.out.println(" name of the author:");
System.out.println(author);
System.out.println("price of book:");
System.out.println(price);
System.out.println("pages of book");
System.out.println(numpages);*/
}
}

class book
{
public static void main(String args[])
{
Scanner sc = new Scanner(System.in);
int i,n;
System.out.println(" enter number of books:");
n=sc.nextInt();
books[] arr=new books[n];
books ident=new books();
for(i=0;i<n;i++) {
arr[i]=new books();
arr[i].get();
}
for(i=0;i<n;i++)
{
  System.out.println("\nThe details of book "+(i+1));

System.out.println(arr[i]);
System.out.println();
```
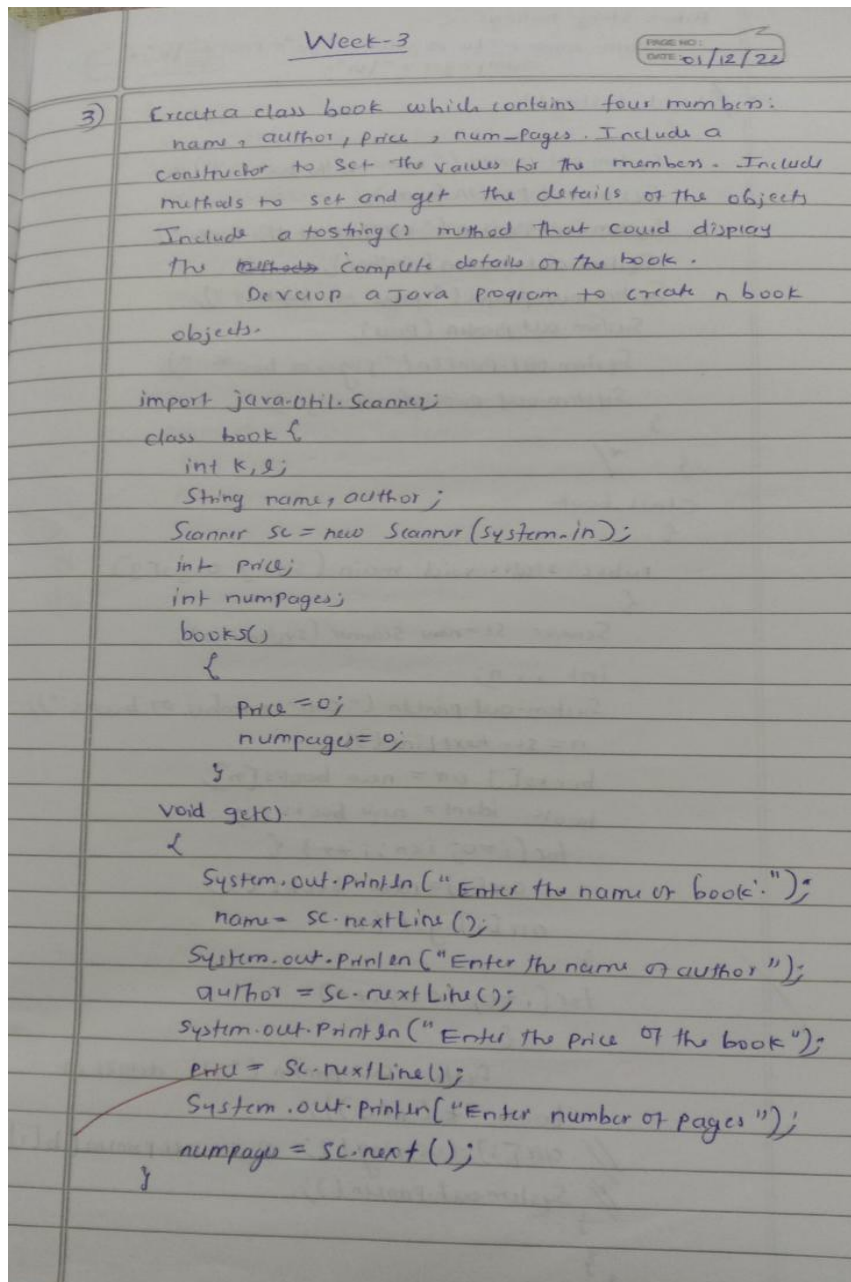
```
}



}
}
```

Observation:

3) 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 object.
Include a tostring() method that could display
the methods complete details of the book.
       Develop a Java program to create n book
objects.

```
import java.util.Scanner;
class book {
    int k, l;
    String name, author;
    Scanner sc = new Scanner(System.in);
    int price;
    int numpages;
    books()
    {
        price = 0;
        numpages = 0;
    }
void get()
{
    System.out.Println("Enter the name of book:");
    name = sc.nextLine();
    System.out.Println("Enter the name of author");
    author = sc.nextLine();
    System.out.Println("Enter the price of the book");
    price = sc.nextLine();
    System.out.Println("Enter number of pages");
    numpages = sc.next();
}
```

```
/* // void testing ()
    {
        System.out.println (" name of the book : ");
        System.out.println (name);
        System.out.println (' author of the book: ")
        System.out.println (author);
        System.out.println (" price of book : ");
        System.out.println (price),
        System.out.println (" pages of book ");
        System.out.println (numpages);
    }
} */
Class book
{
    public static void main (string args[])
    {
        Scanner sc = new Scanner (system.in);
        int i, n;
        System.out.println (" enter number of books:");
        n = sc. nextLine ();
        books[] arr = new books [n];
        book ident = new books ();
        for (i = 0; i < n; i++) {
            arr[i] = new books ();
            arr[i].get ();
        }
        for (i = 0; i < n; i++)
        {
            System.out.println (" The details of
            book " + (i+1));
            // arr[i].tostring (); System.out.println (b[i]);
            // System.out.println ();
```

Output:
Enter the number of books
1

Enter name of book: Toranga
Enter the name author: Sushanth
Enter the price: 100
Enter number of pages: 117

Details of book 1:
Name of book: Toranga
Name of author: Sushanth
Price: 100
Number of pages: 117

N
1/8/22

Output:

```
C:\Users\Admin\Desktop>java book
 enter number of books:
2
Enter the name of the book:
wings of fire
Enter the name of the author:
kalam
Enter the price of the book:
322
Enter the no of pages of the book:
223
Enter the name of the book:
nannna kathe
Enter the name of the author:
unknown
Enter the price of the book:
234
Enter the no of pages of the book:
434

The details of book 1
wings of fire
kalam
322
223



The details of book 2
nannna kathe
unknown
234
434
```

LAB PROGRAM -4:

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

```
abstract class figure
{
   int x,y;
   figure(int a,int b){
     x=a;
     y=b;
   }
   abstract double area();
}

class rectangle extends figure{
   rectangle(int a,int b){
     super(a,b);}
     double area(){
        return x*y;
     }
   }

class triangle extends figure{
   triangle(int a, int b){
     super(a,b);}
     double area(){
        return 0.5*x*y;
     }
   }

class circle extends figure{
   circle(int a){
```

```java
        super(a,a);}
        double area(){
           return 3.14*x*x;
        }
    }



class abst{
    public static void main(String args[]){
        rectangle r=new rectangle(10,20);
        triangle t=new triangle(4,5);
        circle c=new circle(3);

        figure f;

        f=r;
         double a1=f.area();
            System.out.println("area of rectangle:"+a1);

          f=t;
         double a2=f.area();
            System.out.println("area of triangle:"+a2);

        f=c;
         double a3=f.area();
            System.out.println("area of circle:"+a3);

    }
}
```

Observation:

WEEK-4

```
1. Develop a java Program to create an abstract class named
   Shape/figure that contains two integers and an empty
   method named printarea. Provide three classes rectangle,
   triangle, circle include printarea.
   abstract class figure {
       int x; int y;
       figure(int a, int b)
       {
           x=a;
           y=b;
       }
       abstract double print area();
   }

   class rectangle extends figure
   {
       rectangle(int a, int b) {
           super(a,b);
           double printarea() {
               return x*y;
           }
       }
   }

   class triangle extends figure
   {
       triangle(int a, int b) {
           super(a,b);
           double printarea() {
               return 0.5*x*y;
           }
       }
   }
```

```
   class circle extends figure
   {
       circle(int a, int b) {
           super(a,b);
           double printarea() {
               return 3.14*x*x;
           }
       }
   }

   class main {
       public static void main(String args[])
       {
           rectangle s = new rectangle(10,20);
           triangle t = new triangle(2,5);
           circle c = new circle(7, 7);

           f = s;
           double a1 = f.printarea();
           S.O.P("area of rectangle = " + a1);

           f = t;
           double a2 = t.printarea();
           S.O.P("area or triangle = " + a2);

           f = c;
           double a3 = t.printarea();
           S.O.P("area of circle = " + a3))
       }
   }

   output:
   area of rectangle : 200.0
   area of triangle : 10.0
   area of circle : 28.59
```

Output:

```
C:\Users\Admin>set path="C:\Program Files\Java\jdk-19\bin"

C:\Users\Admin>cd "Desktop"

C:\Users\Admin\Desktop>javac abst.java

C:\Users\Admin\Desktop>java abst
area of rectangle:200.0
area of triangle:10.0
area of circle:28.259999999999998

C:\Users\Admin\Desktop>
```

LAB PROGRAM -5:

**Develop 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.**

```java
import java.util.Scanner; class account{
Scanner sc=new Scanner(System.in); String name; int acct_no; int balance,deposit; // String
type;
void deposit()
{
System.out.println("Enter amount for deposit\n"); int amount=sc.nextInt(); balance+=amount;
System.out.println("deposite is successfull\n do you want to check your balance");
System.out.println("1.YES\n2.NO"); int choice=sc.nextInt(); check_balance();
}
void details()
{
System.out.println("Name: "+name);
System.out.println("account_no: "+acct_no);
System.out.println("balance :"+balance);
}
void check_balance()
{
System.out.println("Balance is "+balance);
}
}
class savings extends account
{
double intrest=0; double rate=0.3;
```

```java
savings(String n,int a,int d)
{
name=n; acct_no=a; deposit=d; balance=deposit;
}
void details()
{
System.out.println("Account_type: Savings"); super.details();
}
void withdrawal()
{
System.out.println("Enter amount for withdrawal\n"); int amount=sc.nextInt();
if(balance<amount)
System.out.println("You can withdraw the amount less than "+balance); else
{
balance=balance-amount;
System.out.println("withdrawal is successfull..!!!!!!! \nparty leda "+name);
System.out.println("\nDo you want check your balance\n"); System.out.println("1.YES\n2.NO");
int choice=sc.nextInt(); if(choice==1) check_balance(); return;
}
}
void  get_intrest(int month)
{
month/=3;
intrest=(balance*rate*month/4);
System.out.println("Previous balance: "+balance+"\nIntrest: "+intrest+"\nCurrent balance:
"+(balance+intrest)); balance+=intrest;
}
}
class current extends account
{
int cheque_no;
current(String n,int a,int d)
{
name=n; acct_no=a; deposit=d; balance=deposit;
}
void details()
{
System.out.println("Account_type: Current"); super.details();
}
void withdrawal()
{
System.out.println("Do you have cheque"); System.out.println("1.YES\n2.NO"); int
choice1=sc.nextInt(); if(choice1==1)
{
System.out.println("enter check number"); cheque_no=sc.nextInt();
```

```java
}
System.out.println("Enter amount for withdrawal\n"); int amount=sc.nextInt();
if(balance<amount)
System.out.println("You can withdraw the amount less than "+balance); else
{
balance=balance-amount;
System.out.println("withdrawal is successfull..!!!!!!! \nparty leda "+name);
System.out.println("Do you want check your balance\n");
System.out.println("1.YES\n2.NO"); int choice=sc.nextInt(); if(choice==1) check_balance();
return;
}
}
}
class Main
{ public static void  main(String[] args)
{
Scanner sc=new Scanner(System.in); String name; int acct_no; int deposit; int choice;
System.out.println("Enter your name\n"); name=sc.nextLine();
System.out.println("Type of account\n"); System.out.println("1.current\n2.Savings");
choice=sc.nextInt(); if(choice==2)
{
System.out.println("Enter amount for deposition and minimum amount 1000 rupees\n"); int
x=sc.nextInt(); if(x<1000){
System.out.println("cant deposit less tahn 1000");
} else { deposit=x;
}
System.out.println("Create account number \n"); acct_no=sc.nextInt(); savings s=new
savings(name,acct_no,deposit); while(true)
{
System.out.println("\nEnter your choice\n1.depost\n2.withdrawal\n3.Details of
account\n4.Intrest Calculate\n5.exit\n");
int ch=sc.nextInt(); if(ch==1)
s.deposit(); else if(ch==2)
{
s.withdrawal();
}else if(ch==3)
{
s.details(); }
else if(ch==4)
{ int m;
System.out.println("Enter duration for intrest calculate\n"); m=sc.nextInt();
s.get_intrest(m);
}
else if(ch==5)
break;
```

```java
else
{
System.out.println("Invalid Choice..!!!!!\n"); }
}
}
else
{
System.out.println("Enter amount for deposition\n"); deposit=sc.nextInt();
System.out.println("Create account number \n"); acct_no=sc.nextInt(); current c=new
current(name,acct_no,deposit); while(true)
{
System.out.println("\nEnter your choice\n1.depost\n2.withdrawal\n3.Details of
account\n4.exit"); int ch=sc.nextInt(); if(ch==1)
c.deposit(); else if(ch==2)
{
c.withdrawal();
}else if(ch==3)
{
c.details(); }
else if(ch==4)
break;
else
{
System.out.println("Invalid Choice..!!!!!\n"); }
}
}
}
}
```

Observation:

Create a class account that Stores customer name, account number and type of account. From this derive The classes cun-acct and sav-acct to make more Specific to their requirements. Include the necessary methods in order to achieve the following tasks.

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

```java
import java.util.Scanner;
class account {
        Scanner sc= new Scanner (system.in);
        String name;
        int acct-no;
        int balance, deposit;
    void deposit ()
    {
        System.out.println(" Enter amount for deposit");
        int amount= sc.nextInt();
        balance += amount;
        System.out.println(" Deposit is sucessfull\n
        do you want to check your balance");
        System.out.println("1.yes\n2.no");
        Scanf Sqr int choice = sc.nextInt();
                check f  check-balance();
    }
    void details()
    {
        System.out.println(" Name : "+ name);
```

```java
        System.out.println("account-no: "+acct-no);
        System.out.println(" balance : "+balance);
    }
    void check-balance()
    {
        System.out.println (" Balance is "+ balance ");
    }
}
class Savings extends account
{
        int interest =0; double rate = 0.03;
        Savings (string n, int a, int d)
        {
            nam=n;
            acct-no =a;
            deposit = d;
            balance=deposit;
        }
    void details()
    {
        System.out.println (" Account type : Saving
        super.details();
    }
    void withdrawl()
    {
        System.out.println(" Enter amount for
        withdrawl(u");
        int amount = sc.nextInt();
        if( balance <amount)
        System.out.println(" you can withdraw the
        amount less than "+ balance);
        else
        {
```

```java
balance = balance - amount;
System.out.println("withdrawls successful");
System.out.println("Do you want to check balance");
System.out.println("1.yes 2.no");
int choice = sc.nextInt();
if (choice == 1)
{
    check_balance();
    return;
}
return;
}
}
}
                    ①
class current extends saveaccount
{
    int cheque_no;
    current (String n, int a, int d)
    {
        name = n;
        account_no = a;
        deposit = d;
        balance = deposit;
    }
    void details()
    {
        System.out.println("Account type: current");
        Super.details();
    }
                    ②
    void withdraws()
    {
        System.out.println("Do you have cheque");
        System.out.println("1.yes 2.no");
        int choice = sc.nextInt();
```

```java
if (choice == 1)
{
    System.out.println("Enter cheque number");
    chequeno = sc.nextInt();
}
System.out.println("Enter amount for withdrawl");
int amount = sc.nextInt();
if (balance < amount)
    System.out.println("you can withdraw the
amount less than " + balance);
else
{
    double sclargy = 0; serviccharge(amount);
    balance = balance - amount;
    System.out.println("Successful withdraw");
    System.out.println("Do you want to check
balance");
    System.out.println("1.yes 2.no");
    int choice = sc.nextInt();
    if (choice == 1)
        check_balance();
    return;
}}}
class Main {
    Public static void main (String arg[])
    {
        Scanner sc = new Scanner(System.in);
        String name;
        int acct_no;
        int deposit, choice;
        System.out.println("Enter your name");
        name = sc.nextLine();
        System.out.println("Type of account");
        System.out.println("1.current 2.savings");
```

```java
if (choice == 2)
{
    System.out.println("Enter amount for
deposit and minimum amount is 500");
    deposit = sc.nextLine();
    System.out.println("Create account number");
    acct_no = sc.nextInt();
    Savings s = new Savings (name, acct_no, deposit);
    while(true)
    {
        System.out.println("\n Enter choice
1. deposit 2. withdraws 3. Details of account
4. exit");
        int ch = sc.nextInt();
        if (ch == 1)
            s.deposit();
        else if (ch == 2)
            s.withdraws
        else if (ch == 3)
            s. details
        else if (ch == 4)
            break;
        else
            System.out.println("Invalid choice !!!");
    }
}
else
{
    System.out.println("Enter amount for deposit");
    deposit = sc.nextInt();
    System.out.println("Create Account number");
    acct_no = sc.nextInt();
```

```java
    current c = new current(name, acct_no, deposit);
    while (true)
    {
        System.out.println("Enter your choice
1. deposit 2. withdraws 3. details next");
        int ch = sc.nextInt();
        if (ch == 1)
            c.deposit();
        else if (ch == 2)
            c.withdraws();
        else if (ch == 3)
            c.details();
        else if (ch == 4)
            (exit(0) break;
        else
        {   System.out.println("Invalid choice !!");
        }
    }
    }
    }
    }
                    ①
    void get_intrest (int month)
    {
        month /= 3;
        intrest = (balance * rate * month/4);
        System.out.println("Previous balance : " + balance +
        "\n Intrest : " + intrest + "\n current balance:"
        + (balance + intrest));
        balance = balance + intrest;
    }
}
```

```
└─②   void  serviceCharge(double amount){
      {
          double charge = 0;
      System.out.println("Service charge 6% is
          applied because of low balance");
      charge = 0.05 × (minBalance - (balance - amount));
      System.out.println("Service charge: "+charge);
      balance -= charge;
      return;
      }
```

Output:

Enter your name
   Sushanth
   Type of account
   1. current   2. savings
   1
      Enter amount for deposition
      10000
   check your account number
      301987
      Enter your choice
      1. deposit
      2. withdraw
      3. Detail
      4. Exit
   3.
      Sushanth
      10000
      Current

Output:

```
Account_type: Current
Name: sushanth
account_no: 22233
balance :214

Enter your choice
1.depost
2.withdrawal
3.Details of account
4.exit
4

C:\Users\Admin\Desktop>java Bank
Enter your name

sushanth
Type of account

1.current
2.Savings
1
Enter amount for deposition

122
Create account number

12

Enter your choice
1.depost
2.withdrawal
3.Details of account
4.exit
4
```

LAB PROGRAM-6:

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=father's age.

```java
import java.util.*; class father extends Exception{
   int f_age;    father(int a)
   {
     f_age=a;
   }
   public String toString()
   {
     return "Age : "+f_age+"\n Wrong age Input\nage should be greater than 0";
   }
}
class son extends father{    int s_age;    son(int f,int s)
   {
     super(f);        s_age=s;
   }
   public String toString()
   {
     return "\nson age:"+s_age+"\n"+"father age:"+f_age+"\n"+"Son age should be
less than father age";
   }
}
class MyException
{
   static void WrongAge(int f_age,int s_age) throws son,father
   {
     if(f_age<=0 || s_age<=0)        throw new father(f_age);        else
if(s_age>=f_age)        throw new son(f_age,s_age);
   }
   static void WrongAge(int age) throws father{
     if(age<=0)
     throw new father(age);
```

```java
    }
    public static void main(String []args)
    {
        int f_age,s_age;
        Scanner sc=new Scanner(System.in);

        try{
            System.out.println("Enter father age\n");
 f_age=sc.nextInt();
            WrongAge(f_age);
            System.out.println("Enter son age\n");
s_age=sc.nextInt();
            WrongAge(s_age);
            WrongAge(f_age,s_age);
            System.out.println("No error occured\n"+"son age:"+s_age+"\n"+"father
age:"+f_age);
        }
        catch(son s)
        {
            System.out.println("\nError caught\n"+s);
        }
        catch(father f)
        {
            System.out.println("\nError caught\n"+f);
        }

    }
}
```

Observation:

1) Write a program that demonstrates handling of Exceptions in inheritance tree. Create a base class called "Father" and derived class "Son" which extends 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 takes both father and son's age and throws an exception if Son's age >= father's age.

```
import java.util.*;
clan father extends Exception{
    int f_age;
    father (int a)
    {
        f_age = a;
    }

    Public String toString ()
    {   return "Age:" + f_age "\n wrong a age
        input should be greater than 0";
    }
}

clan Son extends father{
    int s_age;
    Son(int f, int s)
    {   Super(f);
        s_age = s;
    }

        Public String toString ()
        {
            return "\n son age" + s_age "\n father
            age: " + f_age "\n"father age should be
            greater than son";
        }
    }
}
```

```java
class MyException{
    static void WrongAage(int f_age, int s_age)
    throws Son, Father
    {
        if(# f_age <=0 || s_age<=0)
            throw new father(f_age);
        elseif( s_age >= f_age)
            throw new Son(f_age, s_age);
    }

    static void WrongAge(int age) throws father{
        if(age<0)
            throw new father(age);
    }

    public static void main(String arg[])
    {   int f_age, s_age;
        Scanner sc = new Scanner(System.in);

        try{
            System.out.println("Enter father age");
            f_age = sc.nextInt();
            WrongAage(f_age);
            System.out.println("Enter son age");
            s_age = sc.nextInt(); WrongAge(s_age);
            WrongAage(f_age, s_age);
            System.out.println("No error occured
            father age: "+f_age +"\n son age: "+
            s_age );
        }
        catch(Son s)
        {
            System.out.println("error caught \n");
            System.out.println("Error Caught"+s);
        }
    }
}

output:
Enter father Age
50
Enter son age
70
Error caught
Son Age:70
father Age: 50
    son age should be less than father

Enter father Age
0
Error caught
    wrong age input
    Age should be greater than 0

Enter father Age
12
Enter Son Age
-1
Error caught
Wronge Age input
    Age should be greater than 0

Enter father Age
50
Enter Son Age
```

Output:

```
C:\Users\HP\Desktop>java MyException
Enter father age

50
Enter son age

78

Error caught

son age:78
father age:50
Son age should be less than father age

C:\Users\HP\Desktop>java MyException
Enter father age

0

Error caught
Age : 0
 Wrong age Input
age should be greater than 0

C:\Users\HP\Desktop>java MyException
Enter father age

12
Enter son age

-1

Error caught
Age : -1
 Wrong age Input
age should be greater than 0
```

LAB PROGRAM -7:

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.

```
class one extends Thread{
public void run(){
 try{
for(int i=0;i<5;i++){
System.out.println("\nBMS College of Engineering\n");
Thread.sleep(10000);
}
}
catch(InterruptedException e){
System.out.println("sleep exception\n");
}
}
}
class two extends Thread{ public void run(){ try{
for(int i=0;i<5;i++){
System.out.println("\nCSE\n");
Thread.sleep(2000);
}
}
catch(InterruptedException e){
System.out.println("sleep exception\n");
}
}
}
class Main{ public static void main(String args[]){
 System.out.println("\nin main class\n");
new one().start();
 new two().start();
}
}
```

Observation:

1] Write a program which creates two threads one thread displaying "BMS college of Engineering" one every ten seconds and another displaying "CSE" once every two seconds

```
Class one extends Thread {
    public void run() {
        try {
            for (int i=0; i<3; i++) {
                System.out.println("\nBMS college of
                Engineering \n");
                Thread.sleep (10000);
            }
        }
        catch (Exception e) {
            System.out.println(" sleep Exception");
        }
    }
}

Class two extends Thread {
    Public void run() {
        Try {
            for (int i=0; i<3; i++) {
                System.out.println(" CSE");
                Thread.sleep(2000);
            }
        }
        catch (Exception e) {
            System.out.println(" sleep Exception");
        }
    }
}
```

```
Class main {
    Public static void main(String args[]) {
        one t1 = new one();
        two t2 = new two();
        t1.start();
        t2.start();
    }
}
```

Output:

```
BMS college of Engineering
BMS college of Engineering
BMS College of Engineering
CSE
CSE
CSE
```

Output:

```
in main class

BMS College of Engineering

CSE

CSE

CSE

CSE

CSE

BMS College of Engineering

BMS College of Engineering

BMS College of Engineering

BMS College of Engineering
```

OPEN ENDED PROGRAM -8:

Create a package CIE which has two classes- Student and Internals. The class Personal 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 Student. 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.

```
package cie;
public class Internals {
        public int internal[]=new int[5];
}
package cie;

public class Student {
        public String name;
        public int usn;
        public int sem;
}
package see;
import cie.Internals;
public class External extends Student {
        public int external[]=new int[5];
}



import java.util.Scanner;

import cie.Internals;
import see.External;
public class Marks {

        public static void main(String[] args) {
                int n;
                Scanner sc=new Scanner(System.in);
                        System.out.println("enter number of students");
                        n=sc.nextInt();
                        External student[]=new External[n];
                        Internals internal[]=new Internals[n];
```

```java
                        int final_marks[][]=new int[n][5];
                        for(int i=0;i<n;i++)
                        {
                                student[i]=new External();
                                internal[i]=new Internals();
                                System.out.println("Enter Student "+(i+1)+" name and usn and
sem respectively");
                                student[i].name=sc.next();
                                student[i].usn=sc.next();
                                student[i].sem=sc.nextInt();
                                System.out.println("Enter Internal marks of 5 subject in respective
order");
                                for(int j=0;j<5;j++)
                                {

                                        internal[i].internal_marks[j]=sc.nextInt();
                                }
                                System.out.println("Enter external marks of 5 subject in respective
order");
                                for(int k=0;k<5;k++)
                                {

                                        student[i].external[k]=sc.nextInt();
                                }
                        }
                        for(int i=0;i<n;i++)
                        {
                                for(int j=0;j<5;j++)

        final_marks[i][j]=internal[i].internal_marks[j]+(int)(student[i].external[j]/2);
                        }
                        for(int i=0;i<n;i++)
                        {
                                System.out.println("Name: "+student[i].name);
                                System.out.println("USN: "+student[i].usn);
                                System.out.println("Sem: "+student[i].sem);
                                System.out.println("Marks of the student "+(i+1)+" is");
                                for(int j=0;j<5;j++)
                                {
                                        System.out.println(final_marks[i][j]);
                                }
                        }
        }}
```

Observation:

WEEK-8

Create a package CIE and which has two classes — Student and Internals. The class personal has numbers like USN, Name, sem. The class internals has an array that stores the internal marks scored in five subjects of a student. Create another package SEE, which has the class External which is derived class of student. This class has an array that stores the SEE marks scored in five courses of the student of the current semester. Import the two packages in a file that declares the final marks of the n students in all five courses.

```
Package CIE;

public class Internals {
    public int internal[] = new int[5];
}
public class Student {
    public String name;
    public int USN;
    public int sem;
}

Package SEE;
import CIE.*;

public class External Extends Student {
    public int external[] = new int[5];
}
```

```
import java.util.Scanner;
import CIE.*;
import SEE.*;

public class marks {
    public static void main(String args[]) {
        int n;
        Scanner sc = new Scanner(System.in);
        System.out.println(" Enter number of student");
        n = sc.nextInt();

        External student[] = new External[n];
        Student details[] = new Student[n];
        int final marks[][] = new int[n][5];

        for(int i=0; i<n; i++)
        {
            student[i] = new External();
            details[i] = new student();
            System.out.println("Enter student USN and a sem");
            details[i].USN = sc.nextInt();
            details[i].sem = sc.nextInt();
            System.out.println(" Enter Internal marks of subject respectively");
            for(int j=0; j<5; j++) {
                student[i].internal[j] = sc.nextLine();
            }
            System.out.println("Enter External marks of Subject");
            for(int k=0; k<5; k++) {
                student[i].External[k] = sc.nextInt();
```

```
for(int i=0; i<n; i++)
{
    for(int j=0; j<5; j++){
        final_marks[i][j] = student[i].internal[j] +
        (int) student[i].External[j] /2 );
    }
}
for(int j=0; i<n; i++) {
    System.out.println("USN: "+details[i].USN);
    System.out.println("sem "+details[i].sem);
    System.out.println(" marks of the student ");
    for(int j=0; j<5; j++){
        System.out.println("final_marks[i][j];
    }
}
```

Output:

Enter number of students
2

Enter Student USN and Sem
227 3
Enter Internal marks of 5 Subjects
34
33
32
28
40

Enter External marks of 5 Subjects
45
67
87

Enter Student USN and Sem
226
3
Enter Internal marks of 5 subjects
30
28
34
40
43

Enter External marks of 5 subjects
84
87
76
65
34

USN: 227
Sem: 3
Marks of Student is
56
66
75
77
79

USN: 226
Sem: 3
Marks of Student is
74
71
72
72
70

Output:

```
C:\Users\ sushanth \OneDrive\Desktop\Java Notepad>java Marks
enter number of students
1
Enter Student 1 name and usn and sem respectively
Subhash
cs221
2
Enter Internal marks of 5 subject in respective order
23
34
45
43
32
Enter external marks of 5 subject in respective order
56
67
78
89
90
Name: Subhash
USN: cs221
Sem: 2
Marks of the student 1 is
51
67
84
87
77
```

LAB-PROGRAM-9:

Demonstrate Inter process Communication and deadlock

```java
class printer{
      String str;
      printer()
      {
            str="";
      }
      synchronized void print(String str)
      {
            System.out.print("["+str);
            try {
                  Thread.sleep(1000);
            }catch(InterruptedException e)
            {
                  System.out.println("Error occured");
            }
            try {
                  System.out.println("]");
                  Thread.sleep(1000);
            } catch (InterruptedException e) {
                  // TODO Auto-generated catch block
                  e.printStackTrace();
            }

      }
}
class SampleThread implements Runnable
{
      String msg;
      printer pt;
      Thread t;
      public SampleThread(printer pr,String message)
      {
```

```java
            pt=pr;
            msg=message;
            t=new Thread(this);
            t.start();
    }

    @Override
    public void run() {
            // TODO Auto-generated method stub
            pt.print(msg);
    }

}
public class InterThread {
    public static void main(String[] args) {
            printer pt=new printer();
            SampleThread s1=new SampleThread(pt,"HELLO");
            SampleThread s2=new SampleThread(pt,"CSE");
            SampleThread s3=new SampleThread(pt,"WORLD");
            SampleThread s4=new SampleThread(pt,"BMS");

            try {
                    s1.t.join();
                    s2.t.join();
                    s3.t.join();
                    s4.t.join();
            } catch (InterruptedException e) {
                    // TODO Auto-generated catch block
                    e.printStackTrace();
            }
    }
}
```

```
PS C:\Users\HP\Desktop\dark mode> cd "c:\Users\HP\Downloads\"
[HELLO]
[BMS]
[WORLD]
[CSE]
PS C:\Users\HP\Downloads>
```