

Java lab programs:

Book database

import java.util.Scanner;

class Books

```
    String name;  
    String author;  
    int price;  
    int numPages;  
    Books () { }
```

Books (String name, int price, int numPages)

```
    this.name = name;  
    this.author = author;  
    this.price = price;  
    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;  
    return name + author + price + numPages;
```

```
}
```

```
}
```

~~class Books {~~

```
public static void main( String args[] )  
{  
    Scanner s = new Scanner( System.in );  
    int n;  
    String name;  
    String author;  
    int price;  
    int numPage;  
    System.out.println( "Enter the number of books:" );  
    n = s.nextInt();  
    Book b[];  
    b = new Book[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 the price:" );  
        price = s.nextInt();  
        System.out.print( "Enter no of pages:" );  
        numPage = s.nextInt();  
        System.out.print( "Enter" );  
        b[i] = new Book( name, author, price,  
                         numPage );  
    }  
    for (int i = 0; i < n; i++)  
    {  
        System.out.print( "Book" + (i+1) + " " +  
                          b[i] );  
    }  
}
```

{

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

```
    System.out.println ("Book" + (i+1) + ":" + b[i]);
```

{

{

Output :

Enter the number of book : 2

Book 1:

Enter the name of the book : Jungle Book

Enter the author of the book : Rudyard Kipling

Enter the price of the book : 1000.

Enter the number of the book : 400

Book 2:

Book name : Jungle Book Tales of akbar and khan

Enter the name of the book : Tales of akbar & khan

Enter the author of the book : Babul

Enter the price of the book : 900

Enter the number of pages of the book : 200

```
public double calculatePercentage() {  
    int totalMarks = 0;  
    for (int mark : marks) {  
        totalMarks += mark;  
    }  
    return (double) totalMarks / marks.length;
```

```
public void displayDetails()
```

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

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

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

```
for (int i=0; i < marks.length; i++)
```

```
System.out.print("Subject " + (i+1) + ":" + marks[i] + " ");
```

```
}  
System.out.println()
```

```
System.out.println("Percentage: " + calculatePercentage() + "%");
```

```
}
```

Student

classmate

Date _____
Page _____

public class StudentRun {

public static void main(String[] args) {

Scanner scanner = new Scanner(System.in);
System.out.println("Enter the number of students");
int numStudents = scanner.nextInt();

Student[] students = new Student[numStudents];
for (int i = 0; i < numStudents; i++) {

System.out.println("Enter details for
Student " + (i + 1) + ":");

students[i] = new Student("", "");
students[i].acceptDetails();

}

for

System.out.println("Enter details of student");

for (Student student : students) {

student.displayDetail();

System.out.println("-----");

}

}

Output :

Enter USN : 12

Enter Name : 9

Enter marks for Subject 1 : 2

Enter marks for Subject 2 : 3

Enter marks for Subject 3 : 4

Enter marks for Subject 4 : 5

Enter marks for Subject 5 : 6

Enter marks for Subject 6 : 7

import java.util.Scanner;

public class Quadratic {

public static void main (String [] args) {

Scanner scanner = new Scanner (System.in);

System.out.print ("Enter coefficient a:");
double a = scanner.nextDouble();

System.out.print ("Enter coefficient b:");

System.out.print ("Enter coefficient c:");
double b = scanner.nextDouble();

System.out.print ("Enter coefficient c:");
double c = scanner.nextDouble();

double discriminant = b * b - 4 * a * c

if (discriminant > 0) {

double root1 = (-b + Math.sqrt(discriminant))

double root2 = (-b - Math.sqrt(discriminant))

System.out.println ("Root 1: " + root1);

System.out.println ("Root 2: " + root2);

}

else if (discriminant == 0) {

double root = -b / (2 * a);

System.out.println ("Root: " + root);

} else {

double realPart = -b / (2 * a);

double imaginaryPart = Math.sqrt (-discriminant) / (2 * a);

System.out.println ("Root1:" + realPart1 + " + " + imaginaryPart1 + "i");
System.out.println ("Root2:" + realPart2 + " - " + imaginaryPart2 + "i");

{

{

Output

Enter the coefficients of a, b, c

1

2

1

Roots are real and equal

$$\text{root1} = \text{root2} = -1.0$$

Two roots
are equal

Two roots
are equal

Two roots
are equal

Two roots
are equal

2.

{ class Bank

{ person (string, [] array)

savings account savingsAccount = new

savings Account ("Hema P", "SA1001");

current Account currentAccount = new

currentAccount ("Hanitha B", "CA2002");

savingsAccount.deposit(5000);

savingsAccount.displayBalance();

savingsAccount.computeInterest();

savingsAccount.withdraw(2000);

" ", displayBalance();

currentAccount.deposit(8000);

" ", displayBalance();

" ", withdraw(5000);

" ", displayBalance();

{ class Account

{

 protected string ~~CustomerName~~;

string accountNumber;

double balance;

public Account (string customerName, int accountNumber)

{

 this.customerName = customerName;
 ". accountNumber = accountNumber;
 this.balance = 0;

}

public void deposit (double amount)

 balance += amount;
 rep ("Deposit of \$" + amount);

}

public void display Balance ()

 rep ("Account Number: " + accountNumber +
 " In Balance \$");

}

class savings Account extends Account

{

 public savings Account (string customerName, int accountNumber)

{

 super (customerName, accountNumber); }

 public void compute Interest()

{

```
double interestRate = 0.05;
```

```
double interest = balance * interestRate
```

```
balance += interest;
```

```
rep ("Interest of $" + interest));
```

{

```
public void withdraw (double amount)
```

{

```
if (balance >= amount)
```

```
balance -= amount;
```

```
System.out.println ("Withdraw Succeeded");
```

```
UpdateBalance ();
```

```
CheckMinimumBalance();
```

{

```
else
```

{

```
System.out.println ("Insufficient funds. Withdraw Failed!");
```

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

{

class Bank

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

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

```
        System.out.println("Enter customer name for savings  
        Account: ");
```

```
        String SCN = si.nextLine();
```

```
        System.out.print("Enter amount number account  
        number for savings Account: ");
```

```
        long SAN = si.nextLong();
```

```
        System.out.print("Enter initial balance for savings Account:  
        ");
```

```
        double SIB = si.nextDouble();
```

```
        SavingsSA SA = new SavingsSA(SCN, SAN, SIB);
```

```
        System.out.print("Enter customer name for current  
        Account: ");
```

```
        String CNN = si.nextLine();
```

```
        System.out.print("Enter account number for current  
        Account: ");
```

```
        long CAN = si.nextLong();
```

```
        System.out.print("Enter initial balance for  
        current Account: ");
```

```
        double MB = si.nextDouble();
```

```
        System.out.print("Enter foreign exchange  
        for current account: ");
```

~~double FC = SC = si.~~

double SC = SI.nextDouble();

curr Acct CA = new curr Acct(C, CN, CAN, CIB);

System.out.print("Enter deposit amount for savings account");

double SDA = M.nextDouble();

SA.deposit(SDA);

System.out.print("Enter the interest rate for
Savings Account: ");

double SIR = M.nextDouble();

SA.computeAndDepositInterest(SIR);

CA.displayBalance();

3

out put -

Enter customer name for savings Account:

Enter account number for Savings Account: 100

Enter Initial balance for Savings Account: 1000

Enter customer name for current Account:

Deekshitha

Initial balance for current account

Enter initial balance for current account

(Lakhs Rupees) 100

Program to calculate total marks of a student
demonstrating packages.

Student.java.

```
package CIE;  
public class Student {  
    public String USN;  
    public String Name;  
    public int sem;
```

```
public Student (String USN, String Name, int sem)
```

{

```
    this.USN = USN;
```

```
    this.Name = Name;
```

```
    this.sem = sem;
```

}

}

Internals.java.

```
package CIE;  
public class Internals extends Student {  
    public int [ ] internalMarks;
```

Date _____
Page _____ 43

public Internals (String USN, String name,
int age, int[] internalMarks) {
super (USN, name, age);
this. internalMarks = internalMarks;

{
}

External.java.

```
package SEE;  
import CIE.Student;  
public class External extends Student {  
    public int[] externalMarks;  
    public External (String USN, String name,  
    int age, int[] externalMarks) {  
        super (USN, name, age);  
        this. externalMarks = externalMarks;
```

Main.java.

```
import java.util.Scanner;  
import CIE.Internals;  
import SEE.Student;
```

public class Main {

public static void main (String args[])

X

Scanner sc = new Scanner (System.in);

System.out.println ("Enter number of student");
int n = sc.nextInt();

Internal []

internalData = new Internal [n];
External [] externalData = new External [n];

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

{

System.out.println ("Enter the details for
student " + (i + 1) + " ");
Scanner sc = new Scanner (System.in);
String S.N = sc.nextLine();
System.out.println ("Enter name");
String name = sc.nextLine();

System.out.println ("Enter Semester");

int sem = sc.nextInt();

for (int j = 0; j < 5; j++) {

System.out.println ("Subject " + (j + 1) + " ");
internalMarks [i] = sc.nextInt();

System.out.println ("Enter the Marks for
subject");

for (int j = 0; j < 5; j++) {

System.out.println("Subsidy" + (j+1) + " ")
externalMarks[i] = n.readInt();

internalData[i] = n.readInt();
n.readInternalMarks();
externalData[i] = n.readExternal();
n.readExternalMarks();

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

System.out.println("In Student" + (i+1) + " ");
System.out.println("Internal Marks" +
arrayTostring(internalData[i]), internalMarks)
System.out.println("External Marks" +
arrayToString(externalData[i]), externalMarks);

System.out.println("Total marks (ie + SEG)");
calculateTotalMarks(internalData[i], internalMarks,
externalData[i], externalMarks);

}

Scanner.close();

}

{ private static String arrayTostring (int[] arr)

Stringbuilder sb = new StringBuider();
 sb.append("E");
 sb.append("I");

```
for (int i=0; i<array.length; i++) {
    sb.append(array[i]);
    if (i == array.length - 1) {
        sb.append("I");
    } else {
        sb.append("E");
    }
}
return sb.toString();
}
```

Output.

Enter number of student

-1
 Enter details for student 1

Enter USN: IBM 22 CS 104

Enter Name : ~~Constituents~~ HITHE54

Enter Section : 3

Enter Internal Marks

Subject 1 40

Subject 2 45

Subject 3 49

~~Subject 4~~ 43

~~Subject 5~~ 49

~~Enter External Marks~~

subject 1 40

subject 2 42

Subject 3 43
Subject 4 42
Subject 5 43

Studen 1

External marks

External marks.

Total marks (CIE + SEE):

Thread.

class BMS implements Runnable {

public void run() {

try {

while (true) {

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

Thread.sleep(10000);

}

}

} catch (InterruptedException e) {

System.out.println("The program interrupted
or closed");

}

}

class CSE implements Runnable {

public void run() {

try {

while (true) {

System.out.println("CSE");

Thread.sleep(2000);

}

} catch (InterruptedException e) {

~~System.out.println("There is an interrupt
occurred");~~

5

p.

public class Main {
 public static void main (String args) {

BMS bns = new BMS();

Thread t1 = new Thread(bns);
t1.start();

CSE cse = new CSE();

Thread t2 = new Thread(cse);
t2.start();

Output

CSE

CSE

CSE

CSE

CSE

BMS College of Engineering

CSE

CSE

CSE

CSE

CSE

Exception Handling.

Import java.util.Scanner;

class WrongAge extends Exception {

 public WrongAge (String message) {
 super (message);
 }

}

class Father {

 private int age;

 public Father (int age) throws WrongAge {

 if (age < 0) {

 throw new WrongAge ("Age cannot be
 negative");

}

 this.age = age;

}

 public int getAge () {

 return age;

}

}

public class ExceptH {

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

 Father father = new Father (40);

 Son son = new Son (father.getAge(),
 56);

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

```
    catch (WrongAge e) {
        }
```

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

Output
Exception: Son's age should be less than Father's age.

Done
2/3/24