

12/11/2023

```

Fibonacci Series;
class Main {
    public static void main (String [] args) {
        int n = 10, firstTerm = 0, secondTerm = 1;
        System.out.print (" Fibonacci Series till ");
        " +n " terms: " );
        for (int i = 1; i <= n; ++i) {
            System.out.print (firstTerm + ", ");
            int nextTerm = firstTerm + secondTerm;
            firstTerm = secondTerm;
            secondTerm = nextTerm;
        }
    }
}

```

Output:

```

Fibonacci series till 10 terms: 0, 1, 2, 3, 5, 8, 13,
21, 34
-----
class Main {
    public static void main (String [] args) {
        char operator;
        Double number1, number2, result;
        Scanner input = new Scanner (System.in);
        System.out.println ("Choose an operator: +, -, *, or /");
        operator = input.next ().charAt (0);
        System.out.println ("Enter first no.");
        number1 = input.nextDouble ();
        System.out.println ("Enter second no.");
        number2 = input.nextDouble ();
        switch (operator) {
            case '+': result = number1 + number2;
                System.out.println (result);
                break;
            case '-': result = number1 - number2;
                System.out.println (result);
                break;
            case '*': result = number1 * number2;
                System.out.println (result);
                break;
            case '/': result = number1 / number2;
                System.out.println (result);
                break;
            default: System.out.println ("Invalid operator");
        }
        input.close();
    }
}

```

~~Output : choose an operator: +, -, \*, /, or %~~

~~+ prints the sum of two numbers~~

~~- prints the difference of two numbers~~

~~\* prints the product of two numbers~~

~~/ prints the quotient of two numbers~~

~~% prints the remainder of two numbers~~

~~3.0 prints the first number to the console~~

~~first choose a number from 1 to 100~~

⑪ Choose an operator: +, -, \*, /, or %

Enter first number

2

Enter second number

1

~~2.0 + 1.0 = 2.0~~

~~public class Main {~~

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

~~int num = 5;~~

~~boolean flag = false;~~

~~for (int i = 2; i < num; i++) {~~

~~if (num % i == 0) {~~

~~flag = true;~~

~~break;~~

~~} if (flag == false)~~

~~System.out.println("num is prime-no.");~~

~~else~~

~~System.out.println("num is not prime-no.");~~

~~}~~

~~System.out.println("num is prime-no.");~~

~~}~~

~~Output: 5 is a prime number.~~

⑫ 10 is not a prime number.

~~public class HelloWorld {~~

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

~~System.out.println("Hello World!");~~

~~}~~

~~3.0 Hello World!~~

Output:

Hello World!

Top 11/12/23

```
import java.util.Scanner;
class grocery
```

```
{ grocery (double a, double b, double c)
```

```
{ double dal = a;
```

```
double pulses = b;
```

```
double sugar = c;
```

```
{ grocery (double a)
```

```
{ double dal = a;
```

```
double pulses = a;
```

```
double sugar = a;
```

```
{ grocery ()
```

```
{ double dal = 1;
```

```
double pulses = 1;
```

```
double sugar = 0.5;
```

```
{ grocery (obj)
```

```
{ double dal = obj.dal;
```

```
double pulses = obj.pulses;
```

```
double sugar = obj.sugar;
```

```
{
```

```
void total()
```

```
{ sop ("Total is : ");
```

```
sop ( Dal * 150 ) + ( pulses * 80 ) + ( sugar * 50 );
```

```
class Run
```

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

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

```
sop ("Enter quantity of dal: ");
```

```
double a = s.nextDouble();
```

# Lab-3

import java.util.Scanner;

```

class Quadratic {
    int a, b, c;
    double r1, r2, d;
    void getd() {
        Scanner s = new Scanner(System.in);
        System.out.println("Enter the coefficients of a, b, c");
        a = s.nextInt();
        b = s.nextInt();
        c = s.nextInt();
    }
    void compd() {
        while (a == 0) {
            System.out.println("Not a quadratic equation");
            zero_value();
        }
        Scanner s = new Scanner(System.in);
        d = b * b - 4 * a * c;
        if (d == 0) {
            r1 = (-b) / (2 * a);
            System.out.println("Roots are real and equal");
            System.out.println("Root 1 = Root 2 = " + r1);
        } else if (d > 0) {
            r1 = ((-b) + Math.sqrt(d)) / (double) (2 * a);
            r2 = ((-b) - Math.sqrt(d)) / (double) (2 * a);
            System.out.println("Roots are real and distinct");
            System.out.println("Root 1 = " + r1 + "Root 2 = " + r2);
        } else if (d < 0)
    }
}

```

```

system.out.println("roots are imaginary")
r1 = (-b)/(2*a);
system.out.println("root = " + r1 + " + " + r2);
system.out.println("root = " + r1 + " - " + r2);
}
}
class QuadraticMain
{
public static void main(String args[])
{
Quadratic q = new Quadratic();
q.get();
q.compute();
}
}

```

Output 1

a : 1

b : 9

c : 1

The roots are -0.11251 and -8.887

The roots are distinct and real

Output 2

a : 1

b : 2

c : 1

The root is -1.0

The roots are real and equal

Output 3

a : 1

b : 1

c : 1

Roots are not real

}

→ Book Database

```
import java.util.Scanner;  
class Books  
{  
    String name;  
    String author;  
    int price;  
    int numPages;  
    Books () {}  
    Books (String name, String author,  
           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;  
        author = "Author name:" + this.author;  
        price = "Price:" + this.price;  
        numPages = "Number of pages:"  
                  + this.numPages;  
        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 numPages;
```

```

    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.print("Enter name
of book : ");
name = s.nextLine();
system.out.print("Enter author : ");
author = s.nextLine();
system.out.print("Enter price : ");
price = s.nextInt();
system.out.print("Enter no. of
pages : ");
numPages = s.nextInt();
b[i] = new Books(name, author,
price, numPages);
}
for (int i = 0; i < n; i++) {
    system.out.print("Book: " +
(i + 1) + " = " + b[i]);
}
}

```

Output :

```

Enter the number of book : 2
Book 1
Enter Name : Jungle-book
Enter Author : Rudyard- kipling
Enter Price : 100
Enter Number of pages : 500
Book 2
Enter Name : Harry Potter
Enter Author : S K Rowling
Enter Price : 900
Enter Number of pages : 613

```

Book 1:

Name : Jungle - Book King  
Author: Rudyard - King  
Price: 1000  
No. of pages: 5000

Book 2:

Name: Harry - Potter  
Author JK - Rowling  
Price: 900  
No. of pages: 613

✓ 08/01/24

```

import java.util.Scanner
class student
{
    String USN;
    String name;
    int marks [] = new int [6];
    void accept Details()
    {
        Scanner sc = new Scanner (System.in);
        System.out.println("Enter USN:");
        USN = sc.nextLine();
        System.out.println("Enter name");
        name = sc.nextLine();
        System.out.println("Enter marks of 6 subjects");
        for (int i = 0; i < 6; i++)
        {
            marks [i] = sc.nextInt();
        }
    }
    void calculate Percentage()
    {
        int total Marks = 0;
        mark: marks
        for (int i = 0; i < 6; i++)
        {
            sum total Marks +
            percentage = (total Marks / 600) * 100;
            return percentage;
        }
        System.out.println("Enter student details");
        student [i].accept Details();
        student [i].calculate Percentage();
        System.out.println("Student Details");
        for (student student: students)
        {
            student.display Details();
        }
    }
    void display Details()
    {
        System.out.println("USN:" + USN);
        System.out.println("Name:" + name);
        System.out.println("Marks:" + marks [0] + ". " + marks [1] +
                           ", " + marks [2] + marks [3] + marks [4] + ", " +
                           marks [5]);
        System.out.println("Percentage:" + percentage + "%");
    }
}

```

→ Output

Enter number of Students: 2

Enter USN: 1bm22cs001

Enter name: chain

Enter marks for 6 subjects:

98

99

99

99

99

98

Enter USN: 1bm22cs002

Enter name: claruX

Enter marks for 6 subjects:

90

95

98

99

99

90

Student Details:

USN: 1bm 22cs001

Name: Kurt

Marks: 99, 99, 99, 99, 100, 78

Percentage: 93.5 %

USN: 1bm22cs002

Name: clara

Marks: 90, 95, 98, 85, 80, 82

Percentage: 90.0 %

## 1 Area

```

abstract class Shape {
protected int length;
protected int breadth;
public Shape (int length, int breadth) {
    this.length = length;
    this.breadth = breadth;
}
public abstract void printArea();
}

class Rectangle extends Shape {
public Rectangle (int length, int breadth) {
    super (length, breadth);
}
@Override
public void printArea() {
    int area = length * breadth;
    System.out.print("Rectangle Area: " + area);
}

class Triangle extends Shape {
public Triangle (int length, int breadth) {
    super (length, breadth);
}
@Override
public void printArea() {
    double area = 0.5 * length * breadth;
    System.out.print("Triangle Area: " + area);
}

class Circle extends Shape {
public Circle (int radius) {
    super (radius, 0);
}
@Override
public void printArea() {
    double area = Math.PI * radius * radius;
    System.out.print("Circle Area: " + area);
}

public class Main {
public static void main (String [] args) {
    Rectangle rectangle = new Rectangle(50);
}

```

```
Triangle triangle = new Triangle(4, 4);  
Circle circle = new Circle(6);  
rectangle.printArea();  
triangle.printArea();  
circle.printArea();
```

{  
3  
Output:

```
Rectangle Area: 50  
Triangle Area: 16.0  
Circle Area: 113.0973
```

Ais  
22/01/24

```
Bank Account
import java.util.Scanner;
class Account {
    String customerName;
    long accountNumber;
    long balance;
    double balance;
    public Account (String customerName,
                    long accountNumber, String accountType,
                    long balance)
    {
        this.customerName = customerName;
        this.accountNumber = accountNumber;
        this.accountType = accountType;
        this.balance = balance;
    }
    public void deposit (double amount)
    {
        balance += amount;
        System.out.println ("Deposit
successful. Update Balance:" + balance);
    }
    public void displayBalance ()
    {
        System.out.println ("Acc no. " + accountNumber);
        System.out.println ("Customer Name: " + customerName);
        System.out.println ("Account Type: " + accountType);
        System.out.println ("Balance: " + balance);
    }
}
class SavAcct extends Account
{
    public SavAcct (String customerName,
                    long accountNumber, double balance)
    {
        super (customerName, accountNumber,
               accountType : "Savings", balance);
    }
    public void computeAndDepositInterest (
        double rest)
    {
        if (amount <= balance)
        {
            balance += rest;
        }
    }
}
```

balance - = amount;  
sop ("With drawal successful. Update  
balance: " + balance);  
}  
else { sop ("Insufficient funds.  
With drawal failed");  
}}  
  
class CurrAcct extends Account  
{  
 double minBalance;  
 double serviceCharge;  
 public CurrAcct (String customerName,  
 long accountNumber) {  
 super (customerName, accountNumber,  
 "Current");  
 this.minBalance = 1000; mininumBalance;  
 this.serviceCharge = serviceCharge;  
 }  
 private void ~~check~~mininumBalance ()  
 { if (balance < mininumBalance)  
 { balance - = serviceCharge;  
 sop ("Minimum balance  
Service charge not maintained.  
balance" + balance);  
 }  
 public void withdraw (double amount)  
 { if (amount <= balance)  
 { balance - = amount;  
 Update balance: " + ~~success but~~  
 check mininum Balance());  
 }  
 else { sop ("Insufficient funds. With drawal  
failed");  
 }  
}

```
public class Bank
{
    public static void main String[] args)
    {
        Scanner s1 = new Scanner (System.in);
        System.out ("Enter customer name for
        savings Acc");
        String SCN = s1.nextLine();
        String SCN = "Enter account no. for saving
        account : ";
        long SAN = s1.nextLong();
        System.out ("Enter initial balance for
        saving account");
        double SIB = s1.nextDouble();
        Sav Acct SA = new SavAcct (SCN, SAN, SIB);
        System.out ("Enter customer name for current
        account");
        String CCN = s1.nextLine();
        System.out ("Enter account number for
        current : ");
        long CAN = s1.nextInt();
        System.out ("Enter initial balance for
        current");
        double CIB = s1.nextDouble();
        System.out ("Enter min balance for current");
        double MB = s1.nextDouble();
        Curr Acct CA = new currAcct (CCN, CAN,
        CIB, MB, SC);
        System.out ("Enter deposit amount for
        savings Account");
        double SDA = s1.nextDouble();
        SA.deposit (SDA);
        System.out ("Enter interest rate for Saving Acc");
        double SIR = s1.nextDouble();
        SA.compute And Deposit Interest (SIR);
        System.out ("Enter deposit amount for
        current Account");
```

double CNA = SI.next Double  
CA. withdraw (CNA);

SOP("In Final Balances : ");

SOP("Saving Account : ");

SA. display Balance();

SOP("In Current Account : ");

CA. display Balance(); } }

Output :

Enter customer name for Savings: Ray  
Enter account no. " : 202113

Enter initial balance " : 10000

Enter customer name for Current: Chaka

Enter account no. " : 20213

Enter initial balance " : 2000

Enter minimum balance " : 5000

Deposit Successful

updated balance : 15000

Enter interest rate for SA : 4

Interest computed and deposited

updated balance : 15600.0

Enter withdrawal amt

withdrawal successful for SA : 200

updated balance : 14900.0

Package      22/1/2024

package 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;
        }
    }

```

package CIE;

```

public class Internals extends student
{
    public int[] internal
    marks;
    public Internals(string USN,
    String name, int sem,
    int[] internal marks)
    {
        super(USN, name, sem);
        this.internalMarks = internalMarks;
    }
}

```

super(USN, name, sem);  
this.internalMarks = internalMarks;

package SEE;

```

import CIE.student;
public int[] see Marks;
public External(string USN,
String name, int sem,
int[] SEE marks)
{
    super(USN, name, sem);
    this.seeMarks = seeMarks;
}

```

```

import CIE.internals;
import SEE.External;
public class Main
{
    public static void main(String[] args)
    {
        int[] internalMarks = { 75,
                               80, 90, 85, 88 };
        InternalStudent student1 = new InternalStudent("IABC123", "John", 3, internalMarks);
        int[] seeMarks = { 70, 85, 88, 92, 78 };
        ExternalStudent student2 = new ExternalStudent("ZXY42456", "Jane", 3, seeMarks);
        System.out.println("Student 1:");
        System.out.println("USN: " + student1.USN);
        System.out.println("Name: " + student1.name);
        System.out.println("Semester: " + student1.sem);
        System.out.println("Internal Marks: " + Arrays.toString(student1.internalMarks));
        System.out.println("Student 2:");
        System.out.println("USN: " + student2.USN);
        System.out.println("Name: " + student2.name);
        System.out.println("Semester: " + student2.sem);
        System.out.println("SEE Marks: " + Arrays.toString(student2.seeMarks));
    }
}

```

Output: Student 1:  
Student 1: USN: 123 Name: John  
USN: 123 Name: John  
Name: John  
Semester: 3 Internal marks (75, 80, 90, 85, 88)

Student 2:

USN: 2456 Name: James

Name: James

Semester: 3 Internal marks (70, 85, 92, 78)

SEE marks = [70, 85, 92, 78]

{(a) student information data}

{(b) student marks data}

{(c) student marks average}

{(d) student marks total}

{(e) student marks count}

{(f) avg}

{(g) sum}

{(h) count)}

{(i) student information data}

{(j) student marks data}

Final code output

program main block starts adding

new student information block

{(l) student information data}

{(m) new student data for student}

{(n) calculate}

{(o) print}

19/02/2024 Lab - Programs - 5  
 Write a pgm which creates two threads, one thread displaying "BMS College of Engineering" once every ten seconds and another displaying "CSE" once every second.

```

class BMS implements Runnable {
  public void run() {
    while (true) {
      try {
        System.out.println("BMS College of Engg");
        Thread.sleep(10000);
      } catch (InterruptedException e) {
        e.printStackTrace();
      }
    }
  }
}

class CSE implements Runnable {
  public void run() {
    while (true) {
      try {
        System.out.println("CSE");
        Thread.sleep(2000);
      } catch (InterruptedException e) {
        e.printStackTrace();
      }
    }
  }
}

public class Main {
  public static void main(String[] args) {
    Thread t1 = new Thread(new BMS());
    Thread t2 = new Thread(new CSE());
    t1.start();
    t2.start();
  }
}
  
```

Output: BMS college of Engg

and CSE

CSE

CSE

CSE

CSE

CSE

BMS college of Engg

CSE

CSE

CSE

CSE

CSE

BMS college of Engg

CSE

CSE

CSE

CSE

CSE

BMS college of Engg

group x 3

Engineering & B.A group

Write a program that demonstrates handling of exception in inheritance tree. Create a base class called "Father" and derived class called "son" which extends base class. In Father class, implement a constructor which takes the age and throws the exception Wrong Age when input age less than 0. In son class implement a constructor that cases both Father and son's age and throws an exception if son's age is  $\geq$  father's age.

```
import java.util.Scanner;
class WrongAge extends Exception;
public WrongAge(String message)
{
    super(message);
}
```

```
class Father {
    int fatherAge;
    Father() throws WrongAge {
        Scanner s = new Scanner(System.in);
        System.out.println("Enter Father's Age:");
        fatherAge = s.nextInt();
        if (fatherAge < 0)
            throw new WrongAge("Age cannot be negative");
    }
}
```

```
void display()
{
    SOP("Father age = " + fatherAge);
}

class son extends father
{
    int sonAge;
    son() throws WrongAge
    {
        super();
        Scanner s = new Scanner(System.in);
        SOP("Enter son's age : ");
        sonAge = s.nextInt();
        if (sonAge > FatherAge)
        {
            throw new WrongAge("son's age cannot be greater than father's age");
        }
        else if (sonAge == FatherAge)
        {
            throw new WrongAge("son's age cannot be equal to Father's age");
        }
        else if (sonAge < 0)
        {
            throw new WrongAge("Age cannot be negative");
        }
    }

    void display()
    {
        super.display();
        SOP("son's age is : " + sonAge);
    }
}

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

```
try {
    sons = new Son();
    s. display();
}
catch (Wrong Age)
{
    SOP( e. get message() );
}
```

Output:

Enter Father's age: 40

Enter son's age: 18

Father's age is: 40

Son's age is: 18

Enter Father's age: 30

Enter son's age: 30

Son's age cannot be equal to  
Father's age.

Enter Father's age:- 20

Age cannot be negative

*Jit  
19.02.24*