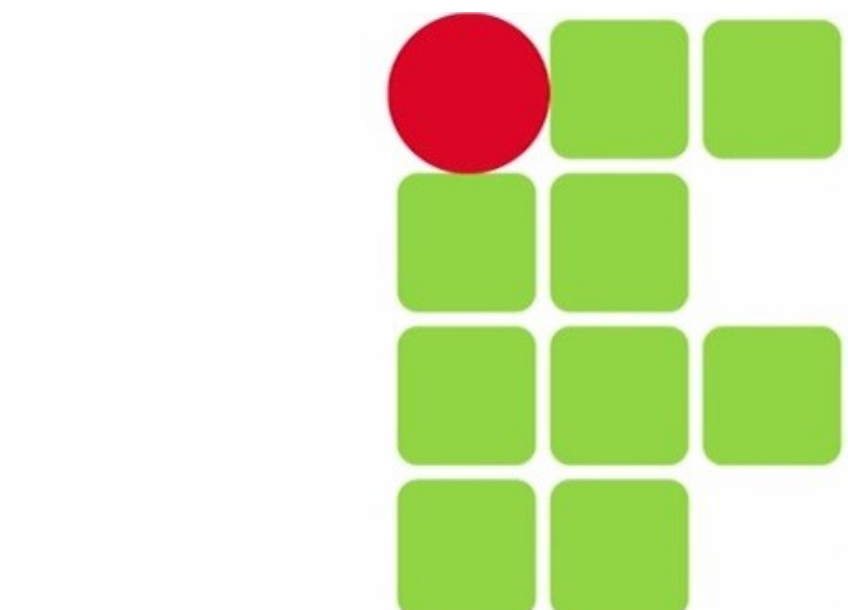


Instituto Federal São Paulo
IFSP
Campus Cubatão

Linguagem de Programação II



FABIO WILLIAM CONCEIÇÃO
ANALISE E DESENVOLVIMENTO DE SISTEMAS
TURMA 481
RA 144091-8

1 – Calculadora básica

Nesse programa faremos uma simples calculadora, onde a classe Calculator é a responsável pelos metodos da calculadora (adição, subtração, divisão e multiplicação). O usuario irá digitar 2 números e depois é chamada cada uma das operações.

```
package calculator;

import java.util.Scanner;

class Calculator{

    // sum
    public static void add(int n1, int n2){
        int result = n1 + n2;
        System.out.println(n1 + " + " + n2 + " = " + result);
    }

    //subtract
    public static void subtract(int n1, int n2){
        int result = n1 - n2;
        System.out.println(n1 + " - " + n2 + " = " + result);
    }

    //multiply
    public static void multiply(int n1, int n2){
        int result = n1 * n2;
        System.out.println(n1 + " * " + n2 + " = " + result);
    }

    //divide
    public static void divide(double n1, double n2){
        double result = n1 / n2;
        System.out.println(n1 + " / " + n2 + " = " + result);
    }
}
```

```
public class main {

    public static void main(String[] args) {
        // TODO Auto-generated method stub
        int n1 = 0;
        int n2 = 0;
        double dn1 = 0;
        double dn2 = 0;
        Calculator calculator = new Calculator();

        Scanner sc = new Scanner(System.in);

        String entrada;

        System.out.println("Digite o primeiro numero: ");
        entrada = sc.nextLine();
        n1 = Integer.parseInt(entrada);
        dn1 = Double.parseDouble(entrada);

        System.out.print("Digite o segundo número: ");
        entrada = sc.nextLine();
        n2 = Integer.parseInt(entrada);
        dn2 = Double.parseDouble(entrada);

        calculator.add(n1, n2);
        calculator.subtract(n1, n2);
        calculator.multiply(n1, n2);
        calculator.divide(dn1, dn2);

    }
}
```

2 – Gerar um arquivo .txt (o arquivo fica na pasta do projeto)

Nesse programa pegaremos o que o usuário digita na tela e salvaremos dentro um arquivo .txt e o mesmo ficará salvo na pasta onde está rodando o código atual.

```
package generateTXTFile;

import java.io.BufferedReader;
import java.io.FileWriter;
import java.io.IOException;
import java.io.InputStreamReader;
import java.io.PrintWriter;

public class main {

    public static void main(String[] args) {
        // TODO Auto-generated method stub

        System.out.print("Digite algo");

        try {
            //create a buffered reader that connects to the console, we use it so we can read lines
            BufferedReader in = new BufferedReader(new InputStreamReader(System.in));

            //read a line from the console
            String lineFromInput = in.readLine();

            //create an print writer for writing to a file
            PrintWriter out = new PrintWriter(new FileWriter("output.txt"));

            //output to the file a line
            out.println(lineFromInput);

            //close the file (VERY IMPORTANT!)
            out.close();
        } catch (IOException e1) {
            System.out.println("Error during reading/writing");
        }
    }
}
```

3 – Ler o arquivo .txt que foi gerado no exercício anterior (coloque o txt na pasta raiz do projeto)

Aqui pegaremos o arquivo .txt que foi gerado no programa anterior e iremos ler o conteúdo do mesmo e imprimi-lo no console.

```
package readTXTFile;

import java.io.BufferedReader;
import java.io.FileInputStream;
import java.io.FileNotFoundException;
import java.io.IOException;
import java.io.InputStream;

public class main {

    public static void main(String[] args) throws IOException {
        // TODO Auto-generated method stub
        InputStream input = new BufferedInputStream(new FileInputStream("output.txt"));
        byte[] buffer = new byte[8192];

        try {
            for (int length = 0; (length = input.read(buffer)) != -1;) {
                System.out.write(buffer, 0, length);
            }
        } finally {
            input.close();
        }
    }
}
```

4 – Calcular média aritmética.

Um programa que calcula a média aritmetica de 4 alunos dadas que o mesmo utilizará 3 notas e depois colocará uma classificação para o aluno da letra 'A' até a letra 'E'.

```
import java.util.Scanner;

public class main {

    public static void main(String[] args) {
        // TODO Auto-generated method stub
        Scanner input = new Scanner(System.in);
        int n1, n2, n3;
        int media, i, cStuinput = 0;

        for(i = 0; i < 3; i++){

            cStuinput++;
            // receive 1º
            System.out.println("Aluno " + cStuinput + ", digite sua 1ª nota");
            n1 = input.nextInt();

            // receive a 2º
            System.out.println("Aluno " + cStuinput + ", digite sua 2ª nota");
            n2 = input.nextInt();

            // recebe a 3º nota
            System.out.println("Aluno " + cStuinput + ", digite sua 3ª nota");
            n3 = input.nextInt();

            // calcula a média
            media = (n1 + n2 + n3) / 3;
            System.out.println("A média do aluno " + cStuinput + " é " + media);

            // mostra a nota do aluno
            if( (media >= 0) && (media <4) ){
                System.out.println("Nota E");
            } else if(media < 5){
                System.out.println("Nota D");
            } else if(media < 7){
                System.out.println("Nota C");
            } else if(media < 8){
                System.out.println("Nota B");
            } else if(media <= 10){
                System.out.println("Nota A");
            }
        }
    }
}
```

5 – Sequencia fibbonaci

Na **matemática**, a **Sucessão de Fibonacci** (também **Sequência de Fibonacci**), é uma **sequência de números inteiros**, começando normalmente por 0 e 1, na qual, cada termo subsequente corresponde a **soma** dos dois anteriores. A sequência recebeu o nome do matemático italiano **Leonardo de Pisa**, mais conhecido por Fibonacci , que descreveu, no ano de 1202, o crescimento de uma população de coelhos, a partir desta. Tal sequência já era no entanto, conhecida na antiguidade.

Os números de Fibonacci são, portanto, os números que compõem a seguinte **sequência** (sequência **A000045** na **OEIS**):

0,1, **1, 2, 3, 5, 8, 13, 21, 34, 55, 89, 144, 233, 377, 610, 987, 1597, 2584, ...**

Fonte: https://pt.wikipedia.org/wiki/Sequ%C3%A2ncia_de_Fibonacci

```
package fibbonaci;

class Fibonacci {

    static long fibo(int n) {
        return (n < 2) ? n : fibo(n - 1) + fibo(n - 2);
    }
}
```

```

}

public class main {

    public static void main(String[] args) {
        // TODO Auto-generated method stub

        // print the first 30 terms
        for (int i = 0; i < 30; i++) {
            System.out.println("(" + i + "):" + Fibonacci.fibo(i) + "\t");
        }
    }
}

```

6 – Calcular hipotenusa

Calcular o valor da hipotenusa.

```

package hipotenusa;

import java.util.Scanner;

class Hipotenusa {

    Scanner read = new Scanner(System.in);

    private int cateto1;
    private int cateto2;
    private float hipotenusa;

    public Hipotenusa() {
        calculaHipotenusa();
    }

    private void calculaHipotenusa(){

        System.out.println("Cateto 1: ");
        cateto1 = Integer.parseInt(read.next());

        System.out.println("Cateto 2: ");
        cateto2 = Integer.parseInt(read.next());

        hipotenusa = (float) Math.sqrt(Math.pow(cateto1, 2) + Math.pow(cateto2, 2));

        System.out.println("Hipotenusa: " + hipotenusa);
    }
}

public class main {

    public static void main(String[] args) {
        // TODO Auto-generated method stub
        new Hipotenusa();
    }
}

```

Repositório git do trabalho: <https://github.com/Messhias/java-federal>