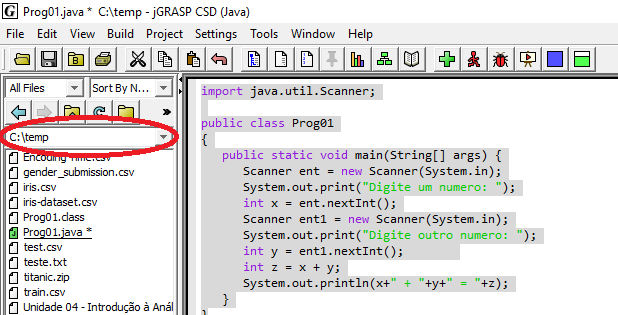
**Algumas dicas sobre o JGrasp**

1 – Você pode definir um diretório de trabalho onde você salvará os seus arquivos. No exemplo abaixo o diretório de trabalho foi configurado com c:\temp.



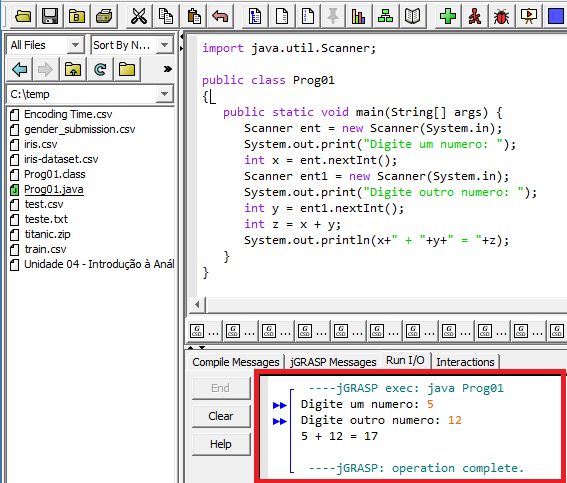
Ao terminar de digitar o seu programa, você deve salvá-lo e depois compilá-lo usando uma das seguintes opções:

* Selecionando **Build >> Compile**
* Pressionado CTRL + B
* Clicando no ícone 

Após compilar o seu programa, se não houver nenhum erro, o compilador do Java (javac) vai gerar um arquivo com o mesmo nome do seu arquivo fonte mas com a extensão .class. Agora você pode executar o seu programa. Para executá-lo você também tem algumas opções:

* Selecionando **Build >> Run**
* Pressionado CTRL + R
* Clicando no ícone 

A execução vai ocorrer na área logo abaixo do seu código.



**Entrada e saída de dados em Java**

Digite (perca um tempo digitando ao invés de simplesmente copiar e colar) o programa abaixo no JGrasp.

import java.util.Scanner;

public class Prog01

{

public static void main(String[] args) {

Scanner ent = new Scanner(System.in);

System.out.print("Digite um numero: ");

int x = ent.nextInt();

Scanner ent1 = new Scanner(System.in);

System.out.print("Digite outro numero: ");

int y = ent1.nextInt();

int z = x + y;

System.out.println(x+" + "+y+" = "+z);

}

}

Perceba que foram **instanciados** dois objetos Scanner (ent e ent1).

Altere o programa para que ele **instancie** apenas o objeto Scanner de nome **ent** e faça a leitura de x e y utilizando esse único objeto.

É realmente necessário criar um objeto Scanner para cada variável que for lida?

Resposta:

Agora você vai alterar o programa acima para que ele calcule e imprima, além da soma entre x e y (x+y), a subtração (x-y), a multiplicação (x\*y), o quociente (x/y), o resto (x%y) e a divisão. Uma vez que x e y são inteiros, x/y retornará o quociente inteiro dessa divisão. Para sabermos o valor exato da divisão é necessário converter – *fazer o type casting* – transformando uma das variáveis para float ou double. Para isso use o comando w = (float)x / y ou w = (double)x / y. Lembre-se: apesar de no exemplo acima a conversão ter sido feita com a variávei x, você pode converter tanto o x quanto o y. Cole o seu programa no quadro abaixo.

Cole aqui o seu programa

Agora digite o programa abaixo e o execute.

import java.util.Scanner;

public class Prog02

{

public static void main(String[] args) {

Scanner ent = new Scanner(System.in);

System.out.print("Digite um numero: ");

int x = ent.nextInt();

System.out.print("Digite seu nome: ");

String y = ent.next();

System.out.print("Digite um numero real: ");

double z = ent.nextDouble();

System.out.print("x = "+x+"\ny = "+y+"\nz = "+z);

}

}

Durante a execução digite os valores 5, Maria e 3.14. Aconteceu algum erro?

Resposta:

Agora digite os valores 5, Maria e 3,14. Por que esse erro aconteceu?

Resposta:

Agora execute novamente o programa e digite os seguintes valores: 5, Maria Silva, 3,14. Aconteceu algum erro?

O método next() da classe Scanner só deve ser utilizado quando digitamos strings sem espaço. Quando digitamos strings com espaços, devemos utilizar o método nextLine(). Altere o programa acima para utilizar o nextLine() ao invés do next(). Execute novamente o programa e digite os seguintes valores: 5, Maria Silva, 3,14. Aconteceu algum erro?

Resposta:

Após trocarmos o next() pelo nextLine(), o problema é que não conseguimos mais digitar o nome. Quando temos um nextLine() logo após a leitura de uma variável numérica, é necessário *limpar o buffer*. Para isso, basta ter um nextLine() antes da leitura do String (esse *nextLine()* extra vai consumir caracteres que ficaram no buffer após a execução, por exemplo, de um *nextInt()*).

import java.util.Scanner;

public class Prog02

{

public static void main(String[] args) {

Scanner ent = new Scanner(System.in);

System.out.print("Digite um numero: ");

int x = ent.nextInt();

ent.nextLine(); *// limpando o buffer de entrada*

System.out.print("Digite seu nome: ");

String y = ent.nextLine();

System.out.print("Digite um numero real: ");

double z = ent.nextDouble();

System.out.print("x = "+x+"\ny = "+y+"\nz = "+z);

}

}

**Funções matemáticas no Java**

Java nos fornece diversas funções matemáticas com o uso da classe Math. Por exemplo as funções Math.abs() para cálculo do valor absoluto, Math.pow() para cálculo de uma potência, Math.sqrt() para o cálculo da raiz quadrada e a constante Math.PI.

Faça os programas abaixo utilizando essas funções.

1 - Faça um programa para calcular a hipotenusa de um triângulo retângulo.

Cole aqui o seu programa

2 – Faça um programa para calcular a distância euclidiana entre 2 pontos.

Cole aqui o seu programa

3 – Faça um programa para calcular a área de um setor circular.

Cole aqui o seu programa

**Compilando e executando um programa com classes definidas em diferentes arquivos**

Digite o código abaixo.

class CalculadoraSimples {

public int soma(int op1, int op2) {

return op1 + op2;

}

public int subtrai(int op1, int op2) {

return op1 - op2;

}

public int multiplica(int op1, int op2) {

return op1 \* op2;

}

public double divide(int op1, int op2) {

return op1 / op2;

}

}

Você deve salvar esse arquivo com o nome CalculadoraSimples.java e compilar para que o java gere o arquivo CalculadoraSimples.class. Note que essa classe não possui o método main() e, portanto, não pode ser executada (apenas compilada).

Agora digite o código a seguir e salve no mesmo diretório. Compile e execute seu programa.

import java.util.Scanner;

public class ExemploExcecao {

public static void main(String[] args) {

CalculadoraSimples calc = new CalculadoraSimples();

Scanner entrada = new Scanner(System.in);

double x = 9f/0;

String continua = "S";

while(continua.toUpperCase().charAt(0) == 'S'){

System.out.print("Digite o numerador: ");

int num1 = entrada.nextInt();

System.out.print("Digite o denominador: ");

int num2 = entrada.nextInt();

double resultado = calc.divide(num1, num2);

System.out.print(num1+" / "+num2+" = "+resultado);

entrada.nextLine(); // esvaziar o buffer

System.out.print("\nDeseja continuar (S|N) ? : ");

continua = entrada.nextLine();

}

}

}