



**INSTITUTO INFNET**

**LÍVIA FARIA BRAZ**

**TESTE DE PERFORMANCE 1**

**Disciplina Regular 1:** Fundamentos de  
Desenvolvimento com Java

**Professor:** Bernardo Petry Prates

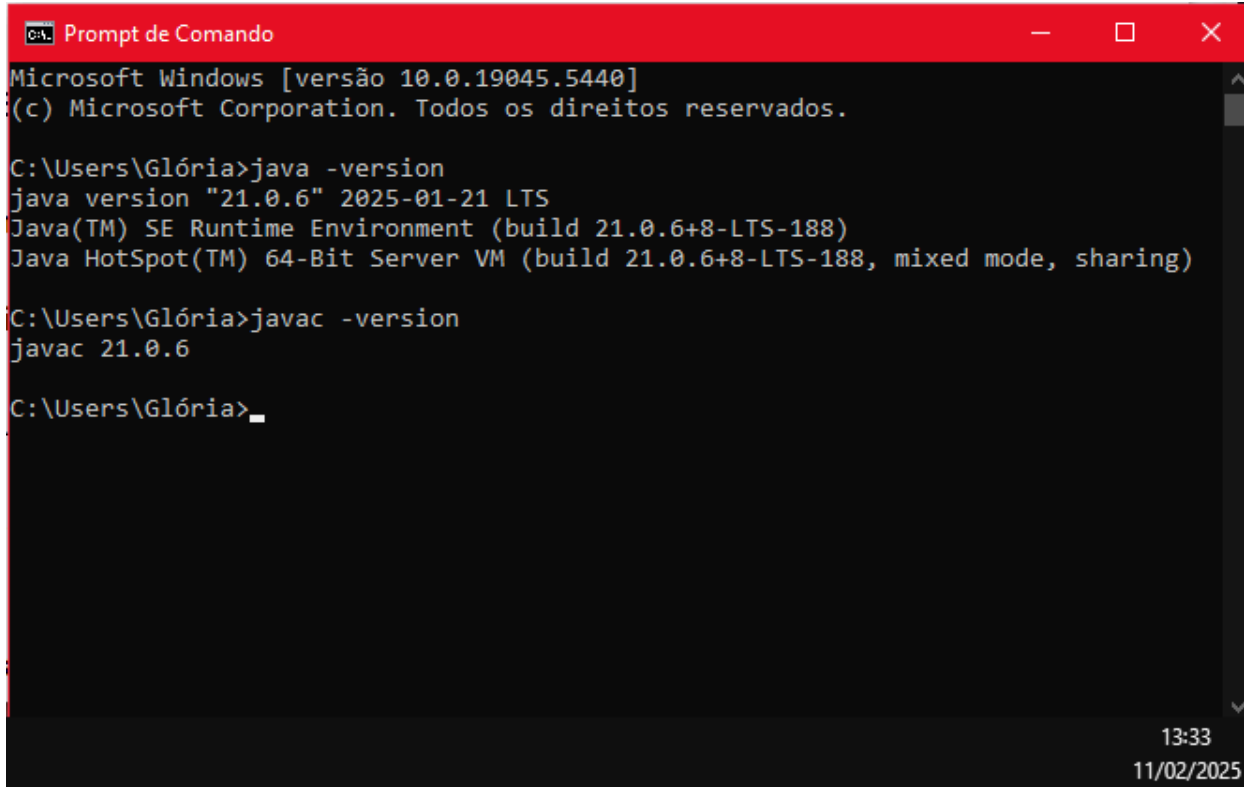
**ITAÚNA - MG**

**11 de fevereiro de 2025**



## Fundamentos de Desenvolvimento com Java - TP1

### 1) Instalar o JDK



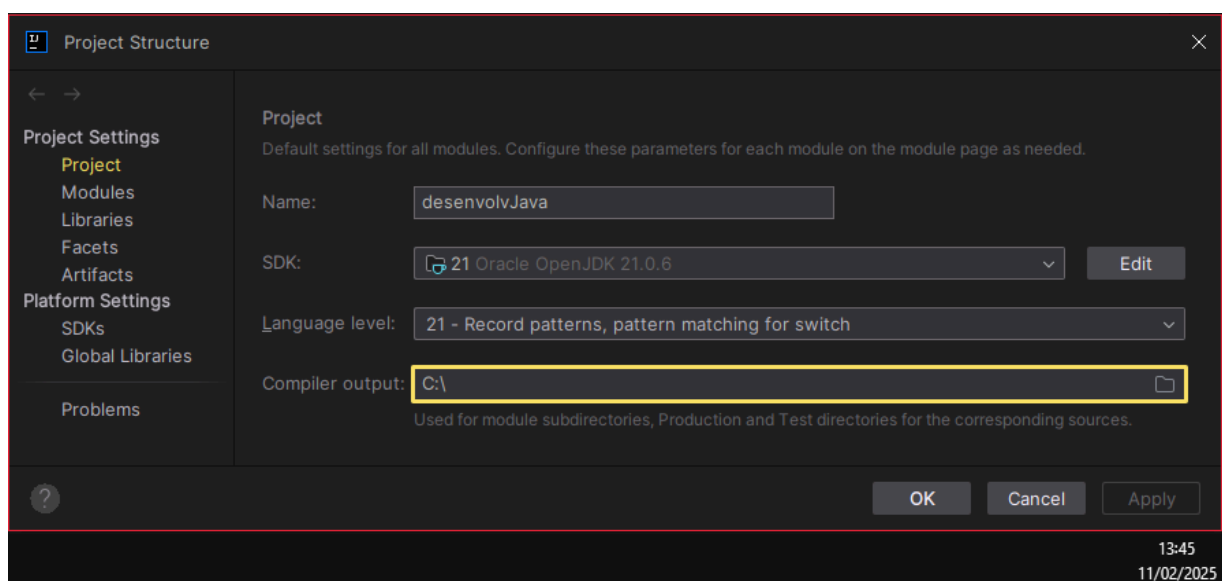
```
C:\Users\Glória>java -version
java version "21.0.6" 2025-01-21 LTS
Java(TM) SE Runtime Environment (build 21.0.6+8-LTS-188)
Java HotSpot(TM) 64-Bit Server VM (build 21.0.6+8-LTS-188, mixed mode, sharing)

C:\Users\Glória>javac -version
javac 21.0.6

C:\Users\Glória>
```

13:33  
11/02/2025

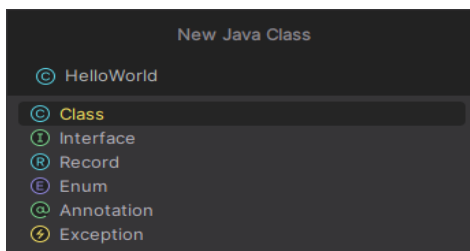
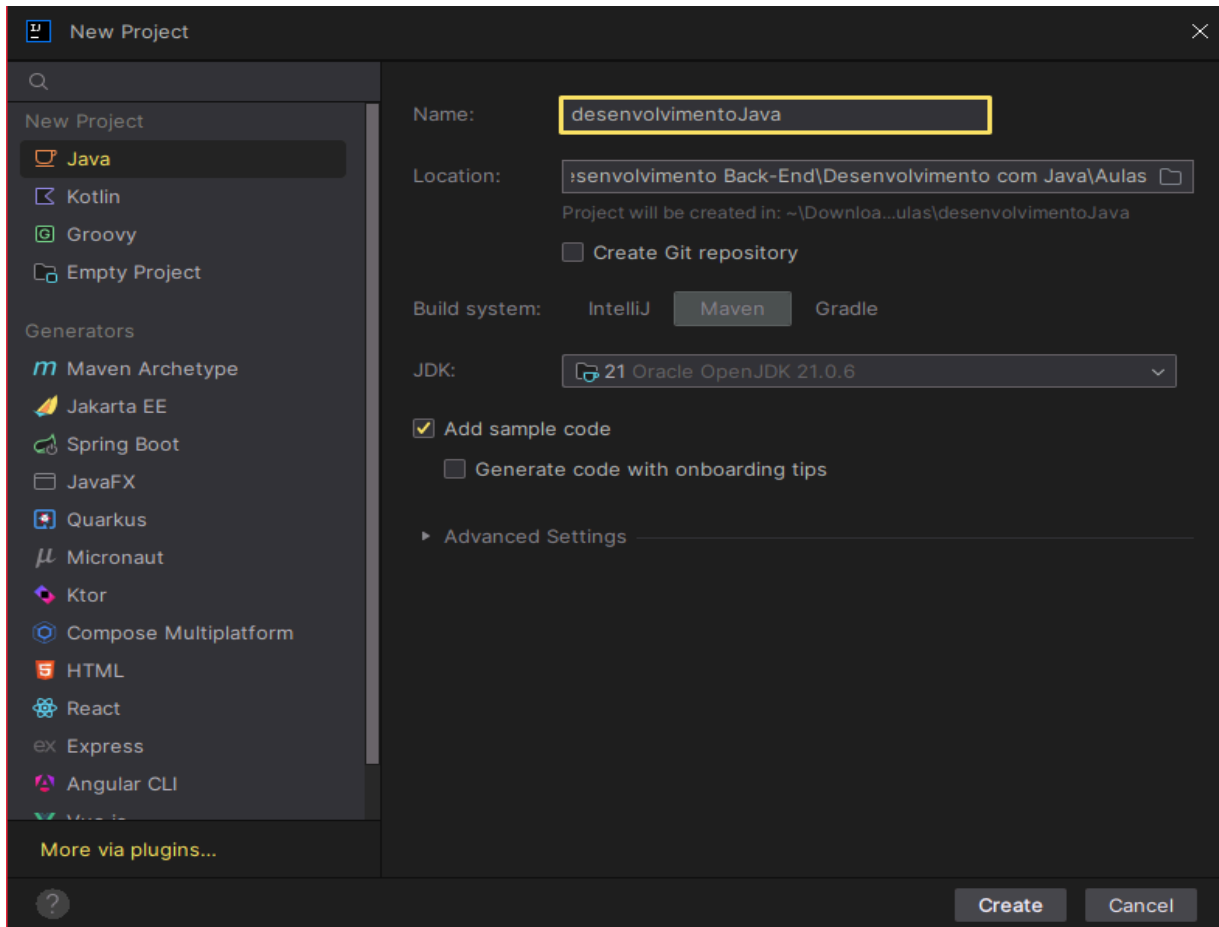
### 2) Instalar e Configurar a IDE IntelliJ IDEA





## Fundamentos de Desenvolvimento com Java - TP1

### 3) Criar um Projeto Básico de Java



```
package com.codingloria.TP1;
public class HelloWorld {
    public static void main(String[] args) {
    }
}
```

### 4) Escrever um Programa Simples em Java

```
public class HelloWorld {
    public static void main(String[] args) {
        System.out.println("Hello, World!");
    }
}
```



```
Build    Build Output  x
✓ desenvolvJava: t5 sec, 353 ms  Updating dependency information... [desenvolvimentoJava]
                                   Parsing java... [desenvolvimentoJava]
                                   Writing classes... [desenvolvimentoJava]
                                   Adding nullability assertions... [desenvolvimentoJava]
                                   Adding pattern assertions... [desenvolvimentoJava]
                                   Dependency analysis found 0 affected files
                                   Running 'after' tasks
                                   javac 21.0.6 was used to compile java sources
                                   Finished, saving caches...
                                   Module 'desenvolvimentoJava' was fully rebuilt due to project config
                                   Executing post-compile tasks...
                                   Synchronizing output directories...
                                   11/02/2025 16:10 - Build completed successfully in 5 sec, 353 ms
```

The screenshot shows an IDE with the following elements:

- Project Structure (Left):**
  - desenvolvimentoJava
    - .idea
    - src
      - main
        - java
          - com.codingloria
            - aula02
            - aula03
            - aula04
            - exercise02
            - exercise03
            - TP1

- Code Editor (Center):**

```

1 package com.codingloria.TP1;
2
3 public class HelloWorld {
4     public static void main(String[] args) {
5         System.out.println("Hello, World!");
6     }
7 }
8

```
- Context Menu (Right):**
- New
- Cut (Ctrl+X)
- Copy (Ctrl+C)
- Copy Path/Reference...
- Paste (Ctrl+V)
- Find Usages (Alt+F7)
- Analyze
- Rename... (Shift+F6)
- Refactor
- Bookmarks
- Reformat Code (Ctrl+Alt+L)
- Optimize Imports (Ctrl+Alt+O)
- Delete... (Excluir)
- Override File Type
- Run 'HelloWorld.main()' (Ctrl+Shift+F10)** (highlighted with a red box)
- Debug 'HelloWorld.main()'
- More Run/Debug
- Open in Right Split (Shift+Enter)
- Open In
- Local History
- Repair IDE on File
- Reload from Disk
- Compare With... (Ctrl+D)
- Diagrams
- Convert Java File to Kotlin File (Ctrl+Alt+Shift+K)
- SonarQube for IDE
- GitHub Copilot
- Bottom Bar:**
- desenvolvimentoJava > src > main
- 8:1 CRLF UTF-8 4 spaces
- 16:19 11/02/2025



## Fundamentos de Desenvolvimento com Java - TP1

---

### 7) Modificar e Re-executar o Programa

```
package com.codingloria.TP1;

public class HelloWorld {
    public static void main(String[] args) {
        System.out.println("Meu nome é Livia, mas todos me chamam de Glória :)");
        System.out.println("Tenho 22 anos. \nEu sou estudante de Engenharia de
Software na INFNET");
    }
}
```

```
Run | HelloWorld x
C:\Program Files\Java\jdk-21\bin\java.exe" "-javaagent:C:\Users\Glória\AppData\Local\Progra
Meu nome é Livia, mas todos me chamam de Glória :)
Tenho 22 anos.
Eu sou estudante de Engenharia de Software na INFNET
Process finished with exit code 0
Performance
```

### 8) Usar Variáveis e Tipos de Dados

```
package com.codingloria.TP1;

public class Exercicio08 {
    public static void main(String[] args) {
        int idade = 22;
        double altura = 1.68;
        String nome = "Gloria";
        char sexo = 'F';
    }
}
```



## Fundamentos de Desenvolvimento com Java - TP1

```
System.out.println("Nome: " + nome);  
System.out.println("Idade: " + idade);  
System.out.println("Altura: " + altura);  
System.out.println("Sexo: " + sexo);  
}  
}
```

```
Run Exercício08 x  
"C:\Program Files\Java\jdk-21\bin\java.exe" "-javaagent:C:\Users\Glória\AppData\Local\Progra  
Nome: Glória  
Idade: 22  
Altura: 1.68  
Sexo: F  
Process finished with exit code 0  
Performance  
FundamentosJava > src > main > java > com > codingloria > TP1 > Exercício08 17:1 CRLF UTF-8 4 spaces  
16:38  
11/02/2025
```

### 9) Ler Entrada do Usuário

```
package com.codingloria.TP1;  
import java.util.Scanner;  
  
public class Exercício09 {  
    public static void main(String[] args) {  
        Scanner sc = new Scanner(System.in);  
  
        System.out.print("Digite seu nome: ");  
        String nome = sc.nextLine();  
    }  
}
```



## Fundamentos de Desenvolvimento com Java - TP1

```
System.out.print("Digite sua idade: ");  
  
int idade = sc.nextInt();  
  
System.out.println("Nome: " + nome + ", Idade: " + idade);  
  
}  
}
```

Run Exercício09 x

"C:\Program Files\Java\jdk-21\bin\java.exe" "-javaagent:C:\Users\Glória\AppData\Local\Progra  
Digite seu nome: Gloria  
Digite sua idade:22  
Nome: Gloria, Idade: 22  
  
Process finished with exit code 0

Performance

main > java > com > codingloria > TP1 > Exercício09 1:1 (418 chars, 17 line breaks) CRLF UTF-8 4 spaces

16:43  
11/02/2025

### 10) Depurar Código

```
package com.codingloria.TP1;  
  
import java.util.Scanner;  
  
public class Exercício09 {  
  
    public static void main(String[] args) {  
  
        Scanner sc = new Scanner(System.in);  
  
        System.out.print("Digite seu nome: ");  
  
        String nome = sc.nextLine();  
  
        System.out.print("Digite sua idade: ");  
  
        sc.close(); // ERRO 1  
  
        int idade = sc.nextInt(); // ERRO 2
```



## Fundamentos de Desenvolvimento com Java - TP1

```
System.out.println( "Digite sua altura: ");

double altura = sc.nextDouble();

System.out.println("Nome: " + nome + ", Idade: " + idade + ", Altura: " +
altura); // ERRO 3

sc.close();

}
```

Após a execução:

```
Debug Exercício09 x
C:\Program Files\Java\jdk-21\bin\java.exe" ...
Connected to the target VM, address: '127.0.0.1:51154', transport: 'socket'
Digite seu nome: Gloria
Digite sua idade: Exception in thread "main" Disconnected from the target VM, address: '127.0.0.1:51154'
java.lang.IllegalStateException: Scanner closed
    at java.base/java.util.Scanner.ensureOpen(Scanner.java:1158)
    at java.base/java.util.Scanner.findWithinHorizon(Scanner.java:1790)
    at java.base/java.util.Scanner.nextLine(Scanner.java:1658)
    at com.codingloria.TP1.Exercicio09.main(Exercicio09.java:14)
Process finished with exit code 1
```

1. Coloquei os seguintes break points, e iniciei o debug

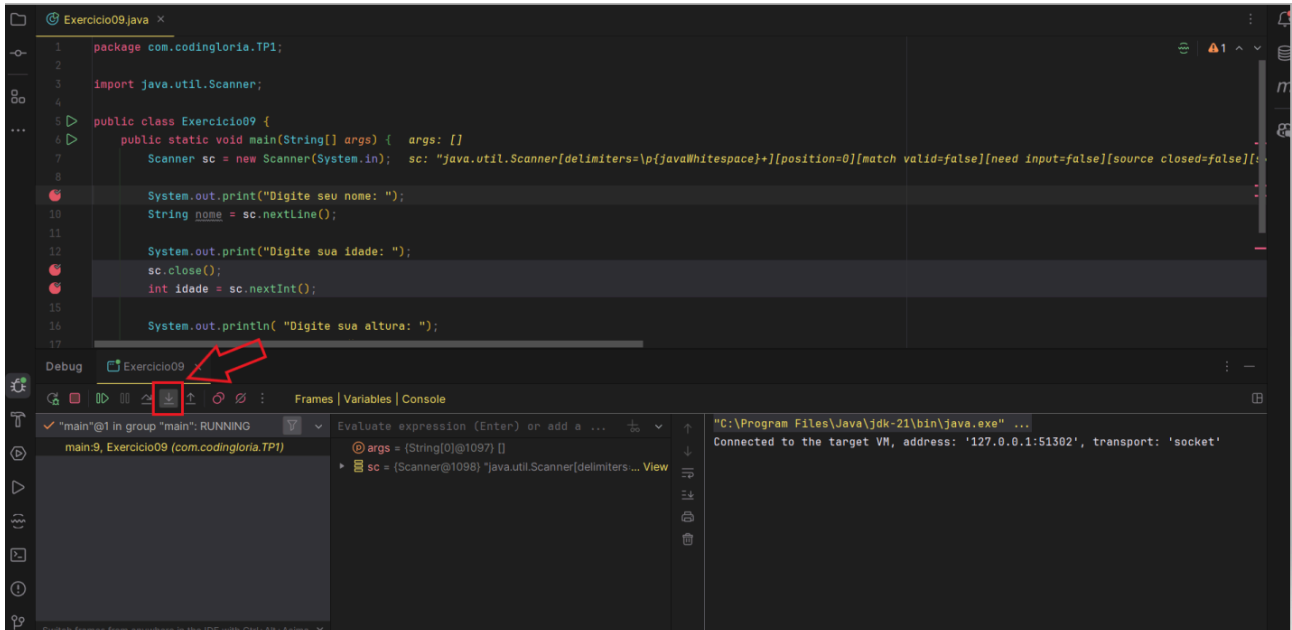
```
1 package com.codingloria.TP1;
2
3 import java.util.Scanner;
4
5 public class Exercicio09 {
6     public static void main(String[] args) {
7         Scanner sc = new Scanner(System.in);
8
9         System.out.print("Digite seu nome: ");
10        String nome = sc.nextLine();
11
12        System.out.print("Digite sua idade: ");
13        sc.close();
14        int idade = sc.nextInt();
15
16        System.out.println( "Digite sua altura: ");
17        double altura = sc.nextDouble();
18
19        System.out.println("Nome: " + nome + ", Idade: " + idade + ", Altura: " + altura);
20    }
21 }
22
```



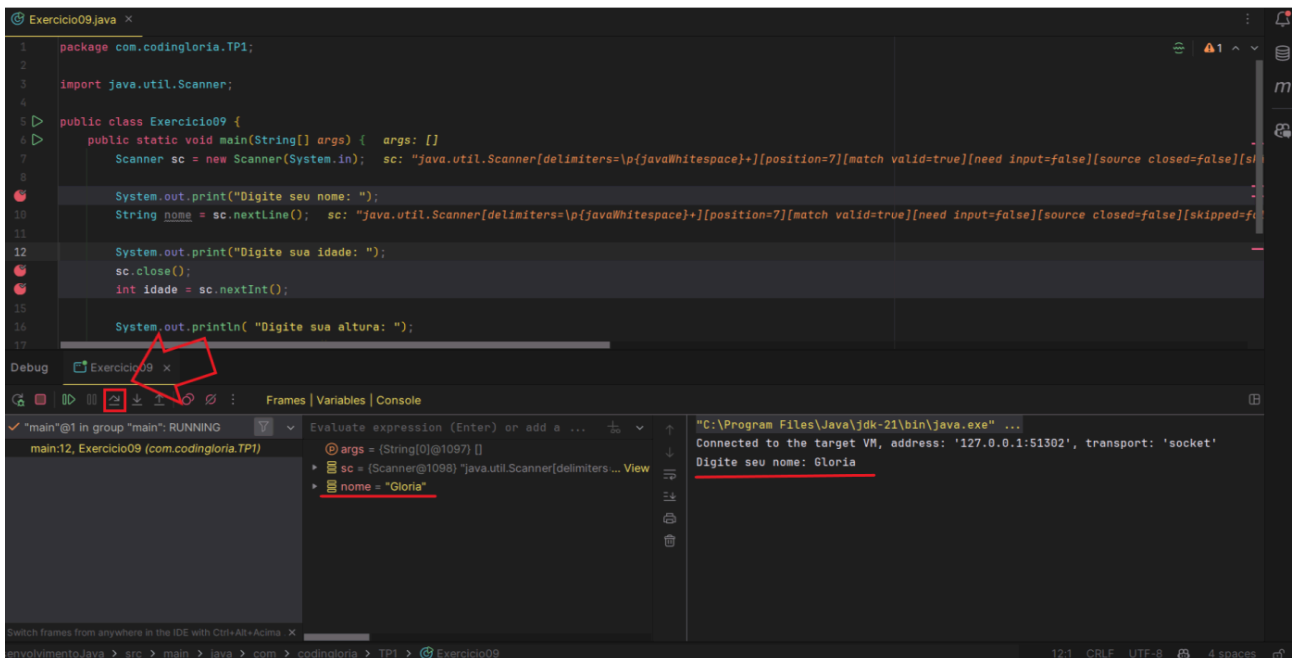


## Fundamentos de Desenvolvimento com Java - TP1

1. Clicamos em Step-Into para verificar se executa corretamente o primeiro print



2. A variável “nome” é armazenada corretamente, então seguimos com o Step-Over





## Fundamentos de Desenvolvimento com Java - TP1

- Após continuarmos as verificações vimos que o scanner foi fechado ANTES de pegar as próximas informações, retornando assim uma exceção:

```
public class Exercicio09 {  
    public static void main(String[] args) {  
        Scanner sc = new Scanner(System.in);  
  
        System.out.print("Digite seu nome: ");  
        String nome = sc.nextLine();  
  
        System.out.print("Digite sua idade: ");  
        sc.close();  
        int idade = sc.nextInt();  
  
        System.out.println("Digite sua altura: ");  
    }  
}
```

Debug Console Output:

```
"C:\Program Files\Java\jdk-21\bin\java.exe" ...  
Connected to the target VM, address: '127.0.0.1:51497', transport: 'socket'  
Digite seu nome: Gloria  
Digite sua idade: Exception in thread "main" java.lang.IllegalStateException: Scanner closed  
    at java.base/java.util.Scanner.ensureOpen(Scanner.java:1158)  
    at java.base/java.util.Scanner.next(Scanner.java:1581)  
    at java.base/java.util.Scanner.nextInt(Scanner.java:2267)  
    at java.base/java.util.Scanner.nextInt(Scanner.java:2221)  
    at com.codingloria.TP1.Exercicio09.main(Exercicio09.java:14)  
Disconnected from the target VM, address: '127.0.0.1:51497', transport: 'socket'  
  
Process finished with exit code 1
```

- Fazemos o ajuste, e reiniciamos o processo de debug. Agora, todas as variáveis são armazenadas corretamente e o código vai fluindo.

```
public class Exercicio09 {  
    public static void main(String[] args) {  
        args: []  
  
        System.out.print("Digite sua idade: ");  
        int idade = sc.nextInt(); idade: 22  
  
        System.out.print("Digite sua altura: ");  
        double altura = sc.nextDouble(); altura: 1.68  
  
        System.out.println("Nome: + nome + , Idade: " + idade + ", Altura: " + altura); idade: 22 altura: 1.68  
  
        sc.close(); sc: "java.util.Scanner[delimiters=\p(javaWhitespace)+][position=14][match valid=true][need input=false][source closed=false][skipped=false][group separator=\\n]"  
    }  
}
```

Debug Console Output:

```
"C:\Program Files\Java\jdk-21\bin\java.exe" ...  
Connected to the target VM, address: '127.0.0.1:51543', transport: 'socket'  
Digite seu nome: Gloria  
Digite sua idade: 22  
Digite sua altura: 1,68  
Nome: + nome + , Idade: 22, Altura: 1.68
```

Variables Window:

```
@ args = (String[0]@1097) []  
sc = {Scanner@1098} "java.util.Scanner[delimiters=\p(javaWhite... View  
nome = "Gloria"  
idade = 22  
altura = 1.68
```



## Fundamentos de Desenvolvimento com Java - TP1

5. No entanto, como podemos ver no print anterior, o resultado esperado não é retornado no console. O output aparece errado.

The screenshot shows an IDE with a Java file named `Exercicio09.java`. The code is as follows:

```
public class Exercicio09 {  
    public static void main(String[] args) {  
        System.out.print("Digite sua idade: ");  
        int idade = sc.nextInt(); idade: 22  
  
        System.out.print("Digite sua altura: ");  
        double altura = sc.nextDouble(); altura: 1.68  
  
        System.out.println("Nome: + nome + , Idade: " + idade + ", Altura: " + altura); idade: 22 altura: 1.68  
  
        sc.close();  
    }  
}
```

The IDE is in a debug state. The console output shows the following sequence of prompts and user input:

```
Digite seu nome: Gloria  
Digite sua idade: 22  
Digite sua altura: 1.68  
Nome: + nome + , Idade: 22, Altura: 1.68
```

A red arrow points to the line `System.out.println("Nome: + nome + , Idade: " + idade + ", Altura: " + altura);` in the code, indicating the source of the formatting error in the output.

6. Analisando a linha 18, notamos um erro de formatação do que deveria ser impresso. Após corrigir, o retorno vem corretamente, como esperado. :]

The screenshot shows the same IDE with the corrected Java code:

```
public class Exercicio09 {  
    public static void main(String[] args) {  
        System.out.print("Digite seu nome: ");  
        String nome = sc.nextLine();  
  
        System.out.print("Digite sua idade: ");  
        int idade = sc.nextInt();  
  
        System.out.print("Digite sua altura: ");  
        double altura = sc.nextDouble();  
  
        System.out.println("Nome: " + nome + ", Idade: " + idade + ", Altura: " + altura);  
  
        sc.close();  
    }  
}
```

The console output now shows the correct formatted output:

```
Digite seu nome: Gloria  
Digite sua idade: 22  
Digite sua altura: 1.68  
Nome: Gloria, Idade: 22, Altura: 1.68  
Disconnected from the target VM, address: '127.0.0.1:51984', transport: 'socket'  
Process finished with exit code 0
```



### ➤ Detalhes sobre os erros que inserimos e como corrigimos

#### Erro 1: Fechar o Scanner antes de usá-lo

- O primeiro erro acontece quando chamamos `sc.close()`; logo após ler o nome. Isso fecha o `Scanner`, e logo depois, tentamos usá-lo para ler a idade com `sc.nextInt()`. Quando o `Scanner` é fechado, não podemos mais usá-lo, e isso resulta em uma exceção.
- **Como depurei:**
  - Coloquei um breakpoint na linha `sc.close()`; logo após ler o nome.
  - Executei o código no modo Debug.
  - Vemos que, ao chegar nessa linha, o `Scanner` foi fechado e o programa não pode mais funcionar corretamente após essa linha.
- **Solução:**  
Remover o `sc.close()`; dessa linha e passo para o final, após a última operação de leitura.

---

#### Erro 2: Tentando ler após o Scanner ser fechado

- Quando o `Scanner` foi fechado antes da leitura da idade, ao tentar ler `int idade = sc.nextInt()`; o programa não consegue acessar o fluxo de entrada, pois o `Scanner` já foi fechado, o que gera uma exceção (`IllegalStateException`).
- **Como depurei:**
  - Coloquei um breakpoint na linha `int idade = sc.nextInt()`; e executei o código.
  - Ao usar **Step Into** (F7), vemos que o código trava ao tentar acessar o `Scanner` já fechado.
  - Na janela do Debug, vimos que o `Scanner` já está fechado e, portanto, não podia ler mais nada.
- **Solução:**  
Após remover o primeiro `sc.close()`, essa linha volta a funcionar corretamente.



## Fundamentos de Desenvolvimento com Java - TP1

---

### Erro 3: Erro de concatenação no `System.out.println`

- Na linha:

```
System.out.println("Nome: + nome + , Idade: " + idade + ",  
Altura: " + altura);
```

Temos um erro de sintaxe: as aspas duplas ao redor do `+ nome +` fazem com que o texto literal seja impresso junto com a variável `nome`. O esperado seria concatenar a string corretamente, mas a forma que foi escrita não vai funcionar como esperado.

- **Como depurei:**

- Coloquei um breakpoint nessa linha e executei o código.
- Com o breakpoint, vimos que o texto não era exibido corretamente no console devido à concatenação incorreta.

- **Solução:**

Corrijo a concatenação removendo as aspas ao redor de `+ nome +`:

```
System.out.println("Nome: " + nome + ", Idade: " + idade + ",  
Altura: " + altura);
```

---

### Resultado final:

Após corrigir todos os erros, o código foi executado corretamente e retornou o que era esperado! :]