

Resumo Quinta-feira Java

Abordamos sobre variáveis, como declarar seguindo as regras proposta pela a Sun.

Variáveis difinição:

Uma variável é um objeto capaz de reter e representar um valor ou expressão. Assim ela pode conter uma informação ou um dado.

Dado pode ou não sofrer alteração ao decorrer do programa.

Regras da Sun sobre variáveis!

Regra	Descrição
Início	Os nomes das variáveis devem começar com uma letra, \$ ou _ . Não podem começar com um dígito.
Case Sensitivity	Java é sensível a maiúsculas e minúsculas, o que significa que nome e Nome seriam diferentes variáveis.
Nomes Reservados	Palavras-chave como int , for , class , etc., não podem ser usadas como nomes de variáveis.
Sem Espaços	Os nomes das variáveis não podem conter espaços.
Caracteres Especiais	Caracteres especiais como ! , @ , # , % , etc., não são permitidos, exceto \$ e _ .
Nomes Significativos	Embora não seja uma regra sintática, é uma boa prática de programação usar nomes de variáveis que representem seu propósito no código.

Caso a variável tenha nome composto, devemos declarar da seguinte maneira, digamos que queremos armazenar nome de um usuário.

```

Tipo      |      nome      | atribuição

String nomeUsuario = "José";
```

Sobre os tipos das variaveis

O tamanho, será o espaço que será alocado na memória, o tamanho é a capacidade de dados que

podem ser atribuídos.

Categoria	Tipo	Tamanho	Intervalo de Valores
Número Inteiro	byte	1 Byte	-128 a 127
Número Inteiro	short	2 Bytes	-32,768 a 32,767
Número Inteiro	int	4 Bytes	-2,147,483,648 a 2,147,483,647
Número Inteiro	long	8 Bytes	-9,223,372,036,854,775,808 a 9,223,372,036,854,775,807
Número Decimal	float	4 Bytes	Aproximadamente $\pm 3.40282347E+38F$ (6-7 casas decimais significativas)
Número Decimal	double	8 Bytes	Aproximadamente $\pm 1.79769313486231570E+308$ (15 casas decimais significativas)
Lógico	boolean	N/A	true ou false

Sobre o bloco de notas

Antes vamos retornar a execução do java com o bloco de notas! Com o JDK instalado e as variáveis de ambiente configuradas, pode criar uma pasta em qualquer lugar do windows, que iremos conseguir executar o nosso arquivo em java, a final, configuramos as variáveis de ambiente.

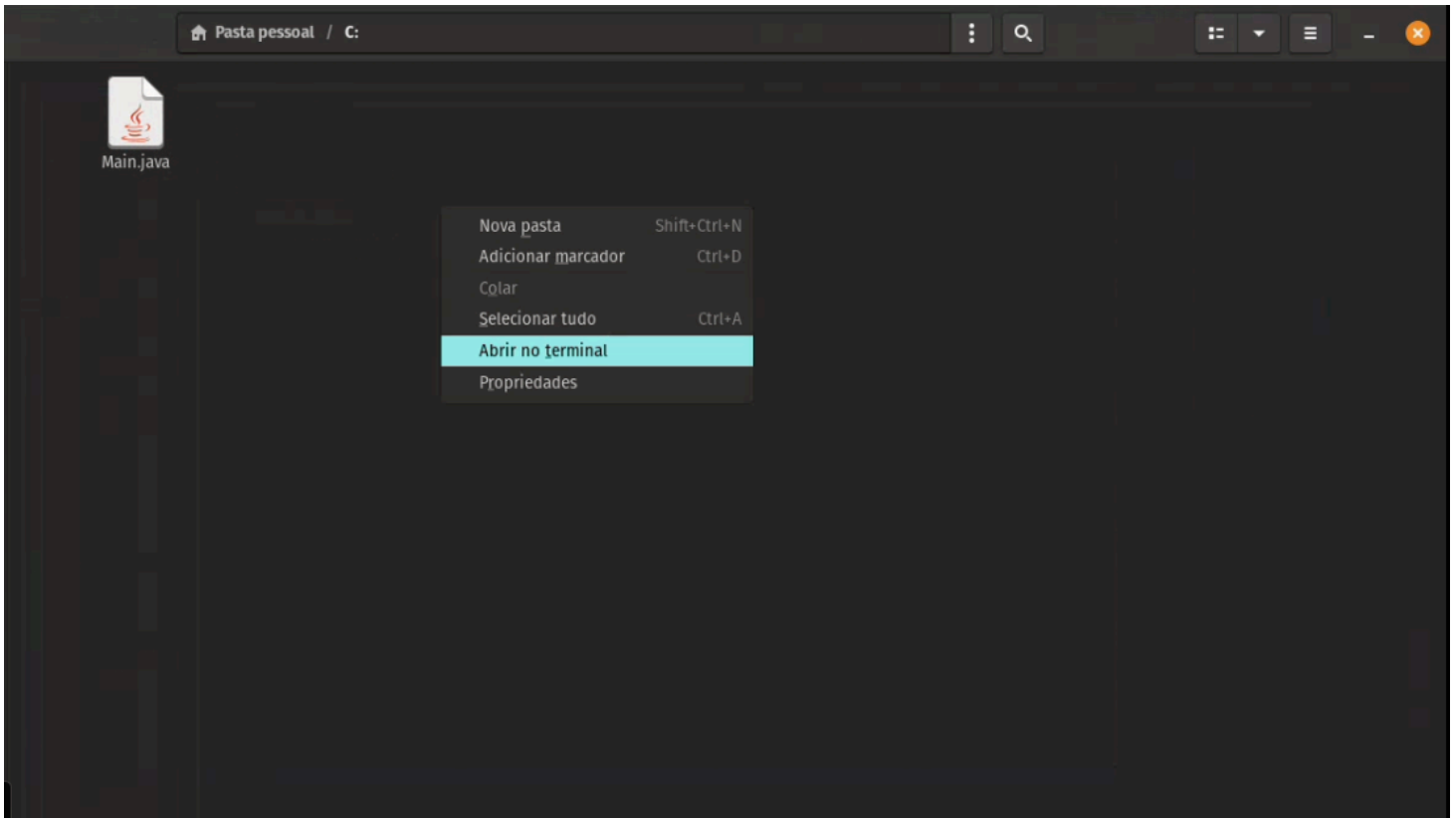
- Regras para o arquivo!
O arquivo deve conter o **nome** que foi atribuído a class quando iniciamos o programa, nessa caso, declarei que o nome da minha classe que vai conter o método principal vai a **Main**, isso vai dizer o que? Que meu arquivo tem que ser chamar **Main.java**.
- O nome pode ser qualquer um, nesse exemplo usei *Main*, mas poderia ser, ExitDado, seu nome, bola, quadrado etc, mas o nome do arquivo terá que conter o mesmo nome, tudo bem?

exemplo.

```
public class Main{  
    public static void main(String[] args){  
        System.out.println("Olá, mundo!!");  
    }  
}
```

Vamos criar o arquivo e fazer a execução dele através do CMD.

Podemos abrir o CMD direto na pasta, basta acessar a pasta que criamos o arquivo e clicar com o botão direito do mouse e ir até **abrir terminal**, veja:



Podemos abrir o terminal assim ou navegar até a pasta que criamos, utilizando o comando **cd**.

```
// Volta uma pasta a cima da atual.  
cd ..
```

```
// Volta duas pasta a cima da atual  
cd ../..
```

```
// O comando dir lista o conteúdo da pasta  
dir
```

Com o terminal acessado na pasta, devemos rodar os seguintes comandos

```
//Javac é responsável por compilar
//a classe, por isso o arquivo deve conter o nome da classe
```

```
//Vai gerar um arquivo chamado
//Main.class, caso não tenha erro no código
//irá acontecer tudo corretamente.
```

```
javac Main.java
```

```
// Uma vez compilado basta rodar
```

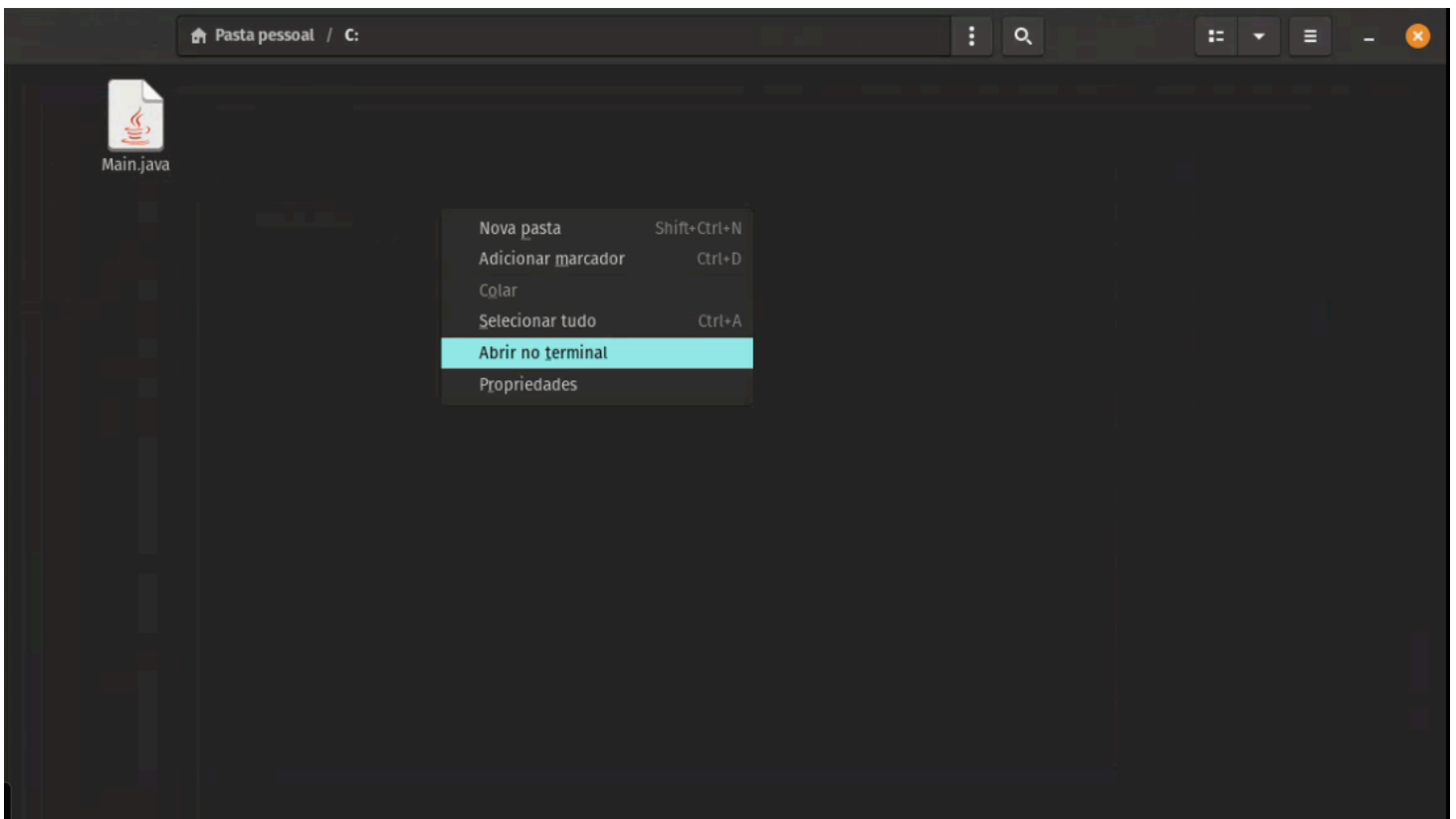
```
//Para rodar o arquivo, usamos
```

```
java Main
```

```
ou
```

```
java Main.java
```

Lembra que no início tinha apenas o arquivo **Main.java**, veja



Com a execução a pasta **C:**, recebe o arquivo **Java.class** como explicado a cima, olha o passo na imagem



Main.class



Main.java



shaka@pop-os: ~/C:



```
shaka@pop-os:~/C:$  
shaka@pop-os:~/C:$  
shaka@pop-os:~/C:$  
shaka@pop-os:~/C:$ javac Main.java  
shaka@pop-os:~/C:$  
shaka@pop-os:~/C:$  
shaka@pop-os:~/C:$ java Main  
Olá, mundo!!  
shaka@pop-os:~/C:$  
shaka@pop-os:~/C:$ # OU  
shaka@pop-os:~/C:$  
shaka@pop-os:~/C:$ java Main.java  
Olá, mundo!!  
shaka@pop-os:~/C:$
```