




Aula 2

- Passo 1: Criar o repositório no github



Required fields are marked with an asterisk (*).


Owner *  romulodf-cesar / Repository name * 

 jkalangosenai is available.

Great repository names are short and memorable. Need inspiration? How about [literate-dollop](#) ?

Description (optional)

 ☒  Public
Anyone on the internet can see this repository. You choose who can commit.

☐  Private
You choose who can see and commit to this repository.

Initialize this repository with:


☐ Add a README file
This is where you can write a long description for your project. [Learn more about READMEs.](#)


Add .gitignore

Choose which files not to track from a list of templates. [Learn more about ignoring files.](#)

Choose a license

A license tells others what they can and can't do with your code. [Learn more about licenses.](#)

 You are creating a public repository in your personal account.




```
git remote add origin https://github.com/romulodf-cesar/jkalangosenai.git
```

1 **2** **3**

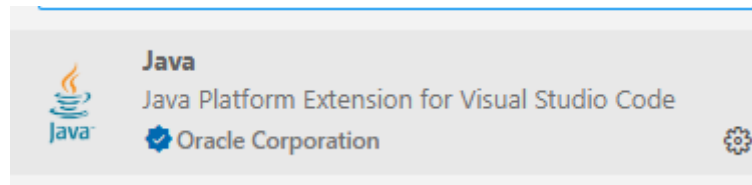
```
git remote add origin https://github.com/romulodf-cesar/jkalangosenai.git
```

Atenção! Cuidado!

 ☒ HTTPS ☐ SSH

Passo 2 – Instalar o Visual Studio Code

- Não é para instalar no sua casa agora, o Visual Studio Community
- A instalação é do Visual Studio Code
 - <https://code.visualstudio.com/>



○



Extension Pack for Java


Microsoft  microsoft.com |  37,148,236 | ★★★★★☆ (82)

Popular extensions for Java development that provides Java IntelliSense, debugging, t...

Disable

Uninstall ▼

Switch to Pre-Release Version

☒ Auto Update 

○



Test Runner for Java

Run and debug JUnit or TestNG test cases.

 Microsoft



○

○ Alô Mundo (Hello World)

- **Java I**

- Revisão

- Valores e Variáveis
 - Conversão de Tipos
 - Primitivos
 - Referência
 - **Cadastro de GDD** (documento principal de um Game Design)
 - **Objeto**
 - Classe
 - Construtor
 - Método
 - Atributo
 - Static - Singleton
 - Sobrecarga (overload)
 - Sobrescrita (override)
 - Herança
 - Polimorfismo
 - Encapsulamento
 - Interfaces
 - Implementação múltipla de interfaces
 - **Associação**
 - **Agregação**
 - **Composição**
 - Modelo Conceitual
 - **Entidade**
 - **Atributo**
 - **Relacionamento**

- **Java II**
 - **Conceitos da POO**
 - Lombok
 - Injeção de Dependências
 - **Conceitos de Web**
 - Client/Servidor
 - HTTP
 - Protocolo
 - Tipos de Protocolo
 - **Conceitos de API Web**
 - Aplicação: IFood, PagSeguro, Mercado Livre, Maquininha, Uber
 - **Conceitos Métodos HTTP**
 - GET
 - POST
 - PUT
 - DELETE
 - OUTROS
 - REST
 - RESTFull
 - Web API Eventos: JKalango
 - Jogos
 - IA
 - Design
 - Outros
- **Java Frameworks Web**
 - Spring
- **Java Frameworks ORM**
 - **JPA (especificação Java – padrão)**
 - Hibernate
- **Java Testes**
 - JUnit
 - Selenium

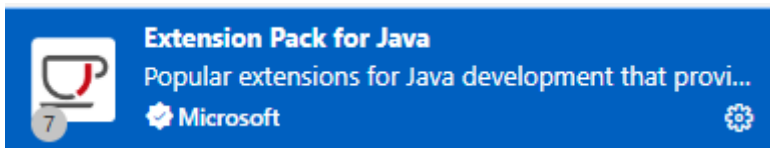
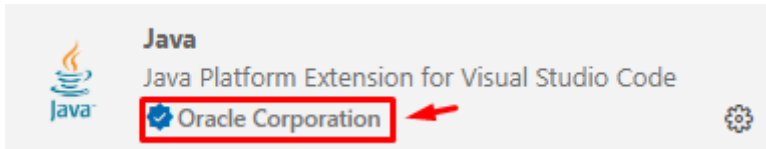
Java I

- **Revisão**
 - **Valores e Variáveis**
 - Conversão de Tipos
 - Primitivos
 - Referência

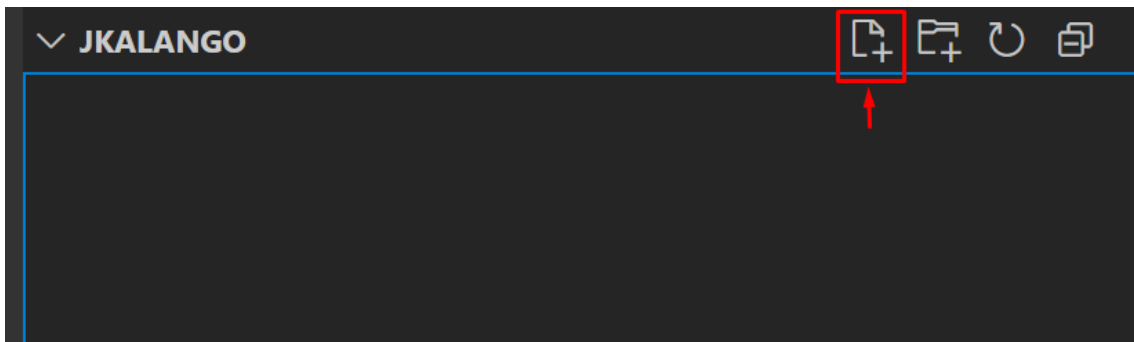
Crie uma pasta na área de trabalho



Abrir a pasta no VS Code e instalar as seguintes extensões:

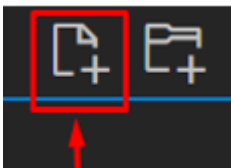


Verifique se abriu a pasta

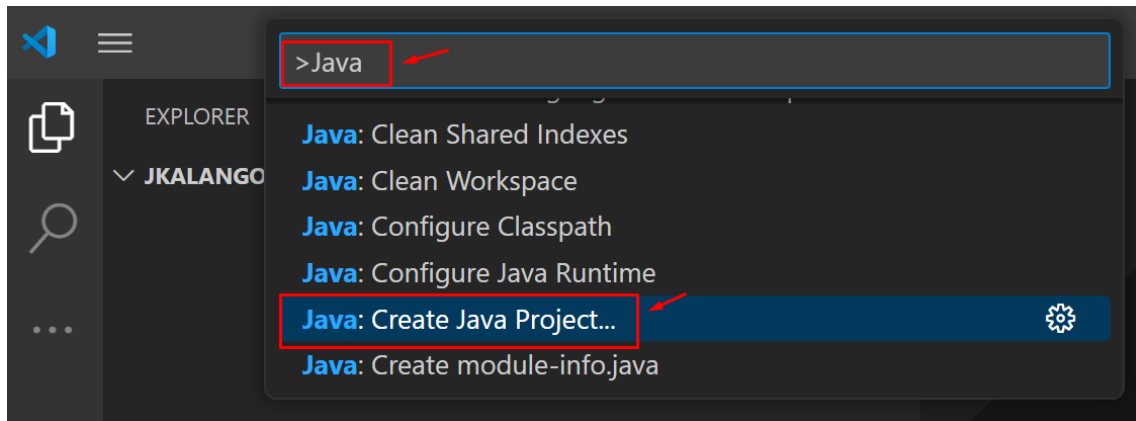


Atenção!

O primeiro é para arquivo e o segundo é para pasta.

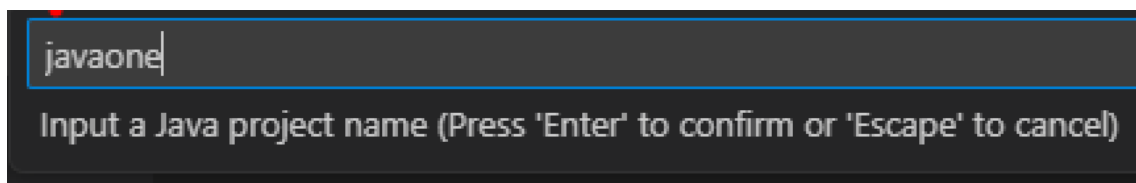
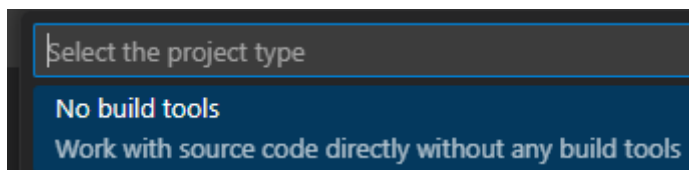


Criando um projeto java

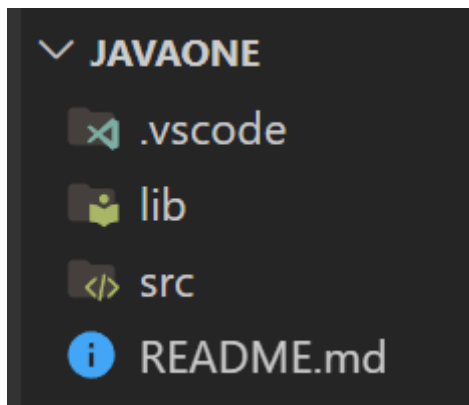


CTRL+SHIFT+P – tecla de atalho para criar o projeto Java.

Marcando a Opção



Módulo I – javaone

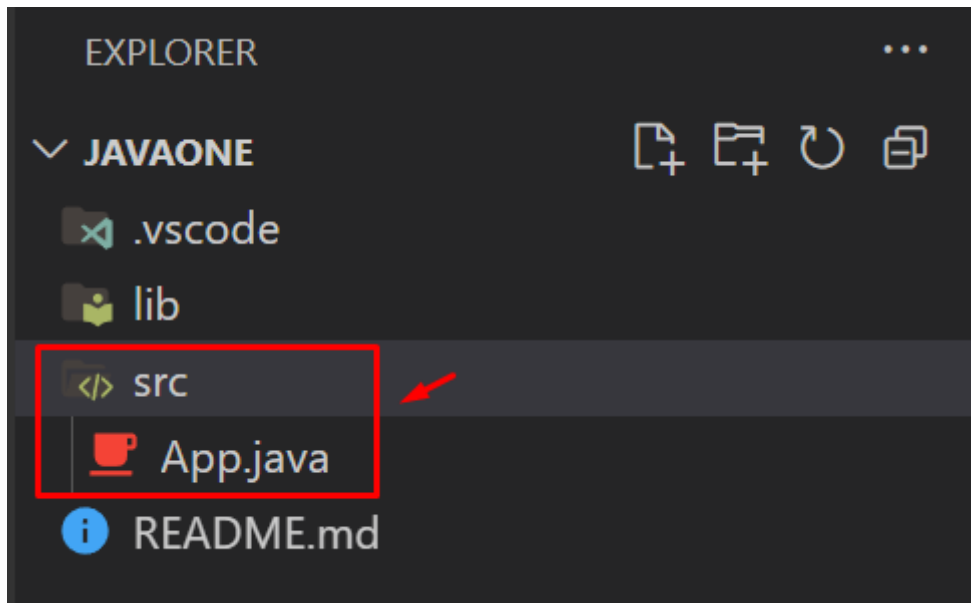


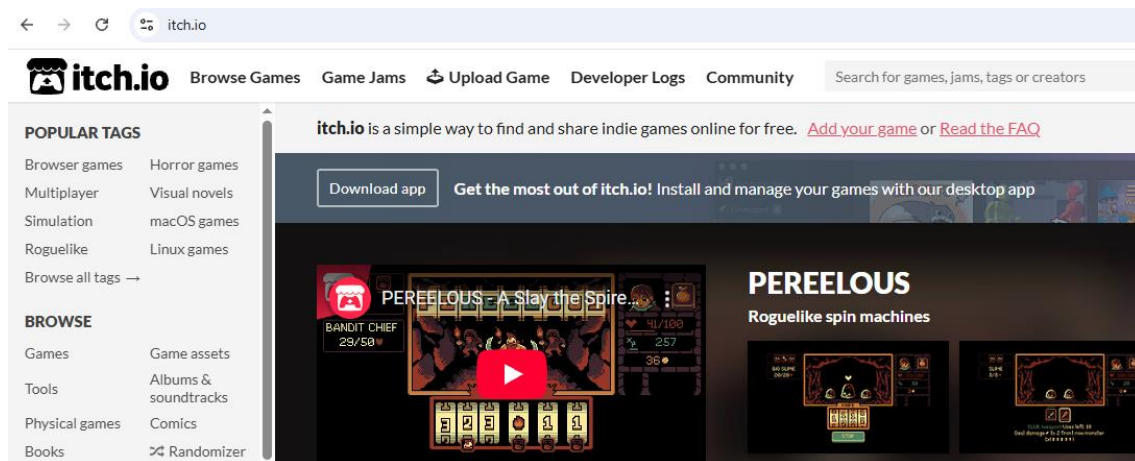
Meta de Entrega

- Valores e Variáveis
- Conversão de Tipos
- Primitivos
- Referência
- **Cadastro de GDD** (documento principal de um Game Design)
 - **Objeto**

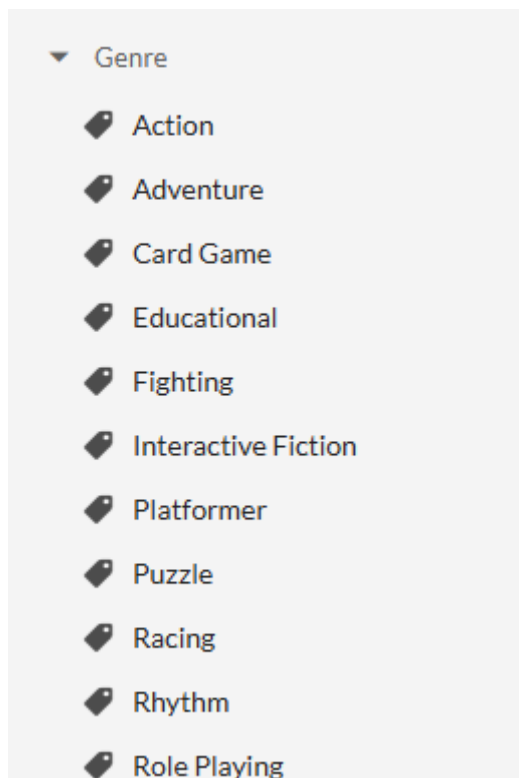
- Classe
- Construtor
- Método
- Atributo
- Static - Singleton
- Sobrecarga (overload)
- Sobrescrita (override)
- Herança
- Polimorfismo
- Encapsulamento
- Interfaces
- Implementação múltipla de interfaces
- **Associação**
- **Agregação**
- **Composição**
- Modelo Conceitual
 - **Entidade**
 - **Atributo**
 - **Relacionamento**

Aplicativo Hello World





Abstração



Classe: Genero

Atributo: nome

UML – Linguagem de Modelagem Unificada

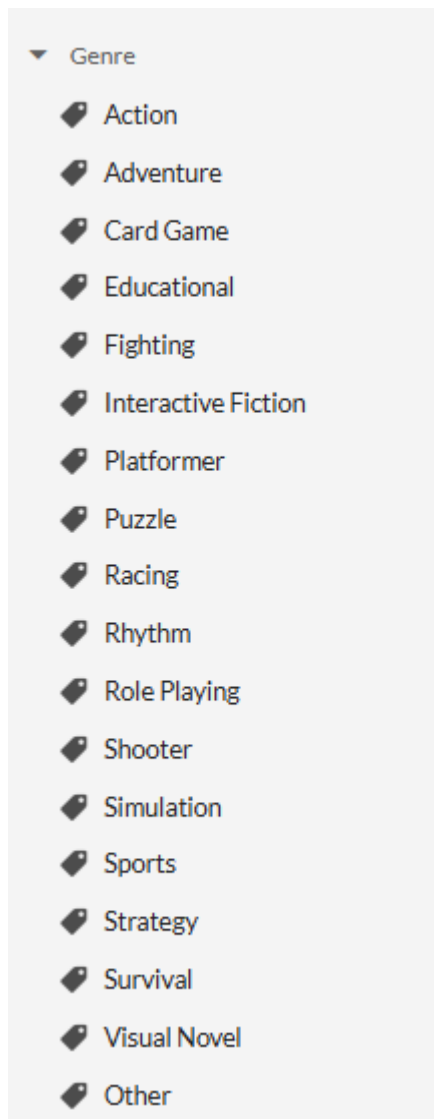
Genero
-nome:String
+setNome():void
+getNome():String

App.java

Genero.java X

src > Genero.java > Java > Genero > nome

```
1 public class Genero {  
2  
3     private String nome;  
4  
5     public Genero(String nome) {  
6         this.nome = nome;  
7     }  
8  
9     public String getNome() {  
10        return nome;  
11    }  
12  
13    public void setNome(String nome) {  
14        this.nome = nome;  
15    }  
16  
17 }  
18
```

18 objetos

Prepare-se para a aula 3 (aula invertida)

- **Criando os objetos jogos**
- Minha primeira **classe**
- Meu primeiro **construtor**
- Usando gets e sets , qual o motivo?
- Meu primeiro commit
- Meu primeiro push
- Criando 5 objetos.

Pesquise sobre o que é classe, para que serve e como fazer

Usando get e set e o encapsulamento

Usando os principais comandos git

Git init

Git add

Git commit

Git status

Git log

Git log oneline

Git push

O seu projeto faz o que?

Quais tecnologias?

Como fazer o README.md bonito, eficaz e útil.