

# **Descomplicando Git e GitHub na prática**



# Sumário

1. Git vs. GitHub
2. GitFlow
3. Conceitos principais
4. Comandos
5. Na prática
6. Notebook no GitHub
7. **Bônus:** extensões VS Code
8. Referências





# Git vs. GitHub

## Git

Sistema para **controle de versionamento de código**, utilizado em projetos para desenvolvimento de software.

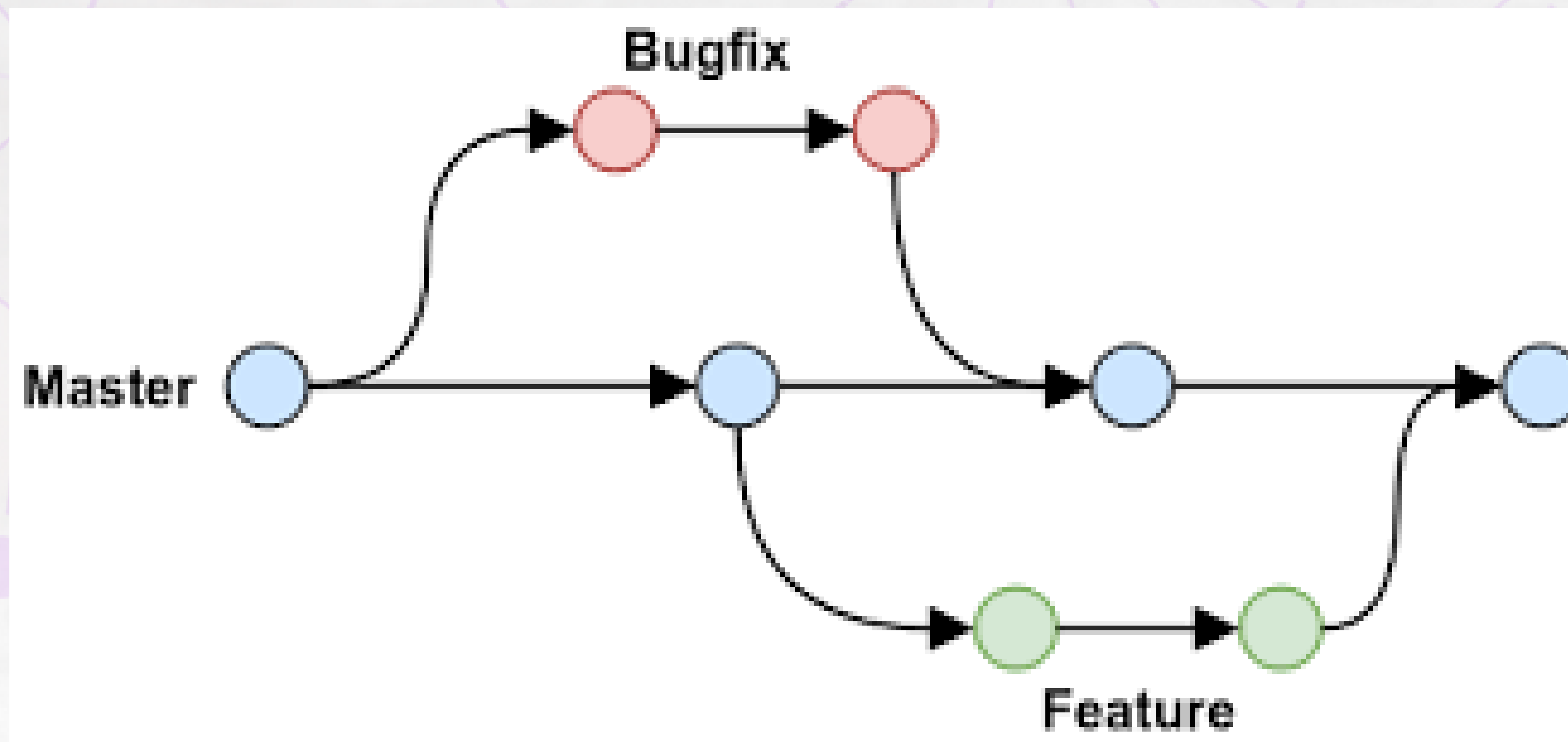
## GitHub

Local onde **armazenamos o nosso código-fonte**, provido de uma interface que apresenta todas as **modificações** realizadas por cada integrante durante o projeto.



# GitFlow

Fluxo de trabalho para que o **Git** seja utilizado por **diversos desenvolvedores** simultaneamente de forma organizada no mesmo projeto.





# Conceitos Principais

## Repositório

É um **armazenamento virtual** para projetos.

## Branch

**Ramificação** em que estamos realizando nossas modificações.

## Merge

Quando **unimos** uma branch à outra, comparando as mudanças.



# Conceitos Principais

## Push Request

Quando **mandamos** nossas modificações para o repo do GitHub.

## Pull Request

Quando queremos **receber** as atualizações feitas no repo.

## Fork

**Cópia** de um repositório para o seu perfil no GitHub..





# Conceitos Principais

## Branch: master

Branch principal do repositório

## Branch: feature

Quando queremos acrescentar algo novo ao projeto

## Branch: fix/hotfix

Modificação para consertar um erro na execução do código



# Comandos

## `git init`

Inicializa um diretório existente como repositório git

## `git clone <url>`

Faz uma **cópia a partir da URL** de um repositório para o seu computador (repositório local)





# Comandos

**git add** <arquivo> ou .

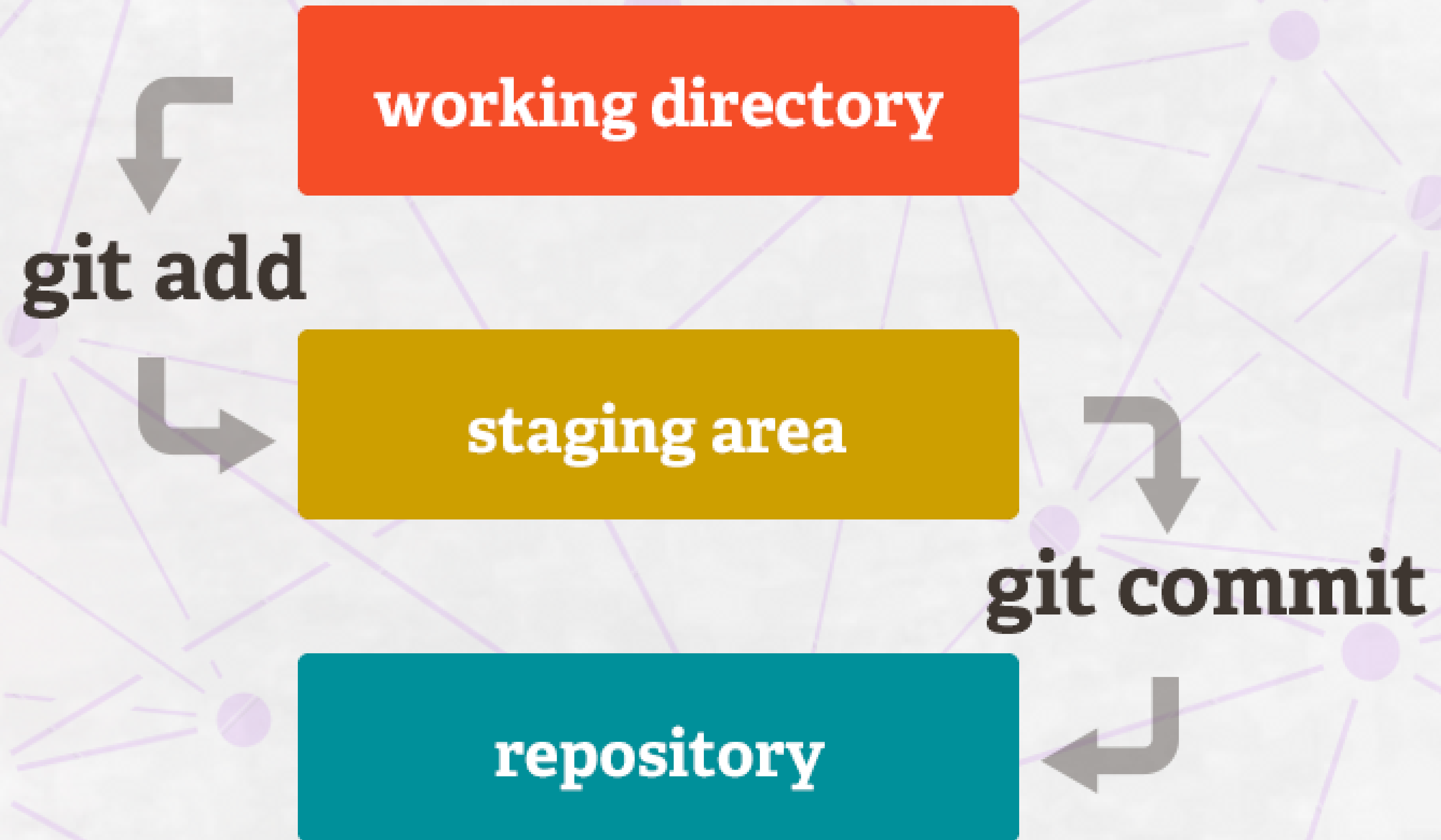
Adiciona os **arquivos modificados** para o que chamamos de **staging area**.

**git commit -m "<descrever mensagem>"**

Após conferidas as modificações na staging area, acrescentamos **uma mensagem para descrever aquela modificação**.



# Comandos





# Comandos

## git branch

Lista todas as branches existentes

## git checkout <nome da branch>

Muda para a branch com o nome indicado

## git checkout -b <nome da branch>

cria uma **nova branch** a partir da branch de origem e muda o diretório de trabalho para ela



# Outros comandos





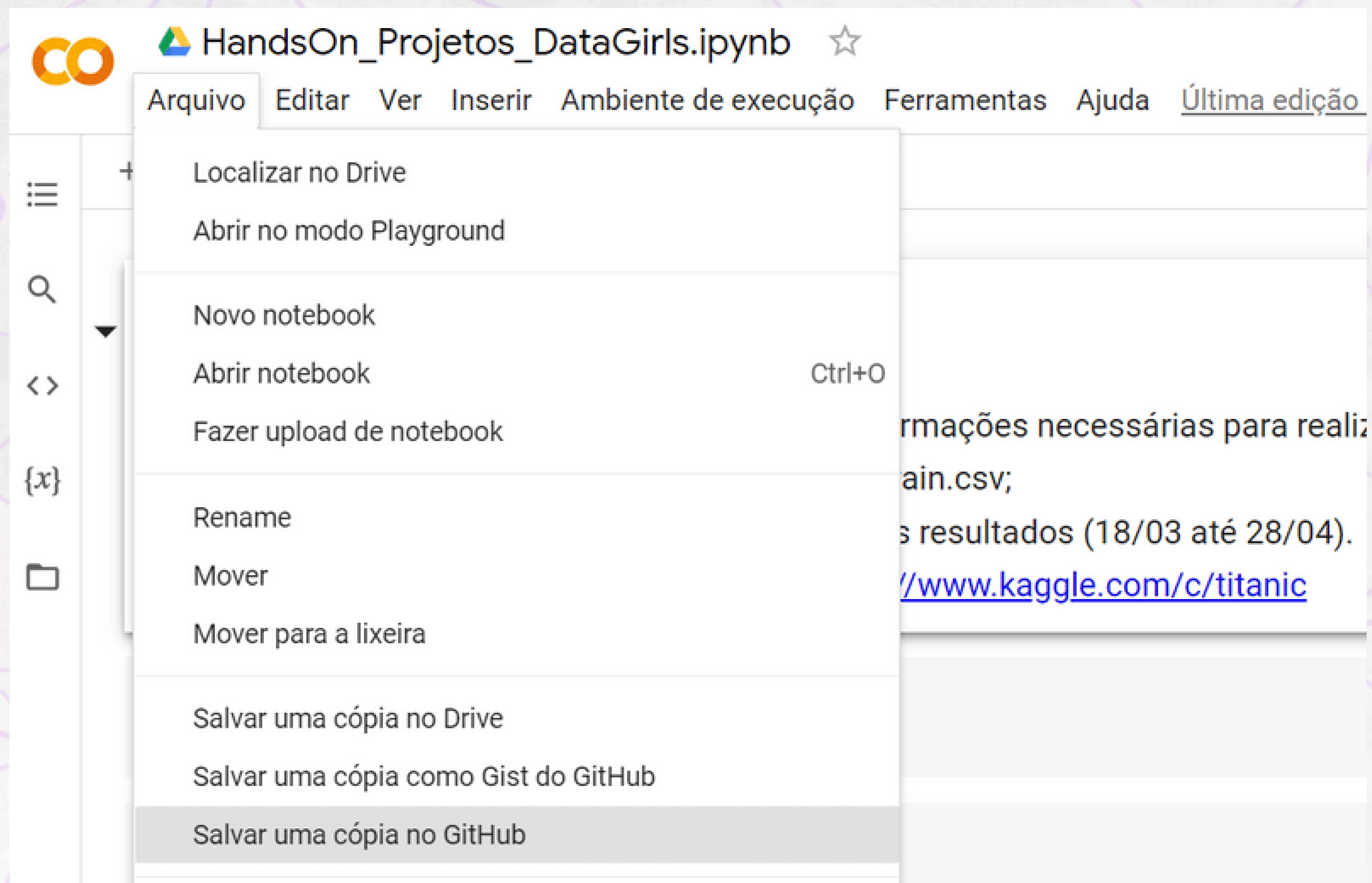
# Na prática

## Mini projeto

1. Criar um **repositório** no GitHub
2. Fazer o **clone** do projeto para o Visual Studio Code do computador
3. Criar um arquivo principal **main.py** na branch **master**
4. Criar uma branch **develop** a partir da principal
5. Adicionar **feature**
6. **Merge** da **develop** com a **master**



# Notebook no GitHub





# Extensões VS Code



## Git Graph v1.30.0

mhutchie |  2.072.757 | ★★★★★ (341)

View a Git Graph of your repository, and perform Git

[Desabilitar](#) ▼ [Desinstalar](#) ▼ 

Esta extensão foi habilitada globalmente.



## GitLens — Git supercharged v11.7.0

 GitKraken |  12.166.432 | ★★★★★ (469)

Supercharge the Git capabilities built into Visual Studio Code —

[Desabilitar](#) ▼ [Desinstalar](#) ▼ 

Esta extensão foi habilitada globalmente.



# Referências

## **Utilizando o Git como VCS:**

<https://giuliana-bezerra.medium.com/utilizando-o-git-como-vcs-3614b2754572>

## **Git Workflows:**

<https://programster.org/git-workflows>

## **Documentação Git:**

<https://git-scm.com/doc>







*Dividas?*