

# Introdução: O que é Git e Github?

Git e Github são duas das ferramentas mais poderosas para desenvolvedores. Nessa apresentação, vamos falar sobre o conceito do Git e Github, e como usá-los de maneira eficiente.



# O que é o Git?

- O Git é um sistema de controle de versão distribuído amplamente utilizado na programação de software e no desenvolvimento colaborativo de projetos.
- O Git permite que várias pessoas trabalhem em um projeto simultaneamente, mesmo quando estão fazendo alterações nos mesmos arquivos.



# O que é o GitHub?

O GitHub é uma plataforma online que oferece um ambiente para armazenar, compartilhar e colaborar em projetos de software. Fundado em 2008, tornou-se rapidamente um dos principais pontos de encontro da comunidade de desenvolvedores ao redor do mundo.



# Principais Componentes e Funcionalidades do GitHub:

## Repositórios

No GitHub, os projetos são organizados em repositórios. Cada repositório é um espaço onde os arquivos de um projeto são armazenados.

## Controle de Versão com Git

O GitHub utiliza o Git, um sistema de controle de versão distribuído. Isso significa que ele mantém um histórico completo de todas as alterações feitas em um projeto.

## Colaboração e Contribuição

Uma das maiores forças do GitHub é a sua capacidade de facilitar a colaboração em projetos de código aberto.

## GitHub Actions e CI/CD

A plataforma oferece uma suíte de ferramentas para automação de tarefas, testes e integração contínua/desdobramento contínuo.

# Benefícios do GitHub

## 1 Colaboração Global

Permite que desenvolvedores de todo o mundo trabalhem juntos em projetos, independentemente de sua localização geográfica.

## 3 Portfólio de Projetos

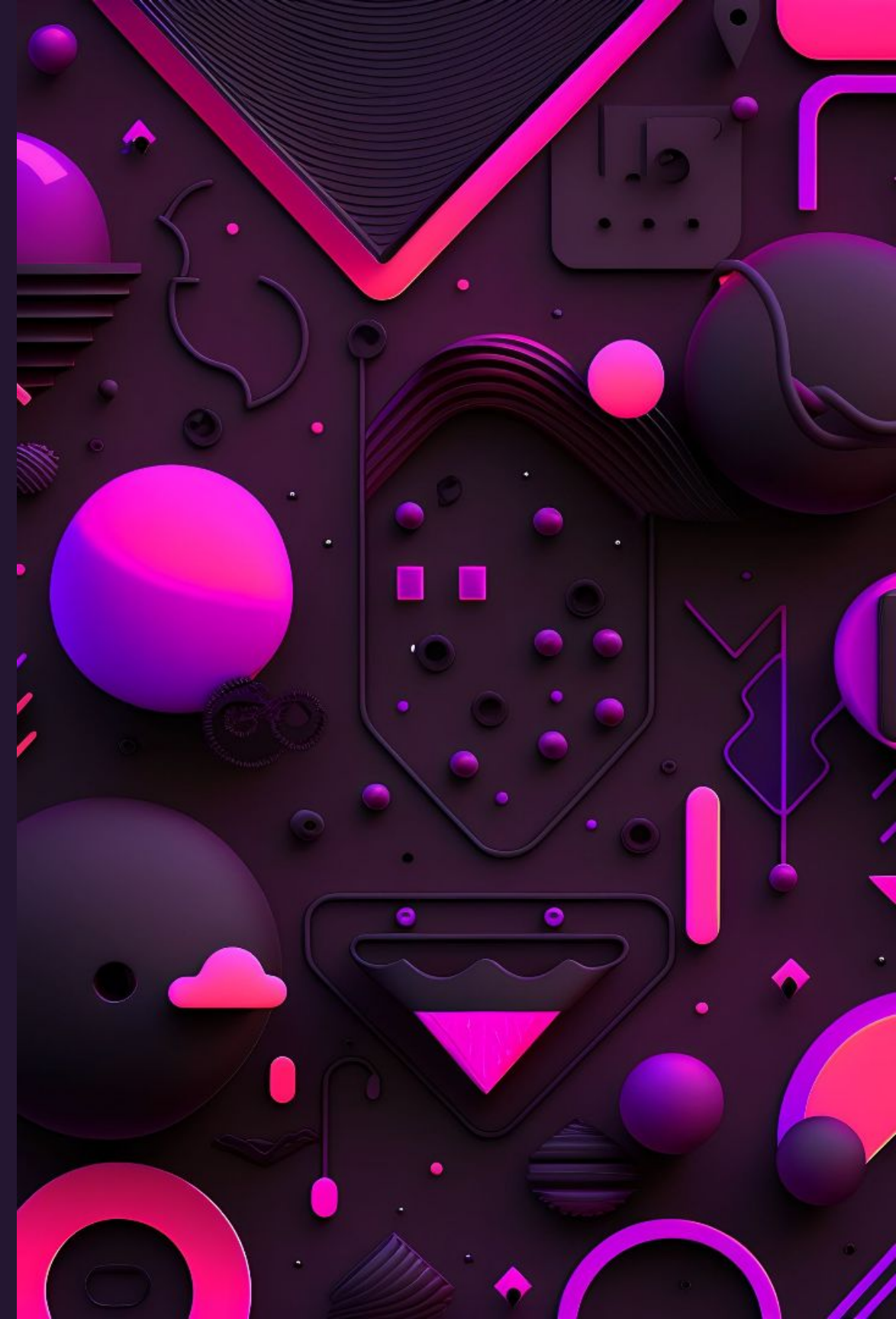
Para os desenvolvedores, o GitHub é uma excelente maneira de mostrar seu trabalho e habilidades para potenciais empregadores.

## 2 Compartilhamento de Conhecimento

GitHub é uma plataforma onde os desenvolvedores compartilham código, soluções para problemas comuns e melhores práticas.

## 4 Melhoria da Qualidade do Código

A revisão de código e a integração contínua ajudam a manter a alta qualidade do código.



# Como inicializar um repositório

## Git init

Para iniciar um novo repositório, basta digitar "git init" no terminal. O git vai inicializar o repositório na pasta atual.

## Github Desktop

Outra maneira de iniciar um repositório é através do Github Desktop. Basta clicar em "Create a new repository" e seguir as instruções.

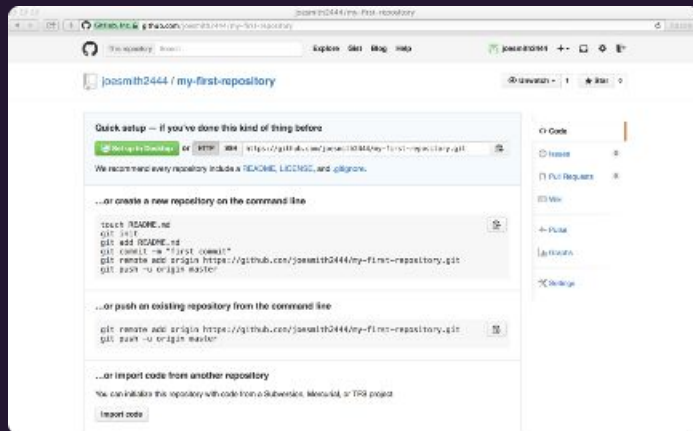
## IDEs

Algumas IDEs, como o Visual Studio Code, permitem criar um novo repositório diretamente na interface.

## Templates

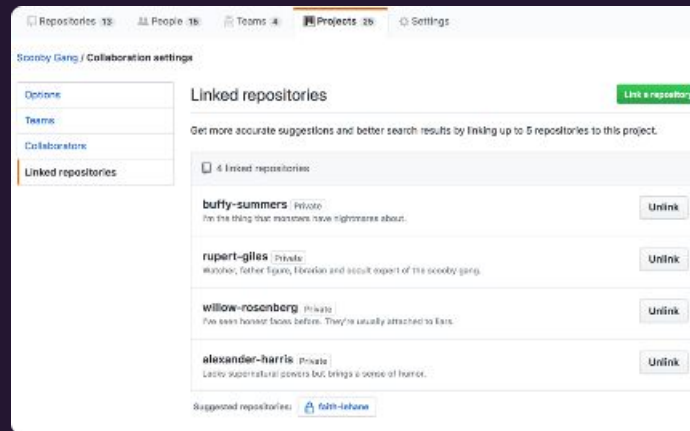
Finalmente, no GitHub é possível criar um novo repositório através de templates, basta escolher o modelo desejado e preencher as informações.

# Criando um repositório no Github



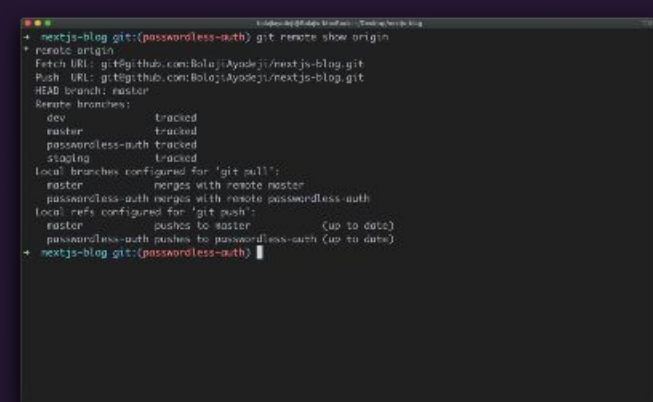
## Criando um repositório

Para criar um novo repositório no GitHub, basta clicar em "New" e preencher as informações básicas do projeto.



## Copiando o link

Após criar o repositório, o próximo passo é copiar o link para o seu computador. No GitHub, basta clicar no botão "Code" e escolher o método de preferência.



## Clonando o repositório

Com o link copiado, é possível clonar o repositório utilizando o comando "git clone". Isso faz com que todos os arquivos sejam baixados para a sua máquina.



# Fazendo commit

## 1 `git add`

Para adicionar um arquivo ao commit, é preciso executar o comando "git add" seguido do nome do arquivo.

## 3 `git status`

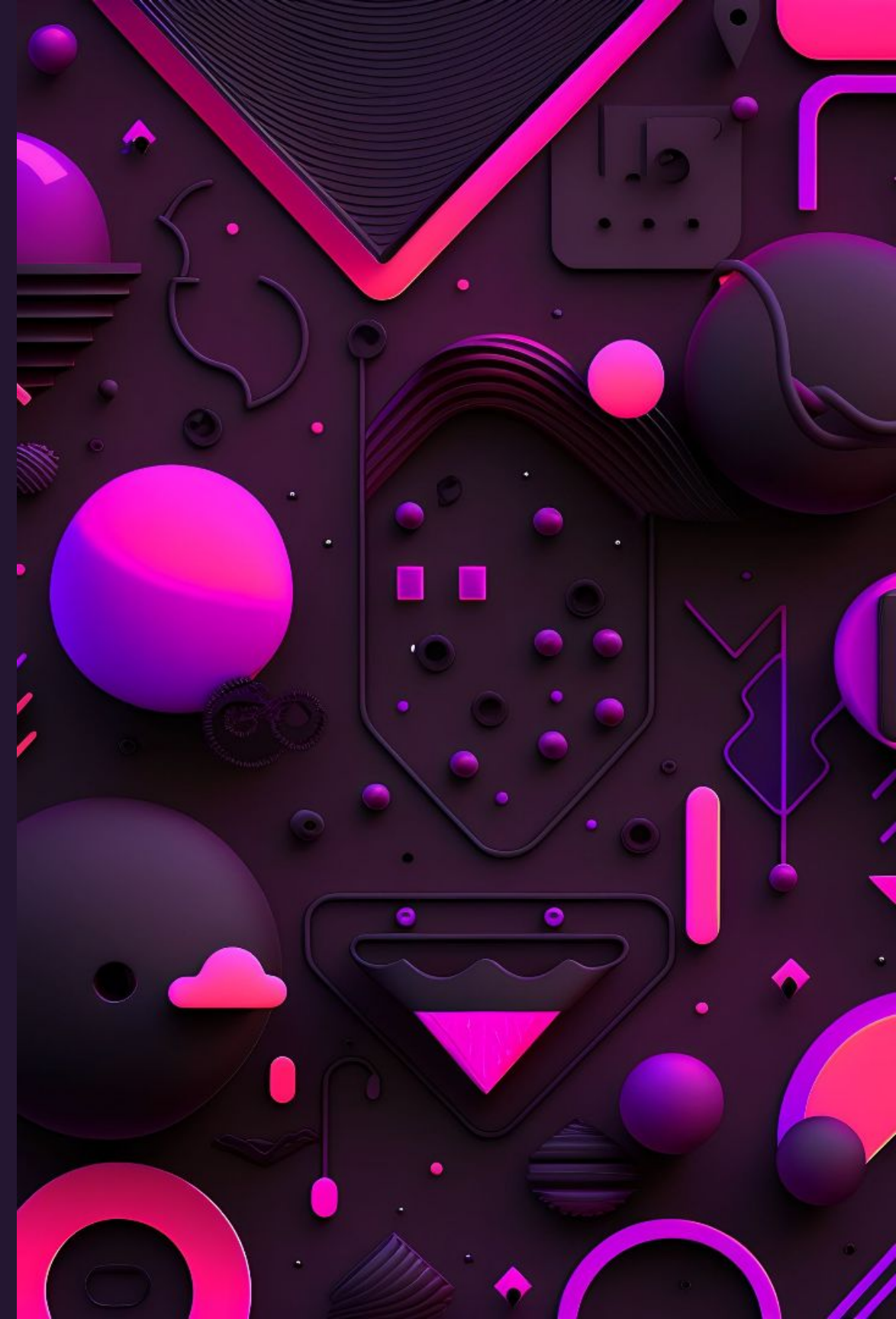
Para verificar o status atual do repositório, é possível utilizar o comando "git status". Ele mostra quais arquivos foram modificados, adicionados ou removidos.

## 2 `git commit`

Depois de adicionar os arquivos desejados, é hora de fazer o commit. Basta executar o comando "git commit" seguido de uma mensagem descritiva.

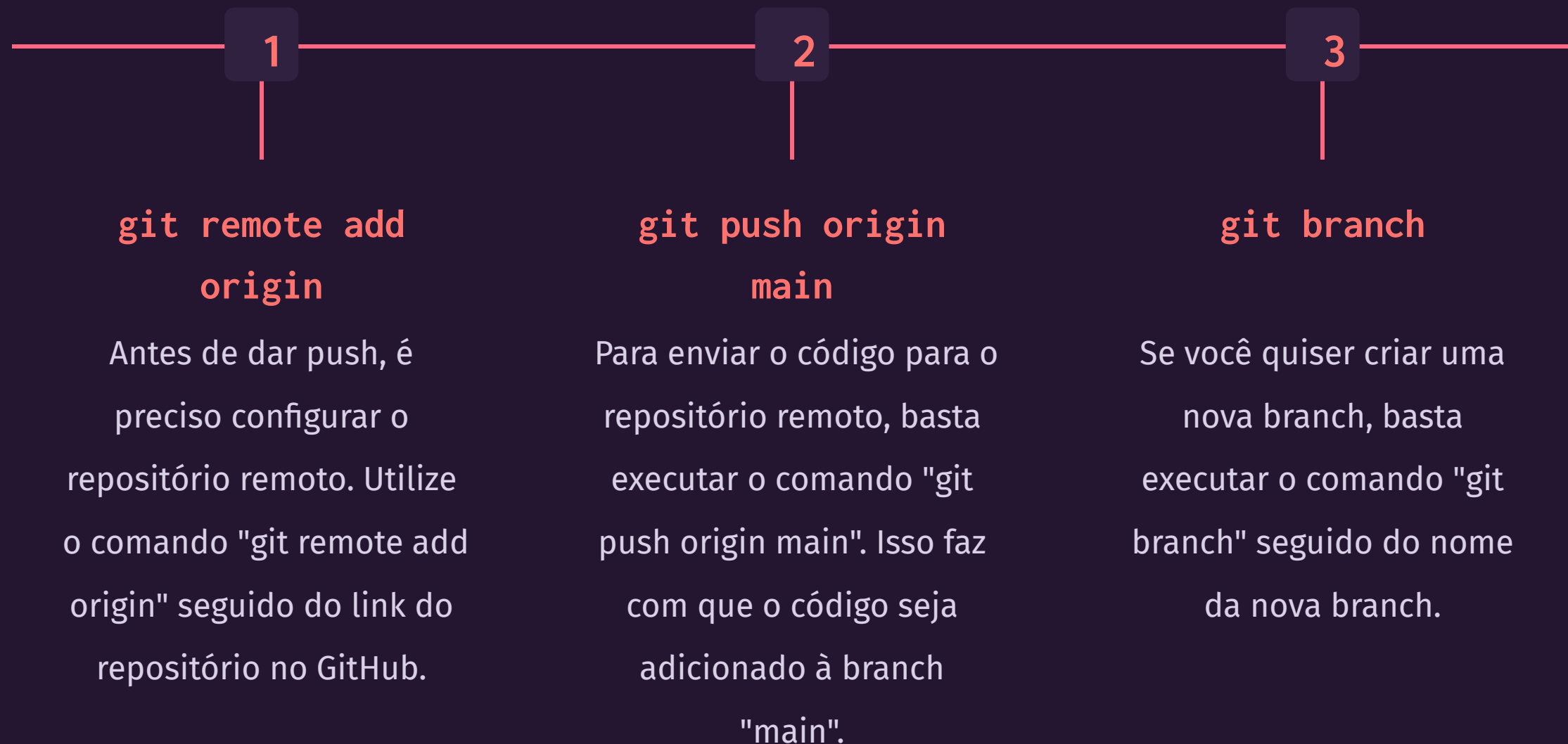
## 4 `git diff`

Se você quiser ver as diferenças entre o commit atual e o último, utilize o comando "git diff".





# Dando push: enviando o código para o repositório



# Clonando um repositório

## Git clone

Utilizando o comando "git clone", é possível baixar uma cópia de um repositório existente para a sua máquina.

## Github Desktop

O Github Desktop também permite clonar um repositório existente através da interface.

## Código-fonte

Finalmente, é possível fazer o download do código-fonte diretamente do GitHub em formato .zip.

# Conclusão e próximos

## Experimente!

Esses foram os conceitos básicos do Git e GitHub. Explore todas as funcionalidades e descubra o que funciona melhor para você.

## Compartilhe!

Compartilhar seu código no GitHub pode ser uma ótima maneira de construir uma presença digital e colaborar com outros desenvolvedores.

## Estude!

Aprender Git é apenas o começo. Continue estudando e acompanhando as tendências para se tornar um desenvolvedor ainda melhor.