

Notas Técnicas - Bioinformática

Prof. Dr. Gilderlanio Santana de Araújo
gilderlanio@gmail.com

25 de julho de 2023

Conteúdo

1	GitHub	5
1.1	Iniciando com o Git	5
1.2	Criar uma conta	5
1.3	Criando um repositório	6
1.4	Clonando um repositório	6
1.5	Efetuando alterações e <i>commits</i>	6
1.6	Enviando alterações para o GitHub	7
1.7	Alternando entre <i>branches</i>	7

CAPÍTULO 1

GitHub

1.1 Iniciando com o Git

O GitHub é uma plataforma web que oferece recursos de controle de versão de forma colaborativa para projetos de desenvolvimento de software. Ele permite que desenvolvedores trabalhem em projetos individualmente ou em equipe, acompanhem as alterações no código e gerenciem repositórios de projetos.

Esta tutorial é apenas uma visão geral básica das funcionalidades do GitHub. Existem muitos recursos e conceitos avançados para explorar, como colaboração com outras pessoas, gerenciamento de problemas e uso do GitHub Pages para hospedar sites. Recomendo que você explore a documentação oficial do GitHub e outros recursos para se aprofundar na plataforma.

1.2 Criar uma conta

- Acesse o site do GitHub (<https://github.com>) e cadastre-se para obter uma conta gratuita.
- Escolha um nome de usuário, insira seu endereço de e-mail e defina uma senha forte.

1.3 Criando um repositório

- Após fazer login em sua conta, clique no botão "New" na página principal para criar um novo repositório.
- Dê um nome ao seu repositório, uma breve descrição e escolha se deve ser público ou privado. Geralmente o nome do repositório é o nome do teu projeto pra que fique mais intuitivo.
- Para descrever teu projeto, você também pode inicializar o repositório com um arquivo README.md, que serve como a página principal do seu projeto. Este arquivo suporta codificação por

1.4 Clonando um repositório

- Se o repositório já existe, você pode cloná-lo em sua máquina local;
- Na página do repositório no GitHub, clique no botão "Code";
- Copie a URL do repositório (com final .git), abra o terminal ou prompt de comando e execute o seguinte comando:

```
git clone url-do-repositorio
```

- O comando fará o download de uma cópia do repositório em sua máquina local.

1.5 Efetuando alterações e *commits*

- Depois de ter o repositório em sua máquina local, você pode fazer alterações nos diretórios e nos arquivos.
- Modificações, inserções e exclusão de arquivos pode ser feito usando o seu editor de texto padrão ou a sua IDE (Pycharm, Jupiter, Eclipse, Netbeans) preferida.
- Após ter realizado alterações, você deve efetuar o commit delas.
- Abra o terminal, navegue até o repositório local e execute os seguintes comandos:

```
git add .
```

```
git commit -m "Deixe uma mensagem clara de suas alterações aqui"
```

1.6 Enviando alterações para o GitHub

- Após efetuar o commit de suas alterações na máquina local, é necessário enviar para o GitHub
- As alterações são enviadas com o seguinte comando:

```
git push origin nome-do-branch
```

- O comando enviará suas alterações para o repositório do GitHub.

1.7 Alternando entre *branches*

O gerenciamento de branches permite que você trabalhe em novos recursos do seu software e correções de erros sem afetar o código principal (main).

Para criar um novo branch, execute o seguinte comando:

```
git branch nome-do-branch
```

Para alternar entre branches:

```
git checkout nome-do-branch
```

Após realizar alterações no novo branch, faça commit e envie-as. Para mesclar suas alterações no código principal, você pode abrir um *pull request* no GitHub. Acesse a página web do repositório, clique na aba "Pull requests" e depois no botão "New pull request". Selecione o branch base (normalmente "main") e o branch com suas alterações. Sempre forneça uma breve descrição. Revise as alterações e clique em "Create pull request".