

**Programa de Pós-graduação em Computação Aplicada (CAP)**

Disciplina: CAP-419-3 Introdução à Programação com Dados Geoespaciais

Professores: Gilberto Ribeiro de Queiroz e Thales Sehn Korting

Aluna: Marcelly Homem Coelho

Contato: [marcellyhc@gmail.com](mailto:marcellyhc@gmail.com)

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**Exercício 04.** Crie um repositório com alguns arquivos texto e envie o link desse repositório para os professores da disciplina, com um resumo da sequência de comandos git utilizados.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

São José dos Campos – São Paulo, 10 de março de 2021.

Inicialmente, foi criado um repositório dedicado à disciplina de Introdução à Programação com Dados Geoespaciais. O repositório foi renomeado como “[CAP-419](https://github.com/marcellyhc/CAP-419)”. A sequência de passos executados foram:

* **Clonar o repositório CAP-419:** Neste tópico, foi realizada uma cópia de repositório remoto para o computador pessoal. Para isso, foi criada uma pasta, com o nome “CAP-419” na máquina local. Na sequência, foram executados os seguintes comandos no Prompt do Windows para acessar a pasta criada:

cd OneDrive

cd “Área de Trabalho”

cd “CAP-419”

cd “CAP-419”

Para efetuar o clone utiliza-se o seguinte comando:

git clone https:/github.com/marcellyhc/CAP-419

Para listar os arquivos existentes dentro da pasta podemos utilizar o comando:

dir

Para listar todos os arquivos existentes, inclusive os ocultos, podemos utilizar o comando:

dir /ah

* **Verificar o status do repositório:** Para encontrar como está o repositório em relações às modificações feitas, podemos utilizar o comando:

git status

* **Modificar um arquivo no repositório:** Após modificarmos um arquivo no repositório, podemos desfazer a modificação ou atualizar o arquivo no repositório, utilizando os seguintes comandos, respectivamente:

git checkout -- NomeArquivo.tipo

git add -- NomeArquivo.tipo

**Observação:** Para adicionarmos vários arquivos que tiveram mudanças, ou invés de adicionarmos um por um, podemos utilizar o comando:

git add .

Para confirmar e salvar todas as modificações realizadas até o momento dos arquivos do nosso repositório utilizamos:

git commit -m “Primeira versão do arquivo NomeArquivo.tipo”

* **Sincronizar a pasta local com o repositório remoto:** Após feitas as alterações/atualizações no repositório local, podemos sincronizá-las com a cópia do GitHub. Primeiro, devemos garantir que a cópia local esteja atualizada com a remota usando:

git pull

Na próxima etapa, utilizamos o comando push para atualizar o repositório remoto com as modificações locais:

git push

**Observação:** Após executar o comando push informeo **usuário** e a **senha** da sua conta GitHub.

Ao final, atualize a página do GitHub e verifique as mudanças no repositório remoto.

* **Adicionar um novo arquivo no repositório:** Após criar um novo arquivo e salvá-lo na pasta local ‘CAP-419’, o próximo passo é incluí-lo no repositório remoto. Podemos fazer esta operação executando:

git add NomeArquivo.TipoArquivo

Para confirmar a inclusão desse arquivo no repositório local podemos utilizar:

git commit -m “Adicionando um novo arquivo”

Para levar as modificações do repositório local para o repositório remoto no GitHub, podemos utilizar o comando:

git push

**Observação:** Após executar o comando push informeo **usuário** e a **senha** da sua conta GitHub.

Ao final, atualize a página do GitHub e verifique as mudanças no repositório remoto.