**Adicionando a pasta ao git e depois github**

git init

git add README.md

git commit -m "first commit"

git branch -M main

git remote add origin https://github.com/diegosantss/CompassUOL.git

git push -u origin main

**Adicionando ao github um repositório já criado**

git remote add origin https://github.com/diegosantss/CompassUOL.git

git branch -M main

git push -u origin main

**Remover origin:**

Remote -v

Remote rm origin

**Add**:

Git add . = adiciona todos os arquivos

Flag -m = mandar uma mensagem no commit

Git commit -a = faz commit em todos os arquivos

**Puxar arquivos atualizados da origin:**

Git pull

**Clonar repositório:**

Git clone “https” .

**Entrar no log do git para verificar as alterações:**

Git log

**Mover ou renomear arquivo de local:**

Git mv style.css css/styles.css - moveu o arquivo ‘style.css’ para dentro da pasta ‘css’

Git mv css/style.css css/stylesRename.css - renomeou o nome do arquivo

**Ignorando arquivos:**

Criar o arquivo .gitignore no projeto e adicionar os nomes das pastas que irá ignorar

**Resetar as alterações e retornar ao original do repositório:**

Git checkout teste.html

**Resetar uma branch já commitada porém não feito o push**

Git reset –hard origin/main

**Visualizar uma Branch:**

Git branch

**Criar uma Branch:**

Git branch <nome>

**Deletar uma branch:**

Git branch -d or –delete <branch>

**Mudar de branch:**

Git checkout -b <nome> - mudar de branch e criar a mesma

Git checkout <nome> - mudar para uma branch existente

**Unir branch:**

Git merge <nome>

**Deletar modificações(podendo ser recuperadas, ficando na área de transf.)**

Git stash

**Recuperar a Stash:**

Git stash list

Git stash apply<numeroStash>

Git stash show -p = mostra o que exatamente foi modificado

**Remover Stach**

Git stash drop <numeroStash>

**Criar tag (criar marcos no desenvolvimento)**

Git tag -a <nome> -m “<msg>”

**Listar e trocar de tags:**

Git show <nome> = lista

Git checkout <nome> = troca

**Enviar Tags:**

Git push origin <nome>

Git push origin - - tags = enviar todas

**Atualizar branchs e tags do repo:**

Git fetch

**Criar submódulo:**

Git submodule add <repo>

Git submodule = verificar os submódulos existentes

**Atualizar submódulos:**

Git push – recurse-submodules+on-demand

**Exibir diferenças de branch:**

Git diff

**Verificar log resumido:**

Git shortlog

**Limpar arquivos untracked:**

Git clean

**Otimizar repositório:**

Git gc

**Checar integridade dos arquivos:**

Git fsck

**Mapear todos os passos do repositório:**

Git reflog

**Transformar o repositório para arquivo:**

Git archive –format zip –output master\_files.zip master

**Markdown:**

Titulos: #(h1), ##(h2) ...

Ênfase: \*\*negrito\*\* ou \_\_texto\_\_

Itálico: \*texto\* ou \_texto\_

Combinado: \_um \*\*texto\*\* combinado\_

Listas:

Ordenadas - \* item

Não ordenadas – 1. Item

Imagens: ![Texto Alt](link da imagem)

Imagem com link: [![Texto Alt](link da imagem)

](link)

Link: [Texto do link](link)

Código: ```codigo````

Task list(GitHub):

Tarefa concluída: - [x] CSS do rodapé

Tarefa não concluída: -[] CSS da pagina de contatos