|  |  |
| --- | --- |
| **DATA:** 11/06/2024 | **CURSO:** Análise e Desenvolvimento de Sistemas |
| **DOCENTE:** Felipe Eduardo Christ | **Disciplina:** Arquitetura e Modelagem de Sistemas |

**GIT**

|  |  |
| --- | --- |
| **RESPOSTAS** | |
| **6** |  |
| **8** |  |
| **9** |  |
| **11** |  |
| **13** |  |
| **15** |  |
| **16** |  |
| **19** |  |
| **21** |  |
| **23** |  |
| **26** |  |
| **27** |  |

1. Crie uma pasta com o nome “aula-asis-2024-local”
2. Transforme essa pasta eu um repositório git (git init)
3. Defina seus dados de usuário e e-mail do repositório

|  |
| --- |
| *git config user.name "<seu-nome>"*  *git config user.email seu@email.com* |

1. Crie um arquivo chamado *index.html* nesta pasta
2. Adicione o seguinte conteúdo ao arquivo:

|  |
| --- |
| *<!DOCTYPE html>*  *<html lang="pt-br">*  *<head>*  *<title>Título da página</title>*  *<meta charset="utf-8">*  *</head>*  *<body>*  *<h1>Página feita por <seu-nome></h1>*  *</body>*  *</html>* |

1. Verifique com a situação do arquivo no repositório (*git status*). Qual a situação do arquivo?
2. Adicione esse arquivo no repositório (*git add*.....)
3. Verifique com a situação do arquivo no repositório (*git* *status*). Qual a situação do arquivo?
4. Crie um *commit* para essa posição atual do repositório (*git* *commit*....). Qual a situação do repositório?
5. Crie uma pasta chamada “*imagens*” e adicione duas imagens de bandas de sua preferência.
6. Verifique com a situação do repositório (*git* *status*). Qual a situação?
7. Adicione essa pasta criada ao repositório.
8. Verifique com a situação do repositório (*git* *status*). Qual a situação?
9. Adicione diretamente as duas imagens com git add imagens/\*
10. Verifique com a situação do repositório (*git* *status*). Qual a situação?
11. Crie um *commit* para essa posição atual do repositório (*git* *commit*....). Qual a situação do repositório?
12. Abra uma das imagens das bandas no paint e faça alguma alteração nas imagens (nome, caretas etc)
13. Verifique com a situação do repositório (*git* *status*). Qual a situação?
14. Crie um *commit* para essa posição atual do repositório (*git* *commit*....). Qual a situação do repositório?

*O git não versiona imagens da mesma forma de arquivos de texto, você precisará dar um git add novamente para o arquivo de imagem que mudou, lembre-se que ele versiona “código fonte/texto”.*

1. Adicione essa imagem alterada novamente ao repositório.
2. Crie um *commit* para essa posição atual do repositório (*git* *commit*....). Qual a situação do repositório?
3. Abra o arquivo index.html e altere o código para exibir as duas imagens usando a tag <img>
4. Utilize o comando *git diff* para verificar se existem mudanças no arquivo. O que foi exibido?
5. Verifique as alterações no repositório e faça um novo commit com essa alteração.
6. Crie um arquivo chamado senhas.txt e escreva nele alguma informação “secreta”. Verifique o status do repositório. *Como não queremos controlar a versão e nem enviar para um repositório remoto no futuro, vamos inserir esse arquivo na lista de ignorados. Chaves de API, dados de conexão com o banco de dados, endereços de diretórios de rede, servidores etc, podem abrir brechas de segurança.*
7. Na pasta raiz do repositório, crie um arquivo chamado .*gitignore* e nele escreva senhas.txt, verifique agora o status do repositório. O que aconteceu?
8. Adicione o arquivo .*gitignore* no repositório e faça um novo *commit*. Qual a situação do repositório?
9. Adicione mais 3 arquivos HTML com algum conteúdo no repositório e insira um deles para ser ignorado. Faça um novo *commit* para finalizar.
10. Abra o GitHub Desktop e faça o login
11. Escolha a opção **“Adicionar um repositório local existente no seu disco local”** e localize a pasta que estávamos trabalhando.
12. Publique esse repositório no Github com o nome “aula-asis-2024-<seu-nome>” de **forma pública**.
13. Verifique no site do Github se o seu repositório apareceu na lista de repositórios na sua conta. Navegue pelo repositório e confirma se os arquivos estão no local.
14. Verifique se o arquivo de senhas está no diretório.
15. Verifique se existem versões diferentes da imagem que você alterou localmente.
16. Por meio do site, adicione as informações no arquivo de README, com uma pequena descrição do projeto e faça um *commit*.
17. Localmente, sem sincronizar com o repositório remoto, crie uma pasta chamada scripts e adicione algum arquivo de código Javascript como por exemplo jquery ou bootstrap. Faça um commit e verifique a mensagem que será exibida. Ao alterar o repositório local e remoto ao mesmo tempo, sem fazer a devida sincronização antes do commit, você verá a mensagem abaixo:

|  |
| --- |
|  |

1. Escolha a opção fetch para primeiro sincronizar o repositório remoto com o local, para depois fazer seu commit e assim fazer o envio do repositório local para o remoto.
2. Localmente, por linha de comando, vamos criando uma branch para uma versão até o ponto que chegamos no código. *git checkout -b "Versao-1-top-das-galaxias"*
3. Utilizando o Github Desktop, envie essas alterações ao repositório remoto. Verifique no site como ficaram os *branchs* do repositório.
4. Compartilhe o link de seu repositório com algum coleguinha e ele deverá no seu repositório, criar um arquivo chamado <nome-do-coleguinha>.txt com o texto, <nome-do-coleguinha> esteve aqui e contribuiu com esse projeto. Precisa ficar registrado que foi ele que fez essa alteração.
5. O coleguinha também deverá entrar no repositório pelo site e registrar um “issue” no repositório. Uma issue é um registro de tarefas a serem feitas, um registro de bug, solicitação de recursos, um report de vulnerabilidade etc.

**ENTREGA PELO AVA**

|  |
| --- |
| Arquivo de texto único contendo:   * Respostas da página 01 * Link do seu repositório * Link do repositório do coleguinha |