

GITHUB

Questão 01. O que é o git? Como você definiria o github?

1º passo: Criar uma conta no github.com.

2º passo: Criar um repositório (no site através de sua conta no github.com)

3º passo: Abrir um terminal no Ubuntu

- 3.1 Aqui você pode experimentar 2 comandos do linux:
 - pwd (*visualizar o diretório local*)
 - ls (*listar conteúdo do diretório local*)
- 3.2 Clonar o seu repositório: git clone
<<https://github.com/seu-usuario/seu-repositorio>>
 - Experimente usar o ls agora

Questão 02. O que você pôde notar de diferente? Comente.

- cd (*acessar diretórios*) - você pode usar este comando para acessar seu repositório local
- Experimente usar o pwd agora
- 3.2 Realizando modificações:
 - editar/criar arquivo: nano nome-do-arquivo (*não esquecer de salvar o arquivo - dica: verifique o menu no rodapé. ^ significa ctrl!*).
- 3.3 Enviando seus arquivos para o seu repositório no github:
 - git add nome-do-arquivo: (*adicionando arquivos*)
 - Alternativamente você pode usar git add . ou git add * se quiser adicionar múltiplos arquivos/diretórios
 - git commit -m "minha mensagem": (*confirmar as mudanças com uma mensagem informando o motivo/razão daquele arquivo/alteração*)
 - git push origin master: (envia alterações para seu repositório remoto)

Questão 03. Existe um conceito muito importante no github chamado de *branch* (*ramo em inglês*). Defina branch no âmbito do github. Por que você utilizaria *branches*? Para qual branch você enviou seu arquivo? E se quisesse enviar para um novo branch, o que você faria de diferente?

Dica: Acesse seu repositório remoto do github e verifique as alterações realizadas até o momento no branch master;

- 3.4 Alternando entre branches:
 - git branch: (verificar atual branch e lista de branches existentes)
 - git checkout -b novo_branch: (crie um novo branch chamado "novo_branch" e selecione-o usando). Experimente usar o git branch novamente.
 - Repita 3.2 e 3.3. *Importante!* Altere o branch destino no passo 3.3.

Dica: Acesse seu repositório remoto do github e verifique as alterações realizadas até o momento no novo branch;

- git checkout master: (retorna para o branch master - experimente usar o comando git branch novamente)
- 3.5 Verificando diferenças:
 - git diff master novo_branch

Questão 04. Que tipo de informação o git apresenta de forma que te permita saber que há diferenças entre branches?

- 3.6 Realizando forks:
 - Acesse <https://github.com/ifbahia/redes> e verifique os branches atualmente existentes.
 - Solicite participação na organização <https://github.com/ifbahia>
 - Faça um fork (pelo site) do repositório <https://github.com/ifbahia/redes> para sua conta no github.
 - git clone <https://github.com/<seu-usuario>/redes> -b seu-branch: (clonando o repositório e o branch de acordo com sua turma).
 - Acesse o diretório redes e crie um diretório com seu 1o e último nome separados por underline (mkdir primeiro_ultimo).
 - Acesse o novo diretório criado por você e submeta as questões de 1 a 4 para o seu repositório redes no branch da sua turma.
- 3.7 Realizando Pull Requests:

Dica: Pesquise sobre Pull Requests!

- Acesse <https://github.com/ifbahia/redes> e faça um pull request das suas alterações no branch da sua turma para <https://github.com/ifbahia/redes> no mesmo branch.
 - As suas alterações agora foram submetidas para o repositório redes em ifbahia. Aguarde retorno (aprovação/ou solicitação de alterações) da sua submissão. Se tudo estiver correto, você verá as suas alterações em <https://github.com/ifbahia/redes> no branch da sua turma. Detalhe: correto

significa que tudo ocorreu bem com a sua submissão. Não significa necessariamente que as respostas estão corretas.

A partir de agora você já deve estar apto a submeter respostas através do github. TODAS as nossas atividades e avaliações serão depositadas lá!

Outros comandos que podem ser úteis:

- `git rm meu-arquivo` (remove o arquivo de nome meu-arquivo).
- `git rm -r meu-diretorio` (remove diretório de nome meu-diretorio).
- `git mv arquivo1 arquivo2` (renomeia arquivo1 para arquivo2).