

## Passos para o commit

Ao acessar o histórico de alterações, o desenvolvedor percebeu que houve um commit no arquivo artigos.php que ele estava trabalhando no momento e resolveu checar as diferenças entre o arquivo commitado e o seu arquivo antes de enviá-lo ao stage e efetuar o commit.

Qual das alternativas abaixo mostra em ordem a sequência de comandos executados pelo desenvolvedor?

```
git log
git diff artigos.php
git add artigos.php
git commit -m "alteração em artigos.php"
```

## Diferenças no script

Considerando a lista de pedidos abaixo, escolha a alternativa com os comandos que cumpram seus objetivos corretamente:

- Verificar se houve alterações no diretório;
- Checar as diferenças entre o arquivo index.js do diretório de trabalho e do repositório local;
- Adicionar o arquivo index.js ao stage;
- Publicar as alterações no repositório local.

```
git status
git diff index.js
git add index.js
git commit -m "Javascript"
```

## Projeto origin

Foi solicitado ao desenvolvedor que ele adicionasse um novo repositório remoto chamado "origin" e apontasse para a nova URL do projeto:

<https://github.com/devuser/projeto.git> e em seguida baixasse o conteúdo desse repositório sem que fossem executadas alterações (sem merge).

Complete os campos abaixo com os comandos necessários para essa operação:

```
git remote add origin https://github.com/devuser/projeto.git
git fetch origin
```

## Configurando repositório remoto

Um desenvolvedor se viu na necessidade de configurar um repositório remoto para o seu trabalho.

Considerando as informações fornecidas pelo gerente do projeto na lista abaixo, escreva o comando necessário para configurar o acesso a esse repositório:

- Nome do repositório deve ser experimento;
- A URL do repositório deve ser <https://github.com/devuser/experimento.git>.

```
git remote add experimento https://github.com/devuser/experimento.git
```

## Criando e deletando

Qual das alternativas abaixo cria um novo branch chamado "experimento" e em seguida deleta o branch "developer"?

```
git branch experimento
git branch -d developer
```

## Confirmando exclusão

Um desenvolvedor precisava limpar sua working area de alguns arquivos desnecessários, mas antes disso ele gostaria de verificar quais arquivos seriam excluídos. Qual das alternativas abaixo mostra na ordem correta a execução desses comandos?

```
git clean -n
git clean -f
```

## Verificando o perfil

Levando em consideração a lista de tarefas abaixo, preencha as lacunas com os comandos corretos:

- Verificar se houve alguma alteração no diretório de trabalho;
- Verificar se existem diferenças entre os arquivos 'perfil.java' do diretório de trabalho e do repositório local;
- Adicionar as alterações do arquivo perfil.java ao stage;
- Commitar as alterações com a mensagem "perfil".

```
git status
git diff perfil.java
git add perfil.java
git commit -m "perfil"
```

## Projeto em dois branches

O projeto de uma equipe é dividido em dois branches principais, o release e o development. Após um longo período de desenvolvimento a equipe vai finalmente publicar suas alterações no branch de lançamento (release).

Observando a lista abaixo, escreva os comandos necessários para que o projeto seja publicado no branch release:

- Selecionar o branch chamado release;
- Verificar se existe alguma diferença entre ele e o branch development;
- Mesclar o conteúdo dos dois branches.

```
git checkout release
git diff release..development
git merge development
```

## Objetivos em ordem

Observe a lista abaixo e escolha a alternativa com os comandos que completam todos os objetivos na ordem em que eles aparecem.

- Verificar por alterações na working area;
- Checar a diferença entre todos os arquivos encontrados que não estejam no stage;
- Adicionar todos os arquivos do diretório atual ao stage;
- remover o arquivo style.css do stage sem que ele seja marcado para exclusão;
- Commitar os arquivos no repositório local.

```
git status
git diff
git add .
git restore --staged style.css
git commit -m "Alteração de HTML"
```

## Excluindo arquivos

Durante o processo de desenvolvimento o programador percebeu que havia um arquivo na sua working area que não havia sido comitado. Percebendo que esse arquivo não era necessário, o desenvolvedor decidiu apagá-lo.

Qual dos comandos abaixo foi utilizado para excluir esse arquivo da working area?

```
git clean -f
```

## Comandos corretos

Seguindo as instruções da lista abaixo, complete as lacunas com os comandos corretos:

- Inicializar o git;
- Criar um branch chamado "teste".
- Acessar o branch criado;
- Verificar se houve alguma alteração no diretório de trabalho;
- Adicionar todas as alterações do diretório atual no stage;
- comitar as alterações com a mensagem "teste";
- Acessar o branch master;
- Mesclar com o conteúdo do branch teste.

```
git init
git branch teste
git checkout teste
git status
git add .
git commit -m "teste"
git checkout master
git merge teste
```

## Restaurando e removendo

Após fazer as suas modificações, um desenvolvedor enviou todos os seus arquivos para o stage, mas antes de comitar percebeu que havia alguns arquivos que não deveriam ser comitados. O arquivo style.css devia ser removido do stage e ser deixado na

workinggit checkout master  
git merge desenvolvimento

area, já o arquivo script.js deveria ser marcado para exclusão antes do commit.

Escolha a alternativa que mostra as ações executadas pelo desenvolvedor na ordem em que aparecem.

```
git add .
git restore --staged style.css
git rm --cached script.js
```

```
git commit -m "novo conteudo"
```

## Baixando dados

Complete a lacuna abaixo com o comando necessário para apenas baixar dados de um outro repositório, sem que um 'merge' seja executado:

```
git fetch development
```

## Alterações no branch

Após todas as alterações no branch 'desenvolvimento' serem validadas e aprovadas, foi solicitado ao desenvolvedor publicar essas alterações no branch principal(master). Qual das alternativas abaixo seleciona o branch principal e adiciona o conteúdo do branch 'desenvolvimento' a ele?

```
git checkout master  
git merge desenvolvimento
```

## Analizando os comandos

Analisar os comandos abaixo e escolha a alternativa que descreve suas ações corretamente:

```
git log  
git reset --hard alu4k97c
```

O histórico é acessado para encontrar o hash e em seguida é feita uma reversão para o commit alu4k97c

## Remover do Stage

Enquanto verificava os arquivos do stage para comitar, o desenvolvedor notou um arquivo chamado 'devteste.html' que não era mais necessário para o projeto. Qual comando o desenvolvedor utilizou para remover o arquivo do stage e marcá-lo para exclusão após o commit?

```
git rm --cached devteste.html
```

## Validando as alterações

Um desenvolvedor precisava subir as alterações do arquivo cursos.php, porém, para se certificar de que tudo estava certo, ele verificou primeiro quais alterações foram feitas no novo arquivo e só depois disso o adicionou ao stage e commitou o projeto com a mensagem "cursos".

Escreva abaixo os comandos na ordem em que foram executados:

```
git diff cursos.php  
git add cursos.php  
git commit -m "cursos"
```

## Comitando a home

Complete os espaços em branco com os comandos corretos seguindo a lista de tarefas abaixo:

- Inicialize o git .
- Verifique se existem alterações no diretório de trabalho;
- Adicione todos os itens ao stage;
- Faça o commit com a mensagem "home".

```
git init
git status
git add .
git commit -m "home"
```

### **Excluindo um branch**

Ao analisar um projeto o desenvolvedor notou que havia três branches: Master, Teste e Teste\_B. O desenvolvedor então decidiu apagar o branch Teste\_B, pois estava sem uso. Qual comando necessário para executar essa ação?

```
git branch -D Teste_B
```

### **Lista de pedidos**

Considere a lista de pedidos abaixo e selecione a alternativa com os comandos necessários para cumprir os objetivos na ordem que aparecem:

- Inicializar o git no diretório atual;
- Após as alterações, adicionar todos os arquivos do diretório atual ao stage;
- Comitar as alterações com a mensagem "Release";
- Criar uma branch chamada "Development";
- Selecionar a branch Development.

```
git init
git add .
git commit -m "Release"
git branch Development
git checkout Development
```

### **Configurando o repositório 'experimento'**

Escolha abaixo a alternativa que configura corretamente um repositório remoto chamado 'experimento' que aponte para a URL <https://github.com/devuser/experiencia.git>.

```
git remote add experimento https://github.com/devuser/experiencia.git
```

### **Diferença entre arquivos**

Ao utilizar o git status durante o seu processo de trabalho, o desenvolvedor identificou uma modificação no arquivo 'home.php' da sua working area que não foi feita por ele. Qual dos comandos abaixo o desenvolvedor poderia utilizar para verificar as diferenças entre o arquivo home.php modificado e a última versão desse arquivo commitada no repositório local?

```
git diff home.php
```

### **Apagando um branch**

Um desenvolvedor precisa excluir um branch chamado "teste". Qual dos comandos abaixo vai ajudá-lo a fazer isso?

```
git branch -D teste
```

### **Pedido do gerente**

Um desenvolvedor recebeu um pedido de seu gerente para que fizesse algumas coisas no projeto antes de iniciar suas tarefas do dia.

Escreva abaixo os comandos necessários para cumprir os pedidos da lista:

- Adicionar todos os arquivos modificados da working area ao stage
- remover o arquivo funcao.js do stage sem marcá-lo para exclusão
- comitar as alterações com a mensagem 'home'

```
git add .  
git restore --staged funcao.js  
git commit -m "home"
```

### **Arquivo dados.js**

Durante uma checagem habitual antes do commit, o desenvolvedor percebeu que ainda precisava trabalhar no arquivo 'dados.js' e por isso, ele não entraria nesse commit.

Qual comando o desenvolvedor deve utilizar para remover o arquivo do stage sem que ele seja excluído da working area?

```
git restore --staged dados.js
```

### **Restaurando arquivo**

Após uma alteração no arquivo 'loja.php', um desenvolvedor detectou um problema enquanto o arquivo ainda estava na working area.

Qual comando ele poderia utilizar para restaurar o arquivo para a última versão comitada no repositório local?

```
git checkout -- loja.php
```

### **Repositório externo**

Um desenvolvedor precisava baixar dados de um repositório externo, mas para isso, antes ele precisava configurar esse repositório.

Qual dos comandos abaixo configura um repositório de nome "experimento" para uso?

```
git remote add experimento https://www.meurepositorio.com.br/
```

### **Lista de tarefas**

Ao chegar em sua mesa de manhã, um desenvolvedor encontrou uma lista com algumas ações que deveria executar.

Analisando essa lista, escolha a alternativa que representa corretamente os pedidos descritos nela:

- Checar por alterações no diretório de trabalho;
- Adicionar o arquivo index.html ao stage;
- Remover o restante dos arquivos da working area;
- Comitar as alterações.

```
git status  
git add index.html  
git clean -f  
git commit -m "index.html"
```

### **Adicionando alterações**

Um desenvolvedor precisava executar um processo de rotina que consistia em verificar se houve alguma alteração na branch, adicionar todas as alterações ao stage e comitar tudo para o repositório local.

Qual das alternativas abaixo possui os comandos necessários para essa tarefa?

```
git status
```

```
git add .
git commit -m "rotina"
```

## Remoção de arquivo

Durante o processo de desenvolvimento um programador notou que ao utilizar o comando 'git rm --cached index.html' o arquivo era removido do stage, porém, também era marcado para exclusão após o commit.

Qual comando o desenvolvedor deveria utilizar para remover o arquivo 'index.html' do stage sem que ele seja marcado para exclusão?

```
git restore --staged index.html
```

## Commitando a loja

Complete os campos abaixo com os comandos necessários para cumprir os pedidos da lista de tarefas:

- Selecionar o branch development;
- Verificar as diferenças entre o arquivo loja.php da working area e do repositório local;
- Adicionar o arquivo loja.php ao stage;
- Efetuar o commit com a mensagem "loja".

```
git checkout development
git diff loja.php
git add loja.php
git commit -m "loja"
```

## Lista de tarefas

Um desenvolvedor recebeu uma lista de tarefas que deveria executar. Com base nessa lista, escolha a alternativa com os comandos que cumprem esses objetivos:

- Verificar quais linhas foram alteradas no arquivo master.php;
- Adicionar esse arquivo ao stage;
- Commitar a alteração.

```
git diff master.php
git add master.php
git commit -m "master alterado"
```

## Novo menu

Complete todas as lacunas com os comandos necessários para cumprir a lista de tarefas informada abaixo:

- Verifique se existem alterações no diretório de trabalho;
- Verifique quais arquivos seriam excluídos pelo comando 'clean' sem de fato efetuar a exclusão;
- Adicione todos os arquivos da working area ao stage;
- Faça o commit com a mensagem "menu".

```
git status
git clean -n
git add .
git commit -m "menu"
```

## Diferenças

Um desenvolvedor estava se preparando para commitar mudanças em diversos arquivos no repositório local, porém, antes de fazer isso ele quis lembrar quais eram as diferenças entre os arquivos que estavam no stage e o último commit do repositório.

Qual das alternativas abaixo tem o comando necessário para fazer essa ação?

```
git diff --staged
```

## Objetivos listados

Considere a lista de tarefas abaixo e escolha a alternativa que cumpra com todos os objetivos listados:

- Verificar se existem diferenças entre o branch master e o branch dev;
- Mesclar o conteúdo dos dois branches;
- Verificar se houveram alterações após a junção dos branches;
- Adicionar as alterações posteriores ao stage;
- commitar todas as alterações.

```
git diff master..dev
git merge dev
git status
git add .
git commit -m "Branch unificado"
```

## Revertendo um commit

Após o commit mais recente do projeto um desenvolvedor encontrou um problema com a aplicação onde era necessário reverter o processo para o commit dcd9a84d, onde tudo estava funcionando. Escreva abaixo o comando para reverter as alterações para o commit associado ao hash.

Obs.: Além de alterar o index o comando também deve alterar os arquivos no seu diretório de trabalho (working directory).

```
git reset --hard dcd9a84d
```

## Arquivos desnecessários

Durante o desenvolvimento de um projeto, o desenvolvedor criou alguns arquivos, dentre eles um chamado index.html e outro style.css. Na hora de publicar as alterações no repositório local, o programador percebeu que os demais arquivos criados não eram necessários e poderiam ser limpos do diretório de trabalho.

Seguindo as instruções da lista abaixo, escreva os comandos que o programador utilizou para alcançar seu objetivo:

- Acessar o branch 'dev' já existente;
- Adicionar o arquivo index.html ao stage;
- Adicionar o arquivo style.css ao stage;
- Remover da working area os arquivos desnecessários que não vão ser commitados;
- Commitar todas as alterações com a mensagem "resolvido".

```
git checkout dev
git add index.html
git add style.css
git clean -f
git commit -m "resolvido"
```



## **Perfil novo**

Siga as instruções abaixo e complete as lacunas com os comandos corretos:

- Adicione o arquivo perfil.html ao stage;
- Remova o arquivo file.js do stage sem que ele seja marcado para exclusão;
- Committe as alterações com a mensagem "perfil".

```
git add perfil.html  
git restore --staged file.js  
git commit -m "perfil"
```