


Agora a prática

 app.betrybe.com/course/fundamentals/git/git-part-2/exercicios/agora-a-pratica

⚠ Para fazer os exercícios a seguir, você deve utilizar o projeto com o texto no arquivo `.txt`, localizado dentro do diretório `trybe-exercises/exercises/2.1`, que você criou nas aulas anteriores.

Dessa vez, cada um deve fazer os exercícios em seu computador, ok? Vamos lá! 😊

1. Navegue até a raiz do projeto com o arquivo `.txt`;
2. Verifique se não existe nada sem "*commitar*" utilizando `git status` ;

Caso exista algo, verifique se é necessário e faça o `commit`, ou remova-o.

3. Crie uma nova `branch` com o nome `trybe-skills-changes` e faça checkout nela;
4. No arquivo `.txt`, ao final da lista de habilidades, adicione mais duas habilidades que serão desenvolvidas na Trybe;

Exemplo:

0 que eu vou aprender na Trybe:

- Unix
- Bash
- Git
- HTML
- CSS

* Faça um ``git add nome-do-arquivo.extensao``;

- Você pode adicionar todos os arquivos que você modificou usando `git add .`, mas evite isso em commits com muitos arquivos para não adicionar nenhuma alteração por engano;

Agora um `git commit -m "Mensagem que você gostaria"` ;

- Uma boa prática é sempre resumir o que o seu commit está alterando, por exemplo, `git commit -m "Adiciona nova skill"`

- Evite juntar muitas modificações em um único commit. Assim, caso haja algum erro no código, ficará mais fácil visualizar em qual alteração ele surgiu.
 - E por último um `git push -u origin trybe-skills-changes` ;
 - Abra um *Pull Request* com uma descrição detalhada:
 - Dê contexto para o que você está fazendo, passe links ou cite especificações que forem relevantes. Ex: *"Trabalho feito para a semana 1 do curso de Software Developer da Trybe. Aqui, o desafio foi... E para resolver o problema fizemos... E o resultado foi..."*
 - O merge deve ser feito apenas quando chegar no exercício 10
- 5. Retorne para a branch principal, `master` , com o comando: `git checkout master` ;
- 6. Verifique que você está na branch `master` , com o comando: `git branch` (esta branch deve estar com o formato original, sem as habilidades recém adicionadas);
- 7. Crie uma nova `branch trybe-skills-updates` a partir da `master` e faça checkout nela;
- 8. No mesmo arquivo `.txt` que você modificou no *passo 4* , também ao final da sua lista de habilidades, adicione mais um aprendizado que você terá nos próximos blocos;
 - Atenção! Aqui o arquivo não terá as alterações feitas anteriormente na outra branch 😊
 - Faça um `git add nome-do-arquivo.extensao`
 - Agora um `git commit -m "Mensagem que você gostaria"`
 - e por último um `git push -u origin trybe-skills-updates`
 - Após o primeiro *"push"* da sua branch, você pode usar apenas o comando `git push`
 - Abra um *Pull Request* com uma descrição amigável.

O merge deve ser feito apenas quando chegar no exercício 10

9. No **Slack** , procure pelo canal "*code-review-**" relativo à sua turma, e avise que você chegou a este ponto. O próximo passo será realizar o *Code Review (CR)* do projeto de outras pessoas estudantes. Entre em pelo menos um *Pull Request* e deixe alguns comentários para seus colegas;

10. Agora, faça o **merge** das *branches* **trybe-skills-changes** e **trybe-skills-updates** na *branch* **master** , através do *Pull Request* :

- Primeiro, vá até a página do primeiro *Pull Request* (branch **trybe-skills-changes**) e faça o *merge* clicando no botão verde "**Merge pull request**" ;
 - Agora, vá até página do outro *Pull Request* (branch **trybe-skills-updates**) e tente fazer o *merge* clicando no mesmo botão. Reparou que ele está bloqueado? Isso acontece porque esse *Pull Request* está tentando alterar a mesma linha de código com um conteúdo diferente, e o **Git** não consegue determinar sozinho qual das duas linhas é a correta.
 - Você terá que resolver esse **conflito** antes de "*mergear*" o *Pull Request* . Clique no botão "**Resolve conflicts**" , escolha uma das duas versões do texto (*lembre-se de apagar as linhas com <<<<<< e >>>>>>* , elas são criadas apenas para dar uma identificação mais visual ao problema) . Depois clique em "**Mark as resolved**" e, em seguida, em "**Commit merge**" .
 - Agora você deve conseguir "*mergear*" seu *Pull Request* normalmente. 😎
-