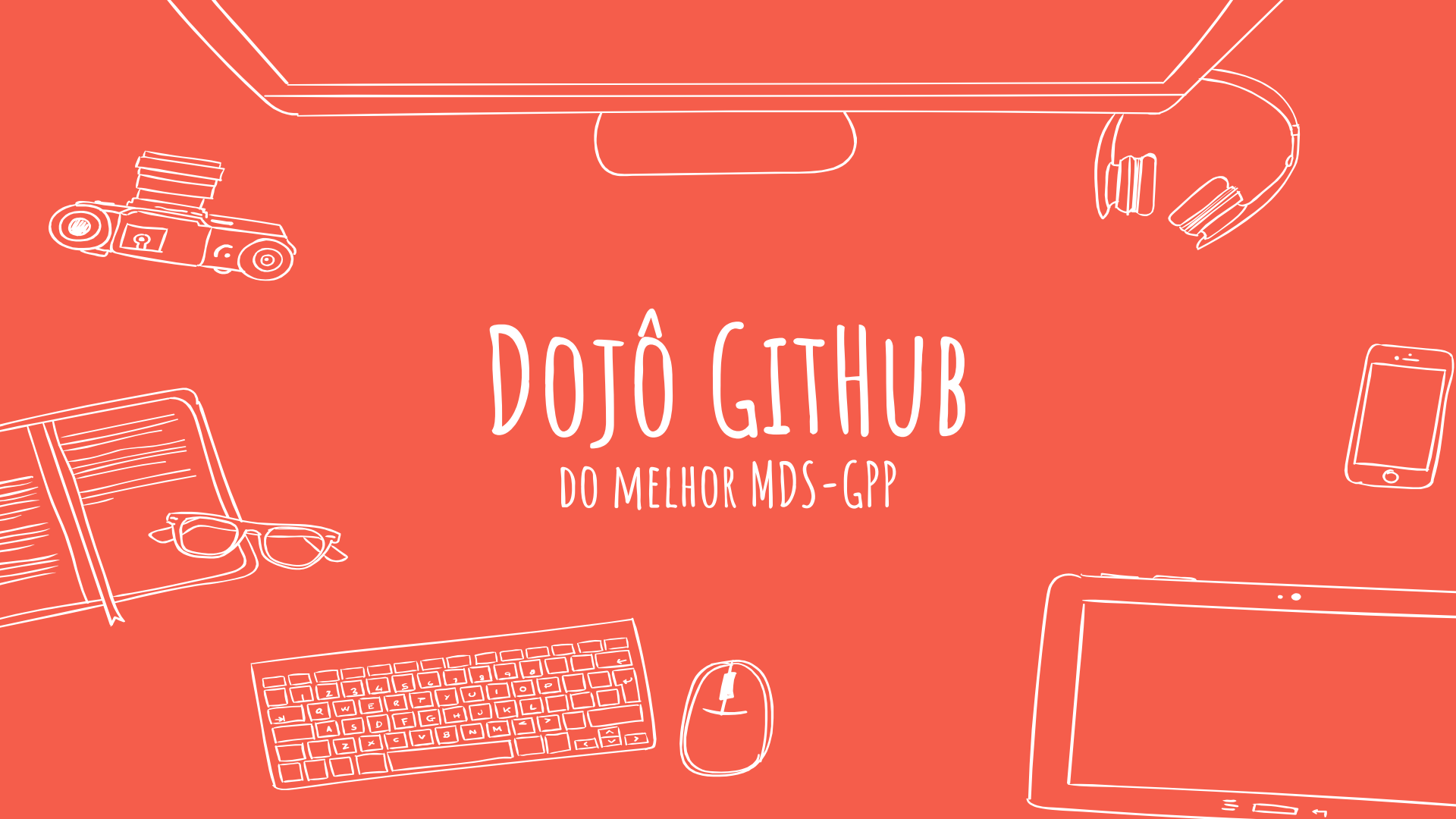
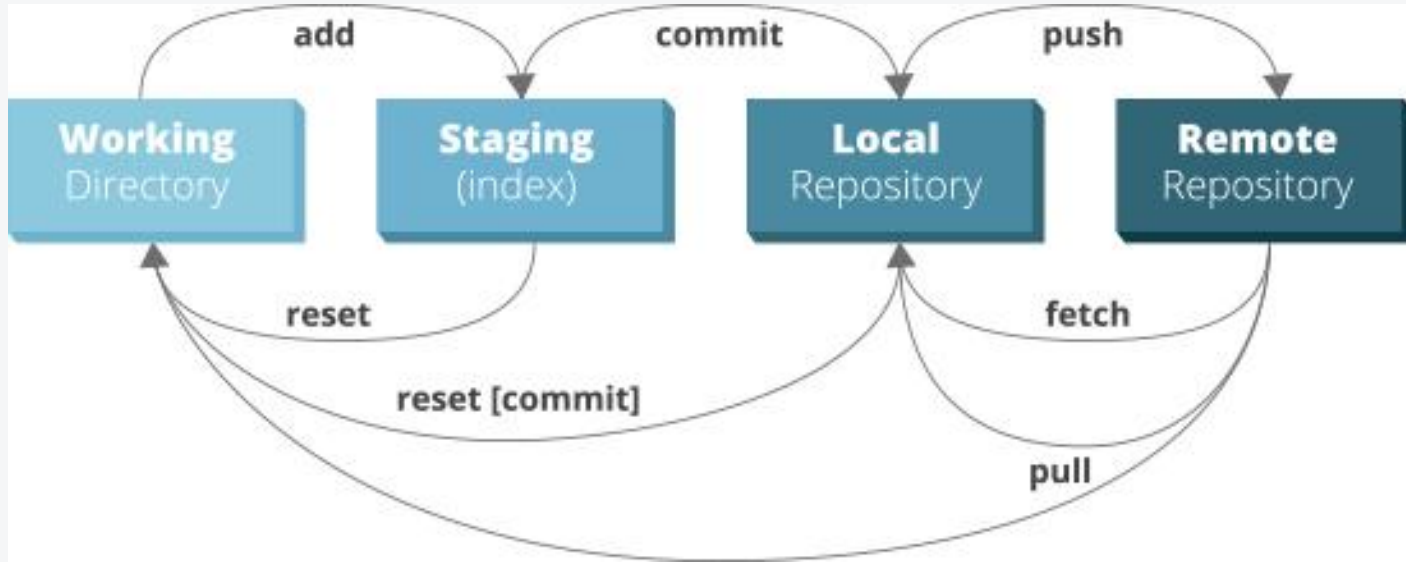


DOJÔ GITHUB

DO MELHOR MDS-GPP



RESUMO





DOJÔ GITHUB

Instalação

✖ `sudo apt-get install git`

Iniciar Repositório

✖ `git init`

Obter Repositório

✖ `git clone <url>`



DOJÔ GITHUB

Coloque seu usuário e e-mail do git

- ✖ `git config --global user.name "John Doe"`
- ✖ `git config --global user.email johndoe@example.com`



DOJÔ GITHUB

Após alterar documento, adicionar e commitar alterações.

✖ `git add <arquivo 1> <arquivo 2>
<arquivo n>`

✖ `git commit -m "nome do commit"`



*Apenas comite quando o que você estiver
feito e funcionando!*

Apenas comite na sua devel!



Coloque comentários úteis de commit

Commits devem ser feitos em inglês



EXEMPLOS DO QUE FAZER

O errado

- ✖ `git commit -m "Add the thing with the stuff"`
- ✖ `git commit -m "É nois"`

O correto

- ✖ `git commit -m "Created main loop & timing control"`
- ✖ `git commit -m "Update getting started documentation"`



DOJÔ GITHUB

Para saber como está o seu projeto em relação a mudanças

✖ git status

Para saber o log de commits

✖ git log



DOJÔ GITHUB

Enviar Alterações

✖ `git push origin devel`

Pegar alterações do remoto

✖ `git pull origin devel`

Adicionar Remoto

✖ `git remote add origin <url>`



DOJÔ GITHUB

Atualiza com o remoto

✖ git fetch

Atenção: isso não atualiza o repositório que você vê, precisa fazer um pull.

Sempre antes de começar a trabalhar atualize seu repositório local e dê pull se houver alguma alteração de alguém naquela branch



DOJÔ GITHUB

Se você não clonou, por exemplo fez download do arquivo, e quer seu repositório local sincronizado com o remoto.

✖ Digite: `git remote add origin <url>`

Assim você é capaz de enviar suas alterações.



DOJÔ GITHUB

Mudando de branch

✖ `git checkout devel_maria`

Out: Switched to branch "devel_maria"



DOJÔ GITHUB

Mudando de Branch

✖ `git checkout -b devel_joao`

Out: Switched to a new branch "devel_joao"

Isso é um atalho para:

✖ `git branch devel_joao`

✖ `git checkout devel_joao`



DOJÔ GITHUB

Mudando para uma branch que já existe
e que está atualizada no seu computador

✖ `git checkout -t devel_joao`

Out: Switched to a new branch "devel_joao"

QUADRO DE COMANDOS

git cheat sheet

learn more about git the simple way at rogerdudler.github.com/git-guide/
cheat sheet created by Nina Jaeschke of ninagrafix.com

create & clone

create new repository `git init`
clone local repository `git clone /path/to/repository`
clone remote repository `git clone username@host:/path/to/repository`

add & remove

add changes to INDEX `git add <filename>`
add all changes to INDEX `git add *`
remove/delete `git rm <filename>`

commit & synchronize

commit changes `git commit -m "Commit message"`
push changes to remote repository `git push origin master`
connect local repository to remote repository `git remote add origin <server>`
update local repository with remote changes `git pull`

branches

create new branch `git checkout -b <branch>`
e.g. `git checkout -b feature_x`
switch to master branch `git checkout master`
delete branch `git branch -d <branch>`
push branch to remote repository `git push origin <branch>`

merge

merge changes from another branch `git merge <branch>`
view changes between two branches `git diff <source_branch> <target_branch>`
e.g. `git diff feature_x feature_y`

tagging

create tag `git tag <tag> <commit ID>`
e.g. `git tag 1.0.0 1b2e1d63ff`
get commit IDs `git log`

restore

replace working copy with latest from HEAD `git checkout -- <filename>`



TAREFA 1 ONLINE □

Sempre confira com git log e git status para ajudar a saber como está o projeto.

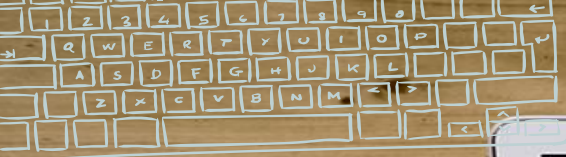
1. Clone o repositório que está na url <https://github.com/eduqg/dojoGit>
2. Entre na pasta dojoGit
3. Abra o arquivo README.md, altere e salve
4. Adicione as alterações
5. Commite para seu repositório local
6. Pushe para a branch master suas alterações
7. Resolva conflitos se tiver



TAREFA 2 OFFLINE □

Sempre confira com git log e git status para ajudar a saber como está o projeto.

1. Crie uma pasta chamada dojoOffline
2. Inicie um repositório dentro da pasta
3. Veja se está na master
4. Crie um arquivo chamado Hello.txt, adicione e commit na própria master
5. Crie uma nova branch chamada devel_offline e mude para ela
6. Crie um arquivo chamado Helloworld.txt
7. Adicione as alterações e commit as alterações
8. Mude para a branch master
9. Pegue as mudanças feitas na devel_offline e passe para a master





Now you can use any emoji as an icon!
And of course it resizes without losing quality and you
can change the color.

How? Follow Google instructions
<https://twitter.com/googledocs/status/730087240156643328>



and many more...