

O que é controle de versionamento?

sábado, 24 de junho de 2023 20:08

- Método usado para gerenciar o código fonte
- Desenvolvedores podem trabalhar em diversos tipos de versões diferentes
- É possível versões de um software em diferentes estágios

O que é um repositório?

terça-feira, 20 de junho de 2023

14:02

- Onde ficará armazenado seu código
- Cada projeto tem um repositório (Blog, Api, Ecommerce)
- Criação de uma repo é o início de um projeto
- Servidores de repo - Github e Bitbucket
- Cada repo pode ser baixado por um desenvolvedor para fazer várias versões diferentes - Clonagem do repositório

Criando repositórios

terça-feira, 20 de junho de 2023 14:11

- `git init` - Criação de uma repo através da pasta que você iniciou
- A partir do `git init` são criados arquivos necessários para aceitar códigos git, ficam armazenados em uma pasta oculta `.git`
- Quando criamos uma pasta no nosso computador e utilizamos o comando `git init` estamos criando um repositório nessa pasta que poderá ser enviado para um servidor posteriormente.

O que é um github?

terça-feira, 20 de junho de 2023

14:20

- É um serviço para gerenciar repositórios, quando precisamos formatar nosso computador e não queremos perder nossos projetos, podemos disponibilizar no github e depois recuperá-los
- Podemos disponibilizar no github os nossos projetos para outros dev's

Observação sobre a Branch principal

Há um tempo atrás houve uma mudança no GitHub, o branch principal se chamava **master** e agora se chama **main**

Enviando repositórios para o GH

terça-feira, 20 de junho de 2023

14:30

- Precisamos criar um repo no servidor github para que seja sincronizado com a pasta local e enviado
 - São executados com poucos códigos e é feito apenas 1 vez projeto
- Já usamos o comando git init agora podemos adicionar os arquivos e enviar um commit (mensagens) para o repositório

```
git init
git add README.md
git commit -m "first commit"
git branch -M master
git remote add origin git@github.com:matheusbattisti/curso_git_1.git
git push -u origin master
```

Verificando alterações - git status

terça-feira, 20 de junho de 2023

15:03

- Deve ser utilizado o código **git status**
- verifica as modificações feitas nos arquivos
- é a diferença do que já foi enviado ao servidor ou salvo no projeto

```
$ git status
On branch main
Your branch is up to date with 'origin/main'.

Changes not staged for commit:
  (use "git add <file>..." to update what will be committed)
  (use "git restore <file>..." to discard changes in working directory)
        modified:   01.Conteudo do curso.docx

no changes added to commit (use "git add" and/or "git commit -a")

Geronimo Morais@DESKTOP-HQ9L9NO MINGW64 /d/Git_Github (main)
$
```

quando usamos o **git status** mostra que o arquivo "conteudo do curso" foi modificado, existe também o untracked que é quando o arquivo foi adicionado a pasta local e ainda não foi adicionado ao repo (github)

```
$ git status
On branch main
Your branch is up to date with 'origin/main'.

Changes not staged for commit:
  (use "git add <file>..." to update what will be committed)
  (use "git restore <file>..." to discard changes in working directory)
        modified:   01.Conteudo do curso.docx

Untracked files:
  (use "git add <file>..." to include in what will be committed)
        02.Trabalhando com branches.docx

no changes added to commit (use "git add" and/or "git commit -a")
```

Adicionando arquivos ao projeto - git add

terça-feira, 20 de junho de 2023

15:32

- Deve-se utilizar o comando git add
- Pode ser adicionado 1 arquivo ou vários de uma vez
- Somente adicionando os arquivos eles estarão sendo monitorados pelo git, se não adicionados não estará no controle de versão

Salvando alterações - git commit

terça-feira, 20 de junho de 2023 15:36

- As alterações do projeto será salvo
- Deve-se usar o comando **git commit**
- flag **-a** - envia para todos os arquivos
- Mensagem pode ser adicionada com a flag **-m**

Salvando arquivo especifico

```
git commit a.txt -m "Estou enviando o arquivo de texto"
```

Salvando todos os arquivos com commit

```
PS H:\15_GIT_BASICO_AO_AVANÇADO\2_comandos_fundamentais\arquivos\1_primeiro_repo> git commit -a -m "Enviando a funcilidade X"
[master 7cd4ee1] Enviando a funcilidade X
5 files changed, 15 insertions(+), 1 deletion(-)
create mode 100644 b.txt
create mode 100644 c.txt
create mode 100644 css/styles.css
create mode 100644 index.html
```


Enviando arquivos para o repo remoto - git push

terça-feira, 20 de junho de 2023

15:41

- Quando finalizamos o commit e todos os outros comandos, precisamos enviar ao código fonte
- Deve-se usar o **git push**
- Após isto, o servidor será atualizado baseado no repo local (Pasta)

Your branch is ahead of origin/master - sua branch está a frente da branch master

```
PS H:\15_GIT_BASICO_AO_AVANÇADO\2_comandos_fundamentais\arquivos\1_primeiro_repo> git status
On branch master
Your branch is ahead of 'origin/master' by 3 commits.
  (use "git push" to publish your local commits)

nothing to commit, working tree clean
PS H:\15_GIT_BASICO_AO_AVANÇADO\2_comandos_fundamentais\arquivos\1_primeiro_repo> git push
Enumerating objects: 14, done.
Counting objects: 100% (14/14), done.
Delta compression using up to 6 threads
Compressing objects: 100% (7/7), done.
Writing objects: 100% (12/12), 1.09 KiB | 1.09 MiB/s, done.
Total 12 (delta 1), reused 0 (delta 0), pack-reused 0
remote: Resolving deltas: 100% (1/1), done.
To https://github.com/matheusbattisti/curso_git_1.git
 44ea6f0..9d1b50c  master -> master
```

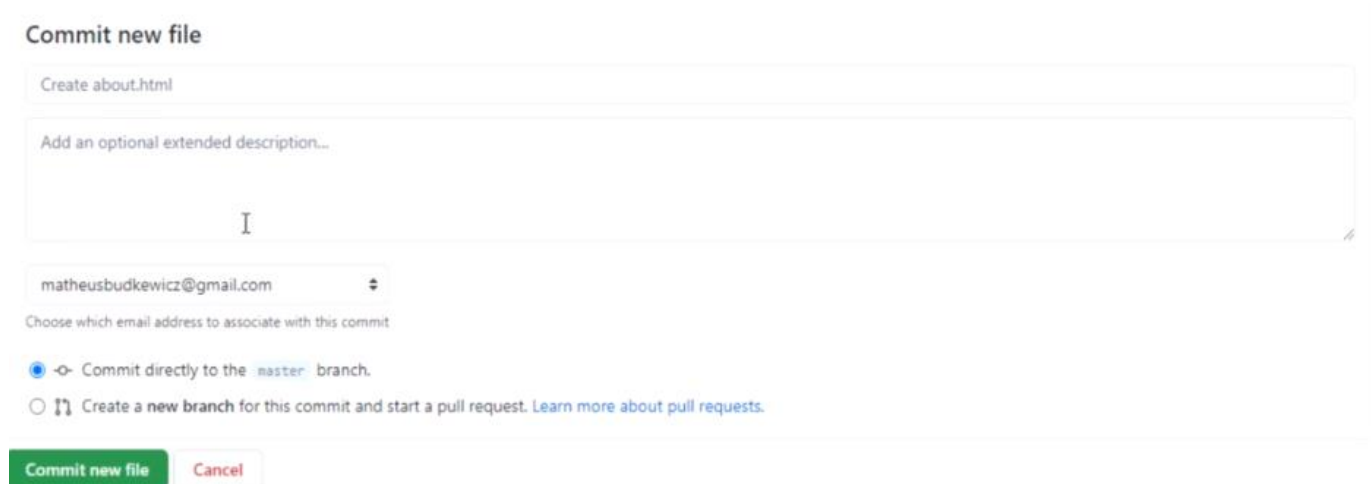
- Quando finalizamos o git push o nosso repositório local está parelho com o repositório remoto

Recebendo alterações - git pull

terça-feira, 20 de junho de 2023 16:34

- Funciona como busca de atualizações dos códigos
- Deve-se usar o comando **git pull**
- Após o comando serem feitas buscam por atualizações, caso seja encontrada serão unidas ao seu repo local.

Criação de um arquivo no repositório online



The screenshot shows the GitHub 'Commit new file' interface. At the top, it says 'Commit new file'. Below that is a text input field with the placeholder 'Create about.html'. Underneath is a larger text area with the placeholder 'Add an optional extended description...'. Below the description area is a dropdown menu showing the email address 'matheusbudkewicz@gmail.com'. Below the email dropdown is a small text label 'Choose which email address to associate with this commit'. There are two radio button options: the first is selected and says 'Commit directly to the master branch.', and the second is unselected and says 'Create a new branch for this commit and start a pull request. Learn more about pull requests.' At the bottom, there are two buttons: a green 'Commit new file' button and a red 'Cancel' button.

Recebendo alterações para o repositório local (Pasta)

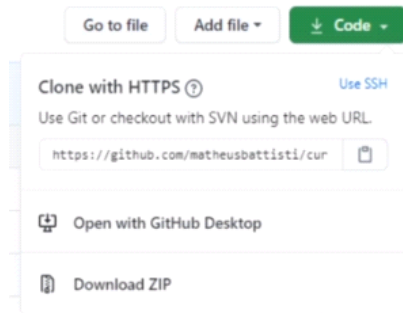
```
PS H:\15_GIT_BASICO_AO_AVANÇADO\2_comandos_fundamentais\arquivos\1_primeiro_repo> git pull
remote: Enumerating objects: 4, done.
remote: Counting objects: 100% (4/4), done.
remote: Compressing objects: 100% (2/2), done.
remote: Total 3 (delta 1), reused 0 (delta 0), pack-reused 0
Unpacking objects: 100% (3/3), 680 bytes | 7.00 KiB/s, done.
From https://github.com/matheusbattisti/curso_git_1
  9d1b50c..8c35092 master    -> origin/master
Updating 9d1b50c..8c35092
Fast-forward
 about.html | 1 +
 1 file changed, 1 insertion(+)
 create mode 100644 about.html
```

Clonando repositório online - git clone

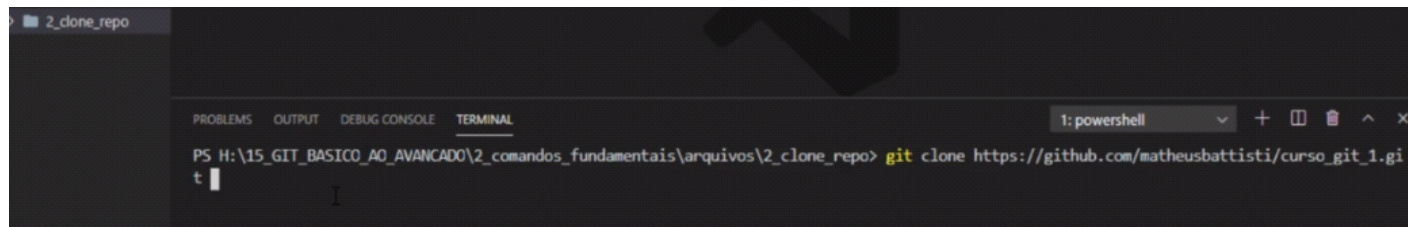
terça-feira, 20 de junho de 2023 17:10

- Baixar o projeto do servidor
- Deve-se usar o **git clone**
- Pegando referências do repositório remoto
- Geralmente utilizado quando entramos em um projeto

Copiando URL do projeto



Clonando repositório online



Removendo arquivos - git rm

terça-feira, 20 de junho de 2023 18:10

- Deve-se usar o comando git rm
- Arquivos deletados da monitoração do git
- Não terá mais atualizações pelo git após a exclusão, apenas quando for adicionado pelo comando git add

Excluindo um arquivo

```
PS H:\15_GIT_BASICO_AO_AVANÇADO\2_comandos_fundamentais\arquivos\2_clone_repo> git rm c.txt
rm 'c.txt'
PS H:\15_GIT_BASICO_AO_AVANÇADO\2_comandos_fundamentais\arquivos\2_clone_repo> git status
On branch master
Your branch is up to date with 'origin/master'.

Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
        deleted:    c.txt
```

Enviando commit com deleção

```
Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
        deleted:    c.txt

PS H:\15_GIT_BASICO_AO_AVANÇADO\2_comandos_fundamentais\arquivos\2_clone_repo> git commit -a -m "Deletando arquivo desnecessário"
[master d6d39e1] Deletando arquivo desnecessário
1 file changed, 0 insertions(+), 0 deletions(-)
delete mode 100644 c.txt
PS H:\15_GIT_BASICO_AO_AVANÇADO\2_comandos_fundamentais\arquivos\2_clone_repo> git push
```

Verificando alterações - git log

quarta-feira, 21 de junho de 2023

18:57

- Deve-se usar o comando **git log**
- Mostra todos os commits feitos no projeto até agora

Mostrando os commits feitos - ordem decrescente

```
$ git log
commit 1ae5c3e2fe28500d140155239fd79cea42ac427a (HEAD -> main, origin/main)
Author: Geronimonetto <geronimomorais@hotmail.com>
Date: Tue Jun 20 16:47:29 2023 -0300

    Deletando arquivo criado para git pull

commit dcbceec3de2199ccc4901dabfb340d681b2ffea0
Author: Geronimo Morais <108224476+Geronimonetto@users.noreply.github.com>
Date: Tue Jun 20 16:46:49 2023 -0300

    Create a.txt

commit f73ee2a1dfbc4e75fff2e5bcbaabbd0da3d5991c
Author: Geronimonetto <geronimomorais@hotmail.com>
Date: Tue Jun 20 15:14:27 2023 -0300

    Criando conteúdo: Comandos
```

Movendo arquivos ou renomeando - git mv

quarta-feira, 21 de junho de 2023

19:17

- Deve-se usar o comando **git mv**
- É usado para mover arquivos para uma outra pasta
- O arquivo anterior é excluído e é criado um novo

Movendo arquivo para uma pasta

```
Geronimo Morais@DESKTOP-HQ9L9NO MINGW64 /d/Git_Github (main)
$ git mv a.txt gtmv/a.txt

Geronimo Morais@DESKTOP-HQ9L9NO MINGW64 /d/Git_Github (main)
$ git status
On branch main
Your branch is up to date with 'origin/main'.

Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
    renamed:   a.txt -> gtmv/a.txt
```

OBS: Caso o arquivo seja novo, devemos primeiro add para o monitoramento do git para depois renomear ou alterar sua pasta.

usamos o comando **git mv** nomedoarquivo.formato
pastadestino/nomedoarquivo.formato

Desfazendo alterações - git checkout

quarta-feira, 21 de junho de 2023

19:31

- Deve-se usar o comando **git checkout**
- O arquivo volta ao seu estado original do github (**Rollback**)
- Após o uso do git checkout o arquivo sai do staging
- Caso feita uma alteração novamente entra em um staging novamente

Cancelando modificação do arquivo para o estado original do repositório do github

```
Geronimo Morais@DESKTOP-HQ9L9NO MINGW64 /d/Git_Github (main)
$ git status
On branch main
Your branch is up to date with 'origin/main'.

Changes not staged for commit:
  (use "git add <file>..." to update what will be committed)
  (use "git restore <file>..." to discard changes in working directory)
        modified:   02.Trabalhando com branches.docx

no changes added to commit (use "git add" and/or "git commit -a")

Geronimo Morais@DESKTOP-HQ9L9NO MINGW64 /d/Git_Github (main)
$ git checkout 02.Trabalhando\ com\ branches.docx
Updated 1 path from the index

Geronimo Morais@DESKTOP-HQ9L9NO MINGW64 /d/Git_Github (main)
$ git status
On branch main
Your branch is up to date with 'origin/main'.

nothing to commit, working tree clean
```

git checkout nomedoarquivo.formato

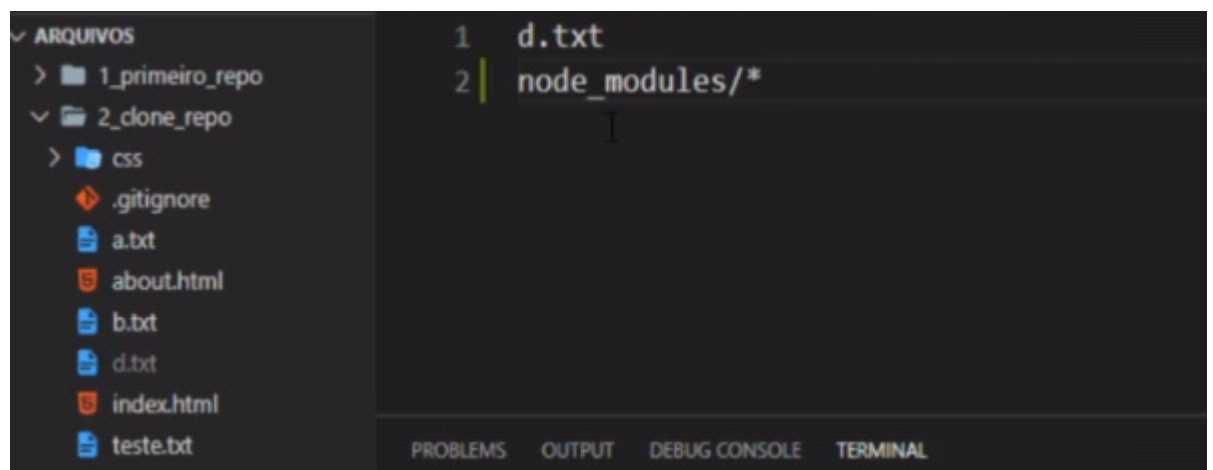
Ignorando arquivos - .gitignore

quinta-feira, 22 de junho de 2023

16:53

- É uma forma muito utilizada por desenvolvedores para ignorar extensão ou arquivos específicos desnecessários ou que outras pessoas não precisam saber
- Deve-se criar um arquivo na raiz do projeto com o nome **.gitignore**
- Existem arquivos gerados automaticamente que são ignorados dessa forma
- Podemos inserir arquivos que não devem entrar no versionamento (Github)
- Primeiro deve-se adicionar no gitignore antes de adicionar ou fazer a criação automática do arquivo, ou seja caso utilize algum projeto que precise ser ignorado, coloque o nome dos arquivos ou pasta antes de iniciar o projeto.

Ignorando arquivo específico ou pasta e tudo que estiver dentro dela



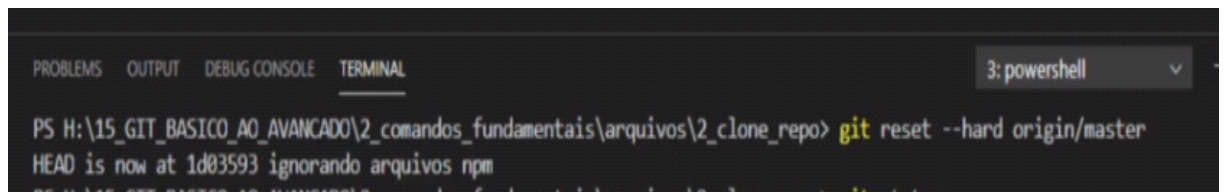
Resetando uma branch - git reset

quinta-feira, 22 de junho de 2023

17:11

- Reseta as mudanças feitas
- Utilizado com a flag --hard
- Todas os commit's feitos no ultimo push e também pendentes são excluídos

Resetando a última alteração feita na branch master



```
PS H:\15_GIT_BASICO_AO_AVANÇADO\2_comandos_fundamentais\arquivos\2_clone_repo> git reset --hard origin/master
HEAD is now at 1d03593 ignorando arquivos npm
```

O git reset deve ser usando antes do push

Ex: Fizemos várias alterações e estão comitadas ou precisando adicionar, porém nosso repositório origin (original - repositório remoto) não foi alterado podemos usar o `git reset --hard origin/main` para excluir todos os commit's e alterações feitas nos arquivos para a forma original no repositório remoto.

Comandos

terça-feira, 20 de junho de 2023

14:19

- **git init** - Iniciando uma repo
- **git add nomearquivo** - trackeia um arquivo para ficar pronto pro commit
- **git commit -m "Mensagem do commit"** - Sinaliza uma mensagem para enviar para o repo como anotação
- **git branch -M nomedabranh** - Criação de uma branch no repositório local
- **git status** - verificar mudanças feitas nos arquivos
- **git add nomearquivo** - Adicionando arquivos para serem commitados
- **git add .** - Adicionando todos os arquivos de uma vez
- **git rm** - removendo arquivos
- **git log** - Verifica todos os commit's feitos até o momento detalhadamente
- **git log --oneline** - Mostra todos os commits em uma linha só
- **git revert HEAD** - Reverte o último commit feito

Casos específicos

terça-feira, 20 de junho de 2023 18:20

Caso o arquivo contenha espaços no nome como excluir?

```
Geronimo Morais@DESKTOP-HQ9L9NO MINGW64 /d/Git_Github (main)
$ git rm "02.Trabalhando com branches"
fatal: pathspec '02.Trabalhando com branches' did not match any files

Geronimo Morais@DESKTOP-HQ9L9NO MINGW64 /d/Git_Github (main)
$ git rm 02.Trabalhando\ com\ branches.docx
rm '02.Trabalhando com branches.docx'
```

deve-se usar \ + espaço para representar o espaço

Movendo um arquivo

```
Geronimo Morais@DESKTOP-HQ9L9NO MINGW64 /d/Git_Github (main)
$ git mv a.txt gtmv/a.txt

Geronimo Morais@DESKTOP-HQ9L9NO MINGW64 /d/Git_Github (main)
$ git status
On branch main
Your branch is up to date with 'origin/main'.

Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
    renamed:   a.txt -> gtmv/a.txt
```

OBS: Caso o arquivo seja novo, devemos primeiro add para o monitoramento do git para depois renomear ou alterar sua pasta.

Modificando arquivos

```
Geronimo Morais@DESKTOP-HQ9L9NO MINGW64 /d/Git_Github (main)
$ git commit -m "Enviando conteudo"
On branch main
Your branch is ahead of 'origin/main' by 1 commit.
  (use "git push" to publish your local commits)

Changes not staged for commit:
  (use "git add <file>..." to update what will be committed)
  (use "git restore <file>..." to discard changes in working directory)
    modified:   02.Trabalhando com branches.docx

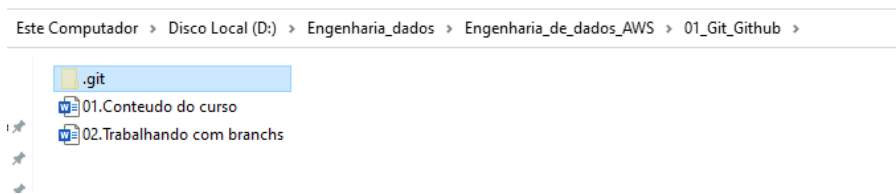
no changes added to commit (use "git add" and/or "git commit -a")

Geronimo Morais@DESKTOP-HQ9L9NO MINGW64 /d/Git_Github (main)
$ git commit -a -m "Criando conteudo: Fazendo modificações"
[main 25149d9] Criando conteudo: Fazendo modificações
 1 file changed, 0 insertions(+), 0 deletions(-)

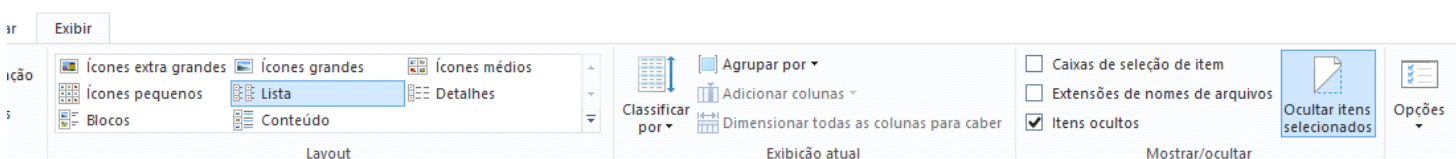
Geronimo Morais@DESKTOP-HQ9L9NO MINGW64 /d/Git_Github (main)
```

Podemos observar no exemplo a seguir, quando modificamos um arquivo já existente, não precisamos mais usar o git add, podemos apenas usar o git commit -a -m "Mensagem" para poder enviar o conteúdo para o repositório.

Pasta inacessível dentro do git



Essa pasta está dentro de uma pasta principal que contém o arquivo .git de início de um repositório porém ela também tem um .git dentro, a pasta no github ficará inacessível, logo devemos apagar essa pasta .git da pasta que encontra-se inacessível.



Marque a opção itens ocultos > Excluir pasta .git