

# **Curso: Ciência da Computação**

## **Git e GitHub**

Professor: Clayton Zambon

# 1. Introdução

## Git

### Git

#### - Controle de Versão:

- Cópias e mais cópias do mesmo projeto;
- Apaguei um arquivo importante e não tem como recuperar;

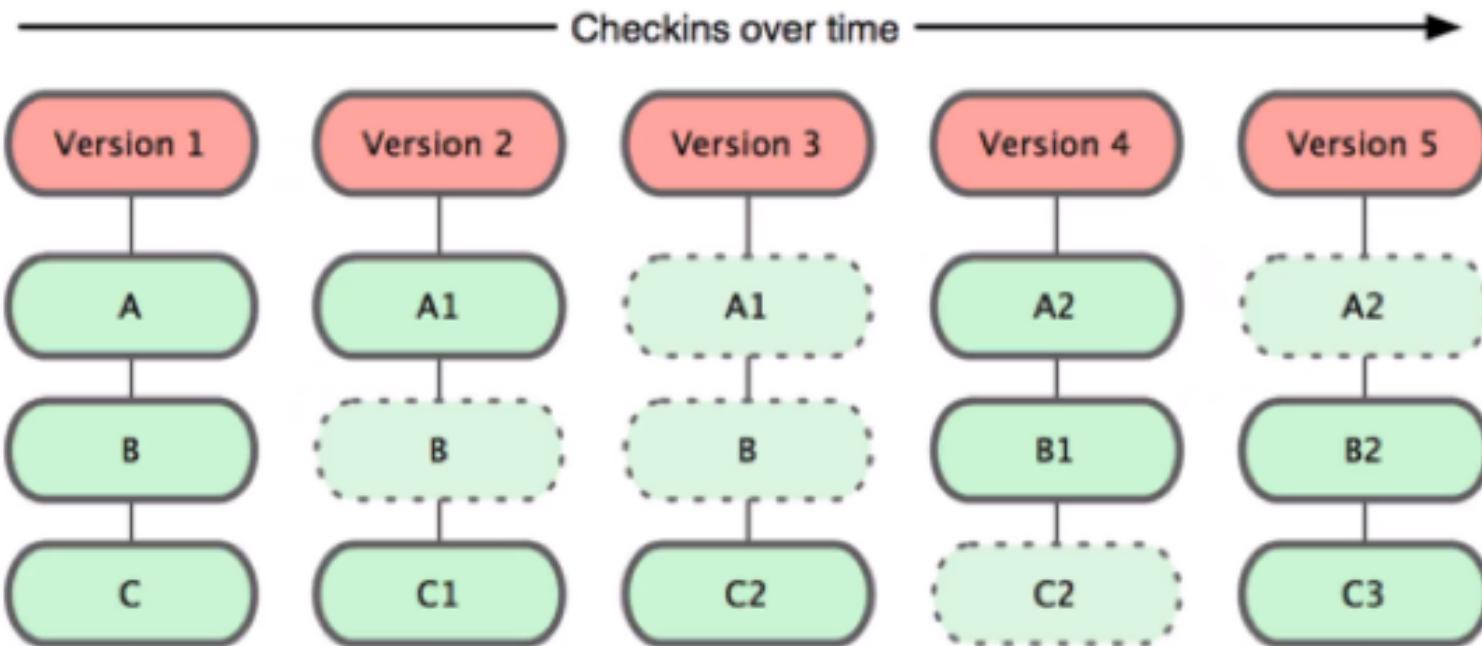
#### - O que é Controle de Versão?

- Sistema com a finalidade de gerenciar diferentes versões de um documento;
- Podemos avançar versões e voltar para versões antigas;

# 1. Introdução

## Git

## Sistema Git



# 1. Introdução

## Git

### Sistema de Controle de Versão

- 1º Responsável por versionar os arquivos do seu projeto;
  - Isto evita que você fique fazendo os versionamentos manualmente;
- 2º Outros sistema de versão trabalham com as diferenças dos arquivos;
- 3º O Git trabalha com os estados dos arquivos;

# 1. Introdução

## Git

### Beve história do Git

- BitKeeper Scalable Version Control;

- Guardava todo o código Kernel do Linux;

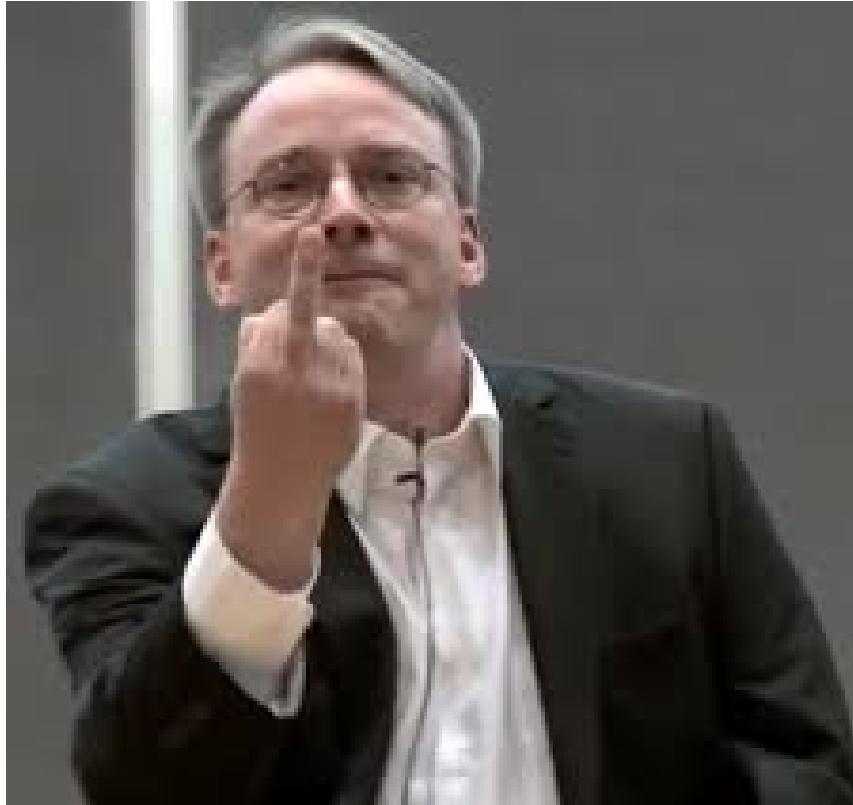


# 1. Introdução

## Git

### Beve história do Git

Linus Torvalds – Criador do Linux e do Git



# 1. Introdução

## Git

### Beve história do Git

#### MELHORIAS

- Velocidade;
- Design simples;
- Suporte robusto a desenvolvimento não linear (milhares de branches paralelos);
- Totalmente distribuído;
- Capaz de lidar eficientemente com grandes projetos como o Kernel do Linux;

# 1. Introdução

## Git

### O que é o GitHub

- Serviço Web compartilhado para projetos que utilizam o Git para versionamento;
  - É um local na nuvem que irá armazenar os projetos versionados pelo Git;



# 1. Introdução

## Git

### Instalando o Git - <https://git-scm.com/downloads>

The screenshot shows the 'Downloads' section of the official Git website. At the top, there are links for Mac OS X, Windows, and Linux/Unix. Below these, a message states that older releases are available and the Git source repository is on GitHub. A large monitor icon displays the latest source release version 2.25.0 with a 'Download 2.25.0 for Windows' button. On the left sidebar, there are links for About, Documentation, Downloads (GUI Clients and Logos), and Community. A sidebar box also mentions the availability of the Pro Git book online for free.

# 1. Introdução

## Git

### Configuração Inicial do Git

- O Git guarda suas informações em três lugares:

- Git Config do sistema como um todo;
- Git Config do usuário;
- Git Config do projeto;

- Config do usuário:

- git config --global user.name “Clayton Zambon”
- git config --global user.email “zambon@outlook.com.br”
- git config user.name
- git config user.email
- git config --list

# 1. Introdução

## Git

### Configuração Inicial do Git

```
C:\WINDOWS\system32\cmd.exe
Microsoft Windows [versão 10.0.18362.592]
(c) 2019 Microsoft Corporation. Todos os direitos reservados.

C:\Users\corel>git config --global user.name "Clayton Zambon"
C:\Users\corel>git config --global user.email "zambon@outlook.com.br"
C:\Users\corel>git config user.nam
C:\Users\corel>git config user.name
Clayton Zambon
C:\Users\corel>git config user.email
zambon@outlook.com.br
C:\Users\corel>
```

# 1. Introdução

## Git

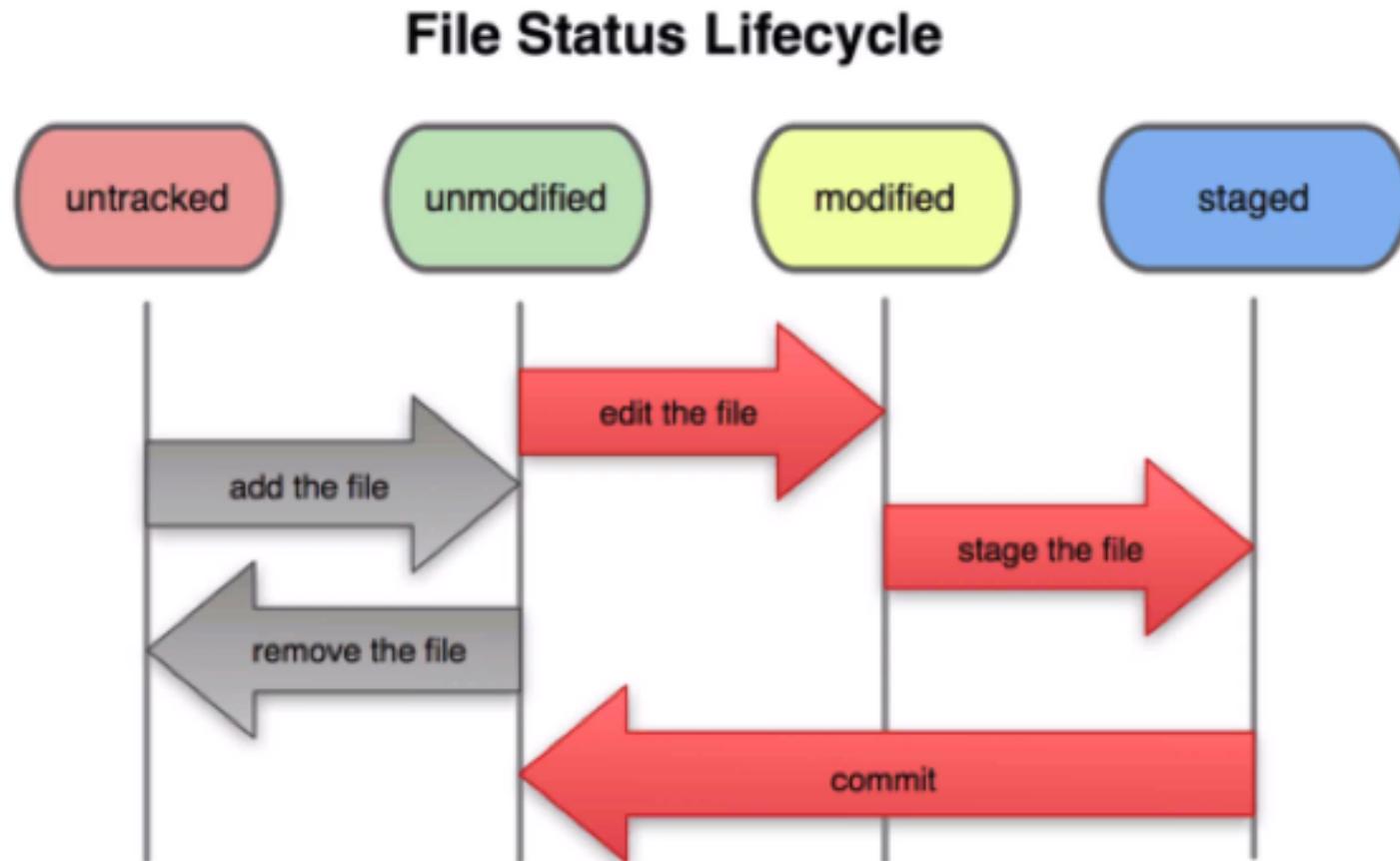
### Inicializando um Repositório Git

- Primeiro crie uma Pasta para alocar seus projetos;
  - Exemplo: git-projetos
- **git init**
  - Acesse a pasta dos projetos e inicialize o repositório com o comando:
  - Exemplo: C:\git-projetos\git init
- O comando acima criará algumas pastas com informações que serão utilizadas pelo Git;

# 1. Introdução

## Git

### Círculo de vida dos Status de seus arquivos



# 1. Introdução

## Git

### Clico de vida dos Status de seus arquivos

#### - git status

- Verifica os status dos seus arquivos:

C:\git-projetos\git status

```
C:\WINDOWS\system32\cmd.exe
C:\git-projetos>git status
On branch master

No commits yet

nothing to commit (create/copy files and use "git add" to track)

C:\git-projetos>
```

# 1. Introdução

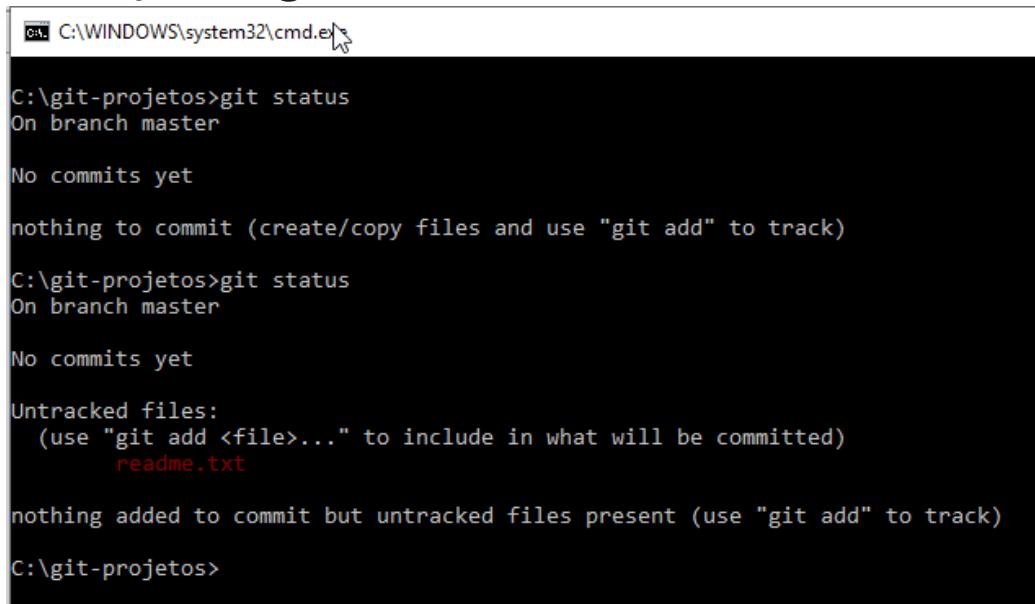
## Git

### Clico de vida dos Status de seus arquivos

#### - git status

- Execução do comando após criar um arquivo novo;

C:\git-projetos\git status



```
C:\WINDOWS\system32\cmd.exe
C:\git-projetos>git status
On branch master

No commits yet

nothing to commit (create/copy files and use "git add" to track)

C:\git-projetos>git status
On branch master

No commits yet

Untracked files:
  (use "git add <file>..." to include in what will be committed)
    README.txt

nothing added to commit but untracked files present (use "git add" to track)

C:\git-projetos>
```

# 1. Introdução

## Git

### Adicionando arquivos no Git

#### - git add <nome do arquivo>

- Comando para adicionar arquivos no Git

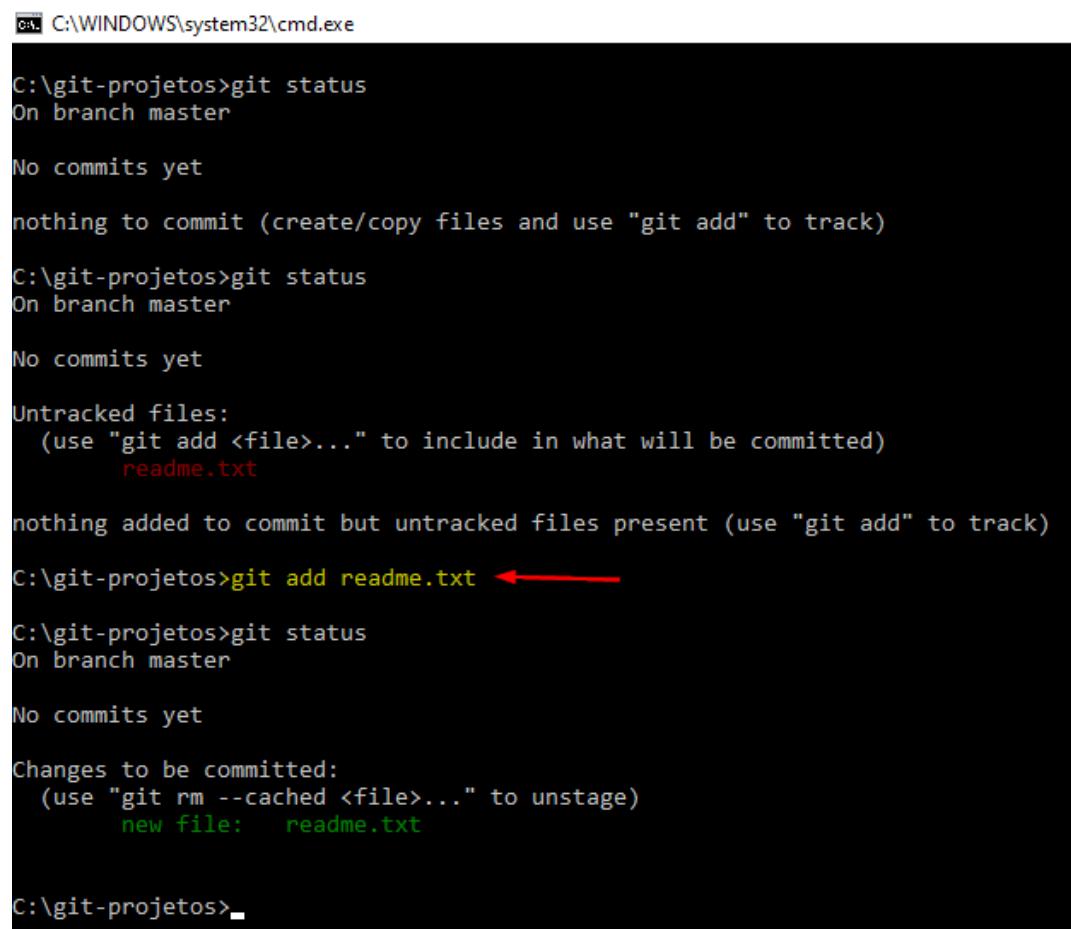
```
C:\git-projetos\git add readme.txt
```

# 1. Introdução

## Git

### Adicionando arquivos no Git

- git add <nome do arquivo>



```
C:\WINDOWS\system32\cmd.exe
C:\git-projetos>git status
On branch master

No commits yet

nothing to commit (create/copy files and use "git add" to track)

C:\git-projetos>git status
On branch master

No commits yet

Untracked files:
  (use "git add <file>..." to include in what will be committed)
    README.txt

nothing added to commit but untracked files present (use "git add" to track)

C:\git-projetos>git add README.txt ←
C:\git-projetos>git status
On branch master

No commits yet

Changes to be committed:
  (use "git rm --cached <file>..." to unstage)
    new file:   README.txt

C:\git-projetos>
```

# 1. Introdução

## Git

Veja o que acontece se eu fizer uma alteração no arquivo adicionado:

```
C:\WINDOWS\system32\cmd.exe
No commits yet
nothing to commit (create/copy files and use "git add" to track)

C:\git-projetos>git status
On branch master

No commits yet

Untracked files:
  (use "git add <file>..." to include in what will be committed)
    README.txt

nothing added to commit but untracked files present (use "git add" to track)

C:\git-projetos>git add README.txt

C:\git-projetos>git status
On branch master

No commits yet

Changes to be committed:
  (use "git rm --cached <file>..." to unstage)
    new file:   README.txt

C:\git-projetos>git status
On branch master

No commits yet

Changes to be committed:
  (use "git rm --cached <file>..." to unstage)
    new file:   README.txt ←

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:   README.txt ←

C:\git-projetos>
```

# 1. Introdução

## Git

- Veja que o Git identificou que houve mudanças no arquivo;
- Se eu quiser levar essas modificações devo passar ele para “staged”;
- Então digito “git add readme.txt”
- Depois “git status”

# 1. Introdução

## Git

```
C:\WINDOWS\system32\cmd.exe
nothing added to commit but untracked files present (use "git add" to track)

C:\git-projetos>git add readme.txt

C:\git-projetos>git status
On branch master

No commits yet

Changes to be committed:
  (use "git rm --cached <file>..." to unstage)
    new file:   readme.txt

C:\git-projetos>git status
On branch master

No commits yet

Changes to be committed:
  (use "git rm --cached <file>..." to unstage)
    new file:   readme.txt

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:   readme.txt

C:\git-projetos>git add readme.txt ←
C:\git-projetos>git status ←
On branch master

No commits yet

Changes to be committed:
  (use "git rm --cached <file>..." to unstage)
    new file:   readme.txt ←

C:\git-projetos>
```

# 1. Introdução

## Git

### Vamos criar o primeiro Commit

#### - **git commit -m “Adicionando arquivo Readme.txt”**

- Este é o momento em que você irá “avisar ao Git” para pegar todos os arquivos do repositório para criar uma imagem deles, um Snapshot ou versão;
- É uma boa prática sempre incluir um comentário no commit informando se você adicionou uma nova funcionalidade, se corrigiu um bug, ou seja, qualquer informação pertinente é importante colocar na mensagem do comentário. Desta forma através dos logs é possível saber o que foi feito;

#### - **git commit -am “Todos os arquivos modificados”**

- Adicionado ao “staged” todos os arquivos modificados

# 1. Introdução

## Git

```
C:\git-projetos>git status
On branch master

No commits yet

Changes to be committed:
  (use "git rm --cached <file>..." to unstage)
    new file:  readme.txt

C:\git-projetos>git status
On branch master

No commits yet

Changes to be committed:
  (use "git rm --cached <file>..." to unstage)
    new file:  readme.txt

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:   readme.txt

C:\git-projetos>git add readme.txt

C:\git-projetos>git status
On branch master

No commits yet

Changes to be committed:
  (use "git rm --cached <file>..." to unstage)
    new file:  readme.txt

C:\git-projetos>git commit -m "Adicionando arquivo readme.txt"
[master (root-commit) a2ade9b] Adicionando arquivo readme.txt
 1 file changed, 1 insertion(+)
 create mode 100644 readme.txt

C:\git-projetos>
```

# 1. Introdução

## Git

```
C:\git-projetos>git commit -m "Adicionando arquivo readme.txt"
[master (root-commit) a2ade9b] Adicionando arquivo readme.txt
 1 file changed, 1 insertion(+)
 create mode 100644 readme.txt
```

```
C:\git-projetos>
```

local atual

Criou um commit com este hash  
com este comentário

Este números e letras são uma identificação da minha versão.  
Cada commit irá gerar um número diferente e é com este  
número que conseguimos voltar em uma versão específica

# 1. Introdução

## Git

### Vamos criar o primeiro Commit

- Após o commit se digitamos “git status” veremos que o git informa que não há nada para ser feito;
- Isto acontece porque com o commit “salvamos” a informação;
- Se alterarmos ou adicionarmos arquivos no repositório o Git irá identificar;
- Se dermos um commit com algum arquivo alterado ou adicionado será exibido um erro;
- Para resolver devemos primeiro adicionar os arquivos e depois fazer o commit;

# 1. Introdução

## Git

### Visualizando Logs do Git

#### - git log

- Exibe uma “hash” do commit;
- Autor da alteração e e-mail;
- Data, hora e a mensagem do commit;

#### - git log --decorate

- Podem ser passadas opções ao git log;
- O Decorate exibe qual branch está, se foi feito algum merge, etc;

#### - git log --author “Cla”

#### - git shortlog ou git shortlog -sn

#### - git log --graph

- git show a2ade9b62d75a9921addcbc56cee1b05e937090a

- Obs.: use o git log sempre que precisar

# 1. Introdução

## Git

### Visualizando o Diff

#### - **git diff**

- Com este comando é possível visualizar as mudanças realizadas nos arquivos antes mesmo de realizar o commit;
- Este comando é muito importante. Com ele você consegue dar uma lida nas diferenças e dar uma revisada evitando enviar “sujeira” antes de fazer o commit;

#### - **git diff --name-only**

# 1. Introdução

## Git

### Desfazendo ações no Git

#### - **git checkout <NomeDoArquivo>**

- Este comando é utilizado após você editar um arquivo sem adicioná-lo ao Git;
- Executando ele, o Git retorna ao arquivo antes da mudança;

#### - **git reset HEAD <NomeDoArquivo>**

- Este comando é utilizado após você adicionar (git add) o arquivo no Staged do Git;
- Este comando tira o arquivo do “staged” do Git;
- Se executar o git checkout <NomeDoArquivo> será removida toda mudança;

# 1. Introdução

## Git

### Desfazendo ações no Git

#### - **git reset --soft --mixed --hard**

- Soft: pega as modificações e ignora o último commit, porém o arquivo continua com a alteração feita e adicionado ao Git.

```
git reset --soft a2ade9b62d75a9921addcbc56cee1b05e937090a
```

- Mixed: ignora o commit, e volta os arquivos para antes da adição ao Git;

```
git reset --mixed a2ade9b62d75a9921addcbc56cee1b05e937090a
```

- Hard: ignora o commit e tudo que foi feito também inclusive a adição ao Git. Muito cuidado ao usar esta opção;

```
git reset --hard a2ade9b62d75a9921addcbc56cee1b05e937090a
```

- Sempre deve-se escolher no mínimo o commit anterior ao último pois estamos “voltando no tempo”;

# 1. Introdução

## GitHub

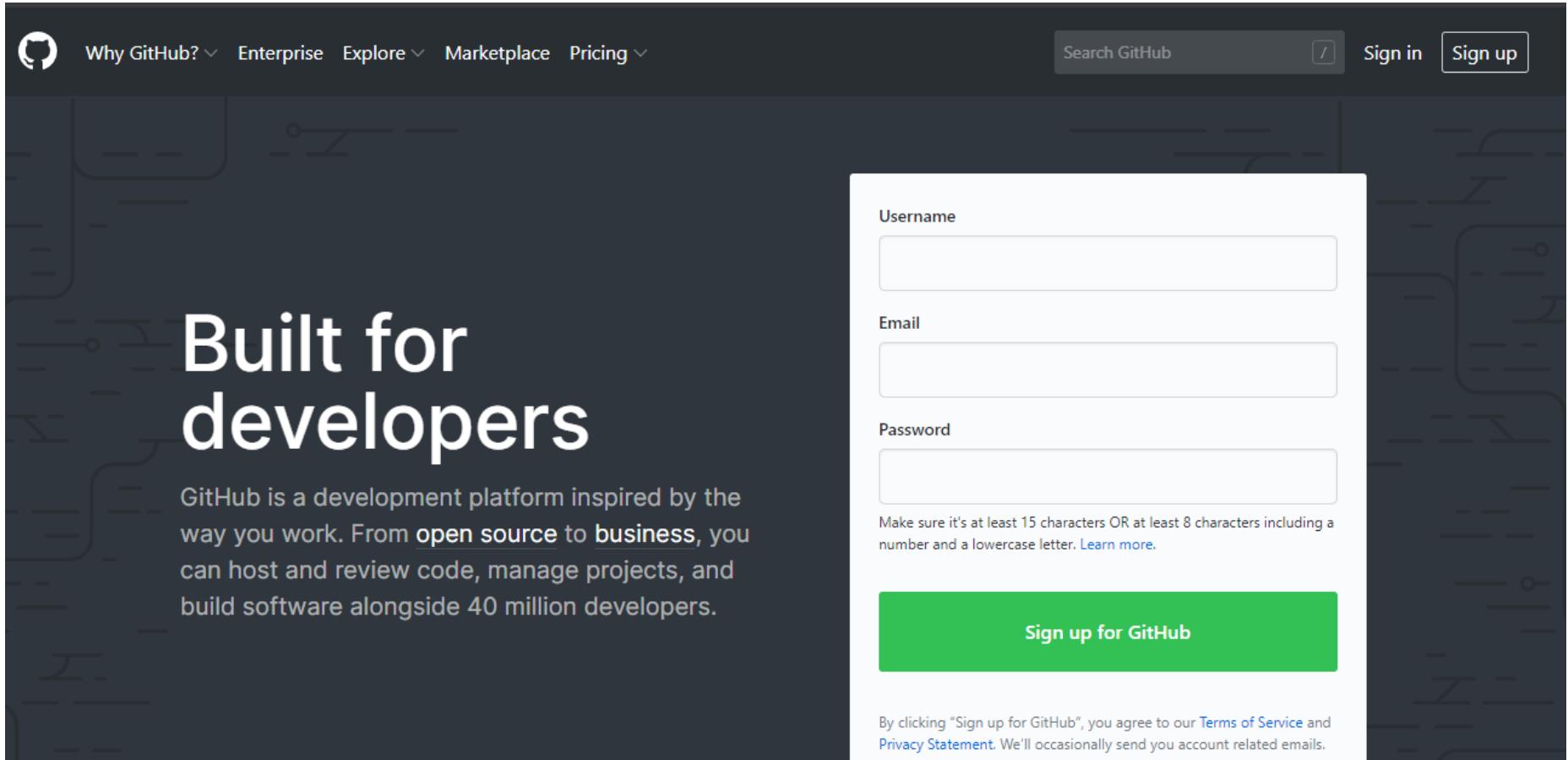
### Repositórios Remotos com o GitHub

- Repositório alocado na Nuvem;
- Maior rede social do códigos fontes do mundo;
- Os principais códigos de projetos OpenSource estão no GitHub;
- Crie uma conta no GitHub;

<https://github.com/>

# 1. Introdução

## GitHub



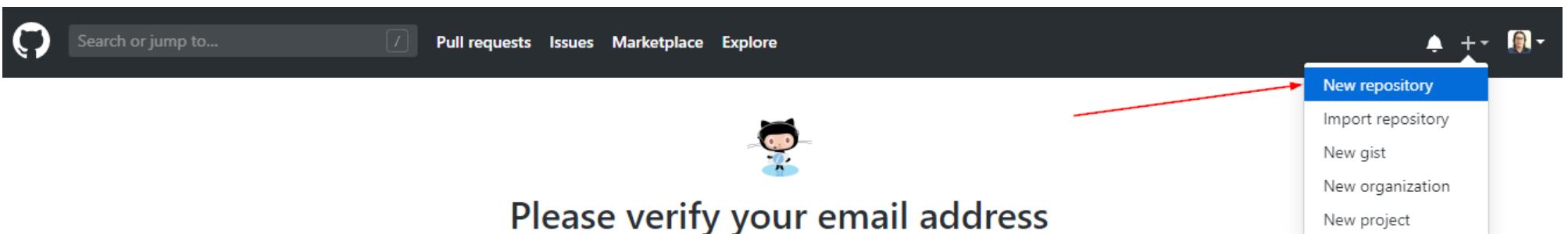
The screenshot shows the GitHub sign-up interface. At the top, there's a navigation bar with links for "Why GitHub?", "Enterprise", "Explore", "Marketplace", and "Pricing". On the right side of the bar are "Search GitHub", "Sign in", and "Sign up" buttons. The main area features a large, bold title "Built for developers" on a dark background with a faint circuit board pattern. Below the title is a paragraph of text: "GitHub is a development platform inspired by the way you work. From open source to business, you can host and review code, manage projects, and build software alongside 40 million developers." To the right of this text is a sign-up form with fields for "Username", "Email", and "Password". A note below the password field specifies that the password must be at least 15 characters or 8 characters including a number and a lowercase letter, with a link to "Learn more". A large green "Sign up for GitHub" button is at the bottom of the form. At the very bottom of the page, a small note states: "By clicking "Sign up for GitHub", you agree to our [Terms of Service](#) and [Privacy Statement](#). We'll occasionally send you account related emails."

# 1. Introdução

## GitHub

### Repositórios Remotos com o GitHub

- Criar um Repositório no GitHub:



The screenshot shows the GitHub homepage. At the top, there is a navigation bar with links for "Pull requests", "Issues", "Marketplace", and "Explore". On the far right of the header, there is a user icon and a dropdown menu. A red arrow points from the text "New repository" in this menu to the "New repository" button. Below the header, there is a large GitHub logo icon. The main content area has a heading "Please verify your email address" followed by instructions: "Before you can contribute on GitHub, we need you to verify your email address. An email containing verification instructions was sent to [zambon@outlook.com.br](mailto:zambon@outlook.com.br)". At the bottom, there is a link "Didn't get the email? [Resend verification email](#) or [change your email settings](#)".

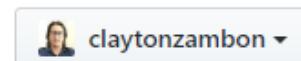
# 1. Introdução

## GitHub

### Create a new repository

A repository contains all project files, including the revision history. Already have a project repository elsewhere?  
[Import a repository.](#)

Owner



claytonzambon ▾

Repository name \*

A text input field with the placeholder text "estruturadedados2020-01".✓

Great repository names are short and memorable. Need inspiration? How about [fluffy-fiesta](#)?

Description (optional)

A text input field with the placeholder text "Repositório para Estrutura de Dados 1 2020/01".

Public

Anyone can see this repository. You choose who can commit.

Private

You choose who can see and commit to this repository.

Skip this step if you're importing an existing repository.

Initialize this repository with a README

This will let you immediately clone the repository to your computer

Add .gitignore: None ▾

Add a license: None ▾



**Não irei marcar porque  
localmente o comando git init já  
criou estes arquivos. Se marcar o  
histórico ficará diferente.**

**Create repository**

# 1. Introdução

## GitHub

Quick setup — if you've done this kind of thing before

 Set up in Desktop

or

HTTPS

SSH

<https://github.com/claytonzambon/estruturadedados2020-01.git>



Get started by [creating a new file](#) or [uploading an existing file](#). We recommend every repository include a [README](#), [LICENSE](#), and [.gitignore](#).

...or create a new repository on the command line

```
echo "# estruturadedados2020-01" >> README.md
git init
git add README.md
git commit -m "first commit"
git remote add origin https://github.com/claytonzambon/estruturadedados2020-01.git
git push -u origin master
```



...or push an existing repository from the command line

```
git remote add origin https://github.com/claytonzambon/estruturadedados2020-01.git
git push -u origin master
```



...or import code from another repository

You can initialize this repository with code from a Subversion, Mercurial, or TFS project.

[Import code](#)

# 1. Introdução

## GitHub

### Criando e Adicionando uma chave SSH

- Para que o GitHub saiba que você é o usuário certo que está enviando arquivos para o repositório remoto, é necessário identificar você de alguma forma;
- Para fazer a identificação o GitHub utiliza o SSH;
  - SSH: é baseado em chaves onde existe uma pública e uma privada. A chave privada consegue abrir a chave pública;
  - Então enviamos a chave pública para o servidor, GitHub, e com a chave privada da nossa máquina somos capazes de abrir a chave pública do servidor sendo possível subir nosso código para o repositório remoto. Maiores informações no link abaixo

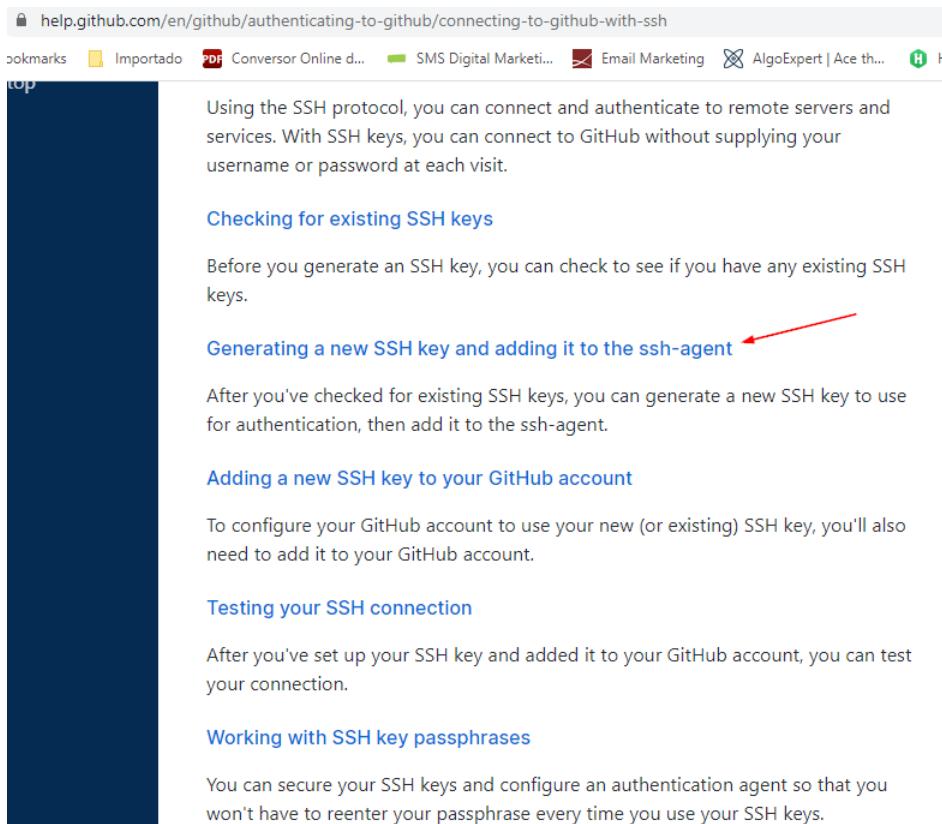
<https://help.github.com/en/github/authenticating-to-github/about-ssh>

# 1. Introdução

## GitHub

### Gerando a chave SSH

<https://help.github.com/en/github/authenticating-to-github/connecting-to-github-with-ssh>



The screenshot shows a web browser window with the URL <https://help.github.com/en/github/authenticating-to-github/connecting-to-github-with-ssh>. The page content is about generating an SSH key for GitHub. It includes sections on checking existing keys, generating a new key, adding it to the ssh-agent (with a red arrow pointing to this section), adding it to the GitHub account, testing the connection, and working with SSH key passphrases.

Using the SSH protocol, you can connect and authenticate to remote servers and services. With SSH keys, you can connect to GitHub without supplying your username or password at each visit.

[Checking for existing SSH keys](#)

Before you generate an SSH key, you can check to see if you have any existing SSH keys.

[Generating a new SSH key and adding it to the ssh-agent](#) 

After you've checked for existing SSH keys, you can generate a new SSH key to use for authentication, then add it to the ssh-agent.

[Adding a new SSH key to your GitHub account](#)

To configure your GitHub account to use your new (or existing) SSH key, you'll also need to add it to your GitHub account.

[Testing your SSH connection](#)

After you've set up your SSH key and added it to your GitHub account, you can test your connection.

[Working with SSH key passphrases](#)

You can secure your SSH keys and configure an authentication agent so that you won't have to reenter your passphrase every time you use your SSH keys.

# 1. Introdução

## GitHub

### Generating a new SSH key

## Gerando a chave SSH

1 Open Git Bash.

2 Paste the text below, substituting in your GitHub email address.

```
$ ssh-keygen -t rsa -b 4096 -C "your_email@example.com"
```

This creates a new ssh key, using the provided email as a label.

```
> Generating public/private rsa key pair.
```

3 When you're prompted to "Enter a file in which to save the key," press Enter. This accepts the default file location.

```
> Enter a file in which to save the key (/c/Users/you/.ssh/id_rsa):[Pres
```

4 At the prompt, type a secure passphrase. For more information, see "[Working with SSH key passphrases](#)".

```
> Enter passphrase (empty for no passphrase): [Type a passphrase]  
> Enter same passphrase again: [Type passphrase again]
```

# 1. Introdução

## GitHub

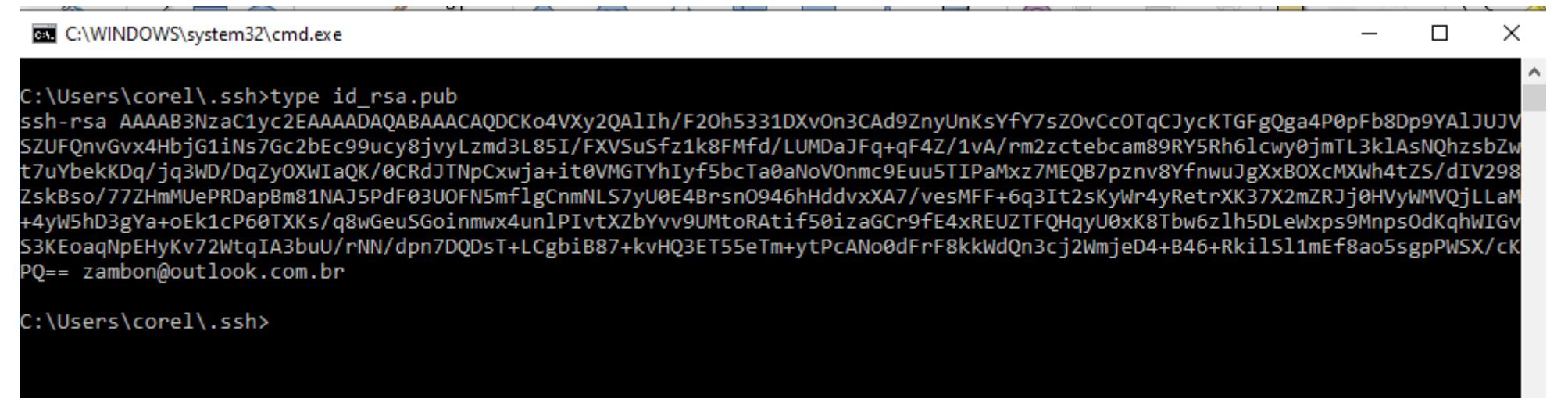
### Gerando a chave SSH

```
C:\WINDOWS\system32\cmd.exe
C:\git-projetos>ssh-keygen -t rsa -b 4096 -C "zambon@outlook.com.br"
Generating public/private rsa key pair.
Enter file in which to save the key (C:\Users\corel/.ssh/id_rsa):
Created directory 'C:\Users\corel/.ssh'.
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in C:\Users\corel/.ssh/id_rsa.
Your public key has been saved in C:\Users\corel/.ssh/id_rsa.pub.
The key fingerprint is:
SHA256:BhJ+3vVFsXmGXenKZNRTGEGRWE81NjSU01+ZFN3SJZA zambon@outlook.com.br
The key's randomart image is:
+---[RSA 4096]----+
|          .*#^^|
|          Eo=^@|
| o o . . * .X|
| + o . . + +.|
|   . S   = .|
|       o      |
+---[SHA256]-----+
C:\git-projetos>
```

# 1. Introdução

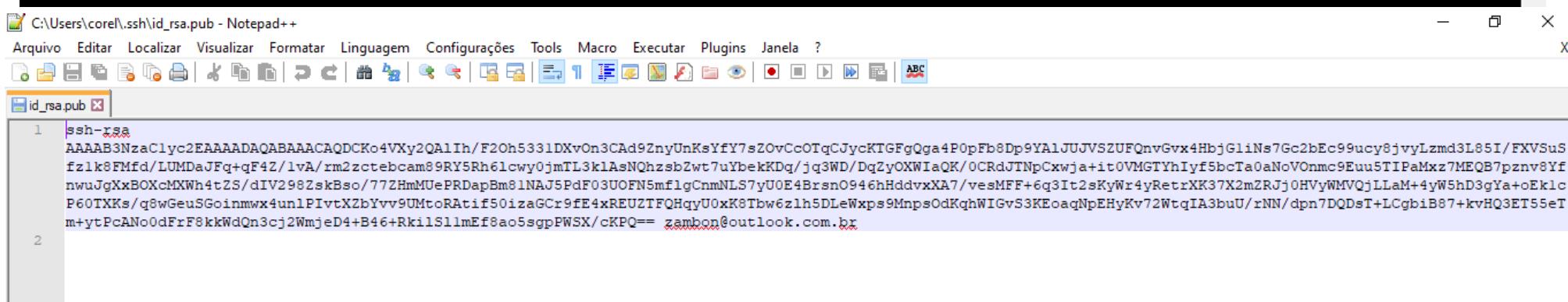
## GitHub

### Visualizando a chave SSH



```
C:\Users\corel\.ssh>type id_rsa.pub
ssh-rsa AAAAB3NzaC1yc2EAAAQABAAQADCKo4VXy2QAlIh/F20h5331DXvOn3CAAd9ZnyUnKsYfY7sZ0vCcOTqCJycKTGFgQga4P0pFb8Dp9YAlJUJV
SZUFQnvGvx4HbjG1iNs7Gc2bEc99ucy8jvyLzmd3L85I/FXVSuSfz1k8FMfd/LUMDaJFq+qF4Z/1vA/rm2zctebcam89RY5Rh6lcwy0jmTL3k1AsNQhzsbZw
t7uYbekKDq/jq3WD/DqZyOXWIaQK/0CRdJTNpCxwj+a+it0VMGTYhIyf5bcTa0aNoVOnmc9Euu5TIPaMxz7MEQB7pz nv8YfnwuJgXxB0XcMXWh4tZs/dIV298
ZskBso/77ZHmMUEPRDapBm81NAJ5PdF03UOFN5mf1gCnmNLS7yU0E4Brsn0946hHddvxXA7/vesMFF+6q3It2sKyWr4yRetrXK37X2mZRJj0HVyWMVQjLLaM
+4yW5hD3gYa+oEk1cP60TXKs/q8wGeuSGoinmwx4un1PIvtXzbYvv9UMtoRAtif50izaGCr9fE4xREUZTFQHqyU0xK8Tbw6zlh5DLeWxps9Mnps0dKqhWIGv
S3KEoaqNpEHyKv72WtqIA3buU/rNN/dpn7DQDsT+LCgbib87+kvHQ3ET55eTm+ytPcANo0dFrF8kkWdQn3cj2WmjeD4+B46+RkilSl1mEf8ao5sgpPWSX/cK
PQ== zambon@outlook.com.br

C:\Users\corel\.ssh>
```



```
Arquivo Editar Localizar Visualizar Formatar Linguagem Configurações Tools Macro Executar Plugins Janela ?
File Edit View Insert Block Tools Plugins Help ABC
id_rsa.pub
1 ssh-rsa
AAAAB3NzaC1yc2EAAAQABAAQADCKo4VXy2QAlIh/F20h5331DXvOn3CAAd9ZnyUnKsYfY7sZ0vCcOTqCJycKTGFgQga4P0pFb8Dp9YAlJUJV
SZUFQnvGvx4HbjG1iNs7Gc2bEc99ucy8jvyLzmd3L85I/FXVSuSfz1k8FMfd/LUMDaJFq+qF4Z/1vA/rm2zctebcam89RY5Rh6lcwy0jmTL3k1AsNQhzsbZw
t7uYbekKDq/jq3WD/DqZyOXWIaQK/0CRdJTNpCxwj+a+it0VMGTYhIyf5bcTa0aNoVOnmc9Euu5TIPaMxz7MEQB7pz nv8YfnwuJgXxB0XcMXWh4tZs/dIV298
ZskBso/77ZHmMUEPRDapBm81NAJ5PdF03UOFN5mf1gCnmNLS7yU0E4Brsn0946hHddvxXA7/vesMFF+6q3It2sKyWr4yRetrXK37X2mZRJj0HVyWMVQjLLaM
+4yW5hD3gYa+oEk1cP60TXKs/q8wGeuSGoinmwx4un1PIvtXzbYvv9UMtoRAtif50izaGCr9fE4xREUZTFQHqyU0xK8Tbw6zlh5DLeWxps9Mnps0dKqhWIGv
S3KEoaqNpEHyKv72WtqIA3buU/rNN/dpn7DQDsT+LCgbib87+kvHQ3ET55eTm+ytPcANo0dFrF8kkWdQn3cj2WmjeD4+B46+RkilSl1mEf8ao5sgpPWSX/cK
PQ== zambon@outlook.com.br
```

# 1. Introdução

## GitHub

### Visualizando a chave SSH

- Copie a chave;
- No seu GitHub acesse a opção Settings
- Dentro de Settings acesse a opção SSH and GPG keys;
- Clique em “New SSH Key” e cole a sua chave no campo Key;
- No campo “Title” informe um título de acordo com a chave. Normalmente é colocado o nome da máquina onde você está trabalhando. Pode colocar também Home, Work, estudo, etc;
- Feito os passos acima clique em “ADD SSH Key”

# 1. Introdução

## GitHub

The screenshot shows the GitHub homepage. At the top, there is a navigation bar with links for Issues, Marketplace, and Explore. On the right side of the header, there is a user profile icon and a dropdown menu. The dropdown menu is open, showing the following options:

- Signed in as claytonzambon
- Set status
- Your profile
- Your repositories
- Your projects
- Your stars
- Your gists
- Feature preview
- Help
- Settings** (this option is highlighted with a blue background and has a red arrow pointing to it)
- Sign out

Below the header, there is a large callout text: "Learn Git and GitHub without any code! In this guided tour, you'll learn how to clone a repository, make changes, and push your work back up to GitHub. By the end of this guide, you'll start a branch, write comments, and open a pull request." A green button labeled "Read the guide" is located below this text.

At the bottom of the page, there is a repository card for "claytonzambon/dos2020-01". The card includes the repository name, a star count of 1, a fork count of 0, and a link to the repository. Below the repository card, there is a navigation bar with links for Actions, Projects, Wiki, Security, Insights, and Settings.

# 1. Introdução

## GitHub

The screenshot shows the GitHub 'Personal settings' sidebar on the left with various tabs: Personal settings, Profile (selected), Account, Security, Security log, Emails, Notifications, Billing, SSH and GPG keys (with a red arrow pointing to it), Blocked users, Repositories, Organizations, Saved replies, and Applications.

**Public profile**

**Name**  
Clayton Zambon  
Your name may appear around GitHub where you contribute or are mentioned. You can remove it at any time.

**Public email**  
Select a verified email to display

You have set your email address to private. To toggle email privacy, go to [email settings](#) and uncheck "Keep my email address private."

**Bio**  
Tell us a little bit about yourself  
You can @mention other users and organizations to link to them.

**URL**

**Company**

# 1. Introdução

## GitHub

Personal settings

Profile

Account

Security

Security log

Emails

Notifications

Billing

...

---

### SSH keys

There are no SSH keys associated with your account.

Check out our guide to [generating SSH keys](#) or troubleshoot [common SSH Problems](#).

New SSH key

---

### GPG keys

There are no GPG keys associated with your account.

Learn how to [generate a GPG key](#) and add it to your account.

New GPG key

# 1. Introdução

## GitHub

Personal settings

Profile

Account

Security

Security log

Emails

Notifications

Billing

**SSH and GPG keys**

Blocked users

Repositories

### SSH keys / Add new

Title

Key

```
ssh-rsa AAAAB3NzaC1yc2EAAAQABAAQDCKo4VXy2QAllh/F2Oh5331DXvOn3CAAd9ZnyUnKsYfY7sZOvCcOTqCJyc KTGFgQga4P0pFb8Dp9YAIUUVSUFQnvGvx4HbjG1iNs7Gc2bEc99ucy8jvyLzmd3L85l/FXVSuSz1k8FMfd/LUMDaJF q+qF4Z/1vA/rm2zctebcam89RY5Rh6lcwy0jmTL3klAsNQhzsbZwt7uYbekKDq/jq3WD/DqZyOXWlaQK/0CRdJTNPcCx wja+it0VMGTYhlyf5bcTa0aNoVOnmc9Euu5TIPaMxz7MEQB7pznv8YfnwuJgXxBOXcMXWh4tZS/dIV298ZskBso/77Z HmMUePRDapBm81NAJ5PdF03UOFN5mflqCnmNLS7yU0E4BrsnO946hHddvxXA7/vesMFF+6q3lt2sKyWr4yRetrXK 37X2mZRJj0HVyWMVQjLLaM+4yW5hD3gYa+oEk1cP60TXKs/q8wGeuSGoinmwx4uniPlvtXZbYvv9UMtoRAtif50iza GCr9fE4xREUZTFQHqyU0xK8Tbw6zlh5DLeWxps9MnpsOdKqhWIGvS3KEoaqNpEHyKv72WtqIA3buU/rNN/dpn7DQ DsT+LCgbIB87+kvHQ3ET55eTm+ytPcANo0dFrF8kkWdQn3cj2WmjeD4+B46+RkilSI1mEf8ao5sgpPWSX/cKPQ== zambon@outlook.com.br
```

Add SSH key

# 1. Introdução

## GitHub

Personal settings

Profile

Account

Security

Security log

Emails

Notifications

Billing

**SSH and GPG keys**

Blocked users

Repositories

### SSH keys

This is a list of SSH keys associated with your account. Remove any keys that you do not recognize.

**Work - Lenovo**  
 f3:94:26:61:d5:82:c4:30:ab:47:26:4c:b8:47:1d:1c  
Added on 5 Feb 2020  
Never used — Read/write

[Delete](#)

Check out our guide to [generating SSH keys](#) or troubleshoot [common SSH Problems](#).

### GPG keys

New GPG key

There are no GPG keys associated with your account.

Learn how to [generate a GPG key](#) and add it to your account.

# 1. Introdução

## GitHub

### Ligando o repositório local a um remoto

The screenshot shows a GitHub user profile for `claytonzambon`. The top navigation bar includes links for Pull requests, Issues, Marketplace, and Explore. Below the navigation bar, there are sections for Overview, Repositories (1), Projects (0), Packages (0), Stars (0), Followers (0), and Following (0). A search bar allows finding a repository, with filters for Type: All and Language. The main content area displays a repository named `estruturadedados2020-01`, described as a 'Repositório para Estrutura de Dados 1 2020/01' updated 4 hours ago. To the right, a sidebar menu is open, showing options like Set status, Your profile, **Your repositories** (which is highlighted in blue), Your projects, Your stars, Your gists, Feature preview, Help, Settings, and Sign out.

# 1. Introdução

## GitHub

Code Issues 0 Pull requests 0 Actions Projects 0 Wiki Security Insights Settings

Quick setup — if you've done this kind of thing before

Set up in Desktop or HTTPS SSH git@github.com:claytonzambon/estruturadedados2020-01.git

Get started by [creating a new file](#) or [uploading an existing file](#). We recommend every repository include a [README](#), [LICENSE](#), and [.gitignore](#).

...or create a new repository on the command line

```
echo "# estruturadedados2020-01" >> README.md  
git init  
git add README.md  
git commit -m "first commit"  
git remote add origin git@github.com:claytonzambon/estruturadedados2020-01.git  
git push -u origin master
```

...or push an existing repository from the command line

```
git remote add origin git@github.com:claytonzambon/estruturadedados2020-01.git  
git push -u origin master
```

# 1. Introdução

## GitHub

### Ligando o repositório local a um remoto

`git remote add origin git@github.com:claytonzambon/estruturadedados2020-01.git`

- No lugar de “origin” pode ser outro nome se quiser

- `git remote`

- `git remote -v`

```
C:\WINDOWS\system32\cmd.exe
C:\git-projetos>git remote add origin git@github.com:claytonzambon/estruturadedados2020-01.git
C:\git-projetos>git remote
origin

C:\git-projetos>git remote -v
origin  git@github.com:claytonzambon/estruturadedados2020-01.git (fetch)
origin  git@github.com:claytonzambon/estruturadedados2020-01.git (push)

C:\git-projetos>
```

# 1. Introdução

## GitHub

### Ligando o repositório local a um remoto

**git push -u origin master**

- Envia todos os logs e modificações que eu posso localmente para o repositório remoto;
- Sintaxe:

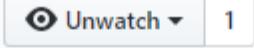
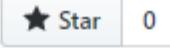
`git push -u <Para onde vai> <De onde vem>`

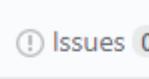
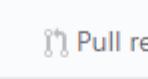
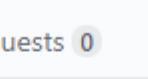
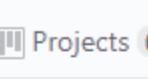
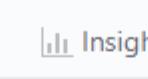
# 1. Introdução

## GitHub

### Ligando o repositório local a um remoto

 [claytonzambon / estruturadedados2020-01](#)

Repositório para Estrutura de Dados 1 2020/01 

[Manage topics](#)

 1 commit  1 branch  0 packages  0 releases  1 contributor

Branch: master     

 claytonzambon Adicionando arquivo readme.txt	Latest commit a2ade9b 21 hours ago
 <a href="#">readme.txt</a> Adicionando arquivo readme.txt	21 hours ago
 <a href="#">readme.txt</a>	
Olá mundo lages-sc	

# 1. Introdução

## GitHub

### Enviando mudanças para um repositório remoto

- git commit -am “Primeira mudança”
- git push origin master

```
C:\git-projetos>git add readme.txt

C:\git-projetos>git commit -m "Primeira alteração"
[master 5190e45] Primeira alteração
 1 file changed, 3 insertions(+), 1 deletion(-)

C:\git-projetos>git push -u origin master
Warning: Permanently added the RSA host key for IP address '18.228.67.229' to the list of known hosts.
Enumerating objects: 5, done.
Counting objects: 100% (5/5), done.
Writing objects: 100% (3/3), 290 bytes | 290.00 KiB/s, done.
Total 3 (delta 0), reused 0 (delta 0)
To github.com:claytonzambon/estruradadedados2020-01.git
  a2ade9b..5190e45 master -> master
Branch 'master' set up to track remote branch 'master' from 'origin'.

C:\git-projetos>git commit -am "Segunda mudança"
[master 94c4f61] Segunda mudança
 1 file changed, 3 insertions(+), 1 deletion(-)

C:\git-projetos>git push origin master
Enumerating objects: 5, done.
Counting objects: 100% (5/5), done.
Writing objects: 100% (3/3), 290 bytes | 290.00 KiB/s, done.
Total 3 (delta 0), reused 0 (delta 0)
To github.com:claytonzambon/estruradadedados2020-01.git
  5190e45..94c4f61 master -> master

C:\git-projetos>
```

# 1. Introdução

## GitHub

### Clonando repositórios remotos

- É possível clonar todo o projeto para sua máquina local;

The screenshot shows a GitHub repository page for a project named 'claytonzambon/estrutura-de-dados'. The top navigation bar includes metrics: 3 commits, 1 branch, 0 packages, 0 releases, and 1 contributor. A red arrow points from the 'contributor' link to the 'Clone or download' button. Below the metrics, there are buttons for 'Branch: master', 'New pull request', 'Create new file', 'Upload files', 'Find file', and the highlighted 'Clone or download' button. The repository content area shows two 'readme.txt' files. The left file contains the text 'Olá mundo', 'primeira mudança', and 'segunda mudança'. The right file contains the text 'Segunda mudança'. To the right of the files is a cloning section with 'Clone with SSH' and 'Use HTTPS' options, and a copy icon. Below this are 'Open in Desktop' and 'Download ZIP' buttons. Red arrows point from the 'Clone with SSH' input field and the 'Download ZIP' button towards the 'Clone or download' button.

# 1. Introdução

## GitHub

### Clonando repositórios remotos

```
g clone git@github.com:claytonzambon/estruturadedados2020-01.git estruturasdedados-clone
```

- O comando acima pega tudo que está no repositório remoto e copia para sua pasta local;
- Isto é útil quando você realiza produções em vários locais diferentes;

# 1. Introdução

## GitHub

### Fazendo FORK de um projeto

- Esta opção permite você pegar um projeto que não é seu e fazer uma cópia dele pra você;

- Imagine que você viu um repositório que você quer contribuir;
- Você faz um FORK, realiza as modificações e depois faz um Pull request pra pessoa avisando sobre a correção ou mudanças que eu fiz;
- Para usar esta opção você tem que ser membro de uma organização;
- Ele é diferente do clone porque o clone só consigo fazer para os meus repositórios. Se eu não for dono do repositório posso clonar mas não enviar alterações.

# 1. Introdução

## GitHub

### Fazendo FORK de um projeto

The screenshot shows the GitHub homepage. At the top, there is a dark navigation bar with the GitHub logo, a search bar, and links for Pull requests, Issues, Marketplace, and Explore. Below the navigation bar, the main content area has a sidebar on the left containing sections for Repositories, a search bar, and a link to 'claytonzambon/estruturadedados2020-01'. The main content area features a large green callout box with the text 'Learn Git and GitHub without any code!' and a description below it. At the bottom of the page, there are two buttons: 'Read the guide' (in a green box) and 'Start a project'.

Repositories

New

Search or jump to...

Pull requests Issues Marketplace Explore

Find a repository...

claytonzambon/estruturadedados2020-01

Working with a team?

GitHub is built for collaboration. Set up an organization to improve the way your team works together, and get access to more features.

Create an organization

Learn Git and GitHub without any code!

Using the Hello World guide, you'll create a repository, start a branch, write comments, and open a pull request.

Read the guide

Start a project

# 1. Introdução

## GitHub

### Fazendo FORK de um projeto

#### Set up your Team

Organization account name \*

 ✓

This will be the name of your account on GitHub.  
Your URL will be: <https://github.com/ED-01-2020-01>.

Contact email \*

 ✓

This organization belongs to: \*

My personal account

i.e., claytonzambon (Clayton Zambon)

A business or institution

For example: GitHub, Inc., Example Institute, American Red Cross

**Next**

By creating an account, you agree to the [Terms of Service](#). For more information about GitHub's privacy practices, see the [GitHub Privacy Statement](#). We'll occasionally send you account-related emails.

# Obrigado!