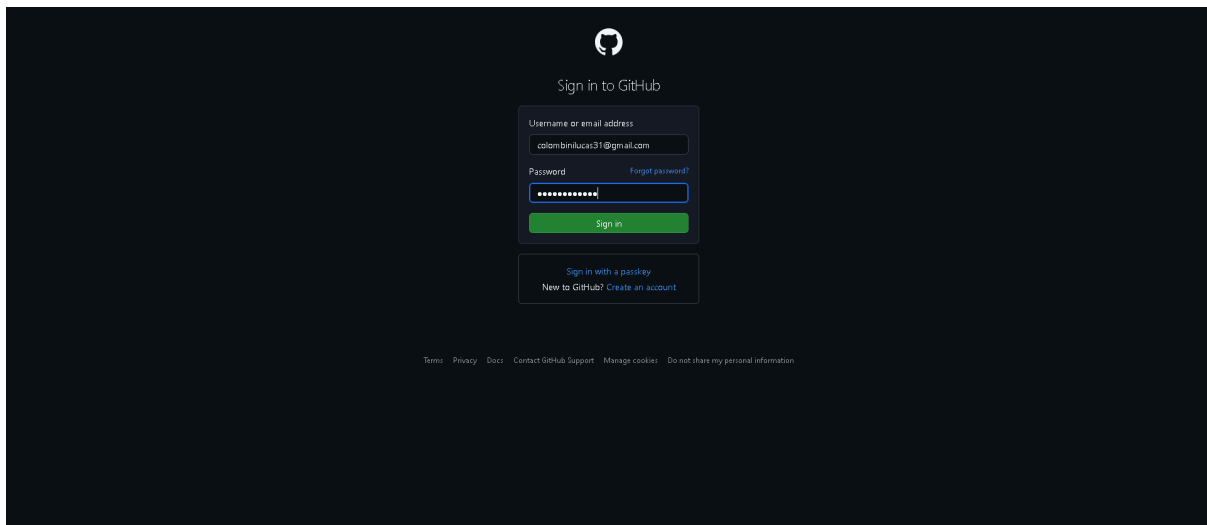


Git & GitHub

LUCAS FERREIRA COLOMBINI
422142408

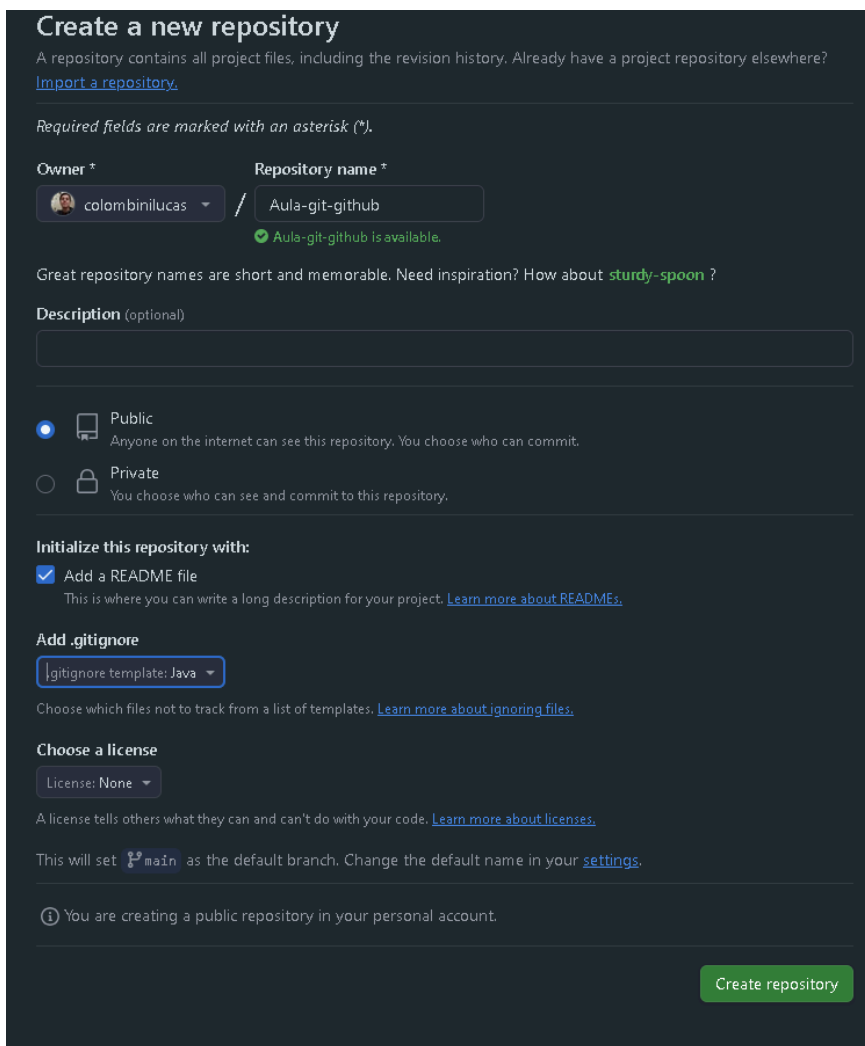
1. Login no site <https://github.com/>
2. Criando um novo Repositório
3. Clonando o Repositório Aula-gi-github
4. Mudando de diretório cd Aula-git-github e detalhamento do conteúdo do diretório
5. Git Status exibe estado do diretório de trabalho e da área de preparação
6. Escolhendo o workspace na IDE Eclipse para Aula-gi-git-hub
7. Criando um projeto Java
8. Criando Classe Teste
9. Exibindo o status do diretório com a nova classe criada.
10. Git add adicionando /Teste arquivos ou alterações no diretório de trabalho para a área de preparação do Git e exibindo os Status após as alterações
11. Git commit - m salvar as mudanças que você já adicionou com o git add. A flag -m significa “message” (mensagem), serve pra escrever uma mensagem descrevendo o que foi feito nesse commit.
12. git push é usado para enviar seus commits locais para um repositório remoto
13. Exibindo os Status após o Git push
14. Realizando alterações no repositório remoto e comitando.
15. git fetch é usado para buscar todas as atualizações do repositório remoto, mas sem aplicar essas mudanças no seu código ainda. E exibindo os status do diretório.
16. git pull Baixa as mudanças do repositório remoto e já junta (mergeia) com o seu código local.
17. Visualizando as alterações realizadas pelo git pull na IDE
18. git pull Baixa as mudanças do repositório remoto e já junta (mergeia) com o seu código local.
19. Realizando mais uma alteração no repositório remoto, agora printando o resultado soma.
20. git pull Baixando as mudanças do repositório remoto e juntando com o seu código local.

1. Login no site <https://github.com/>



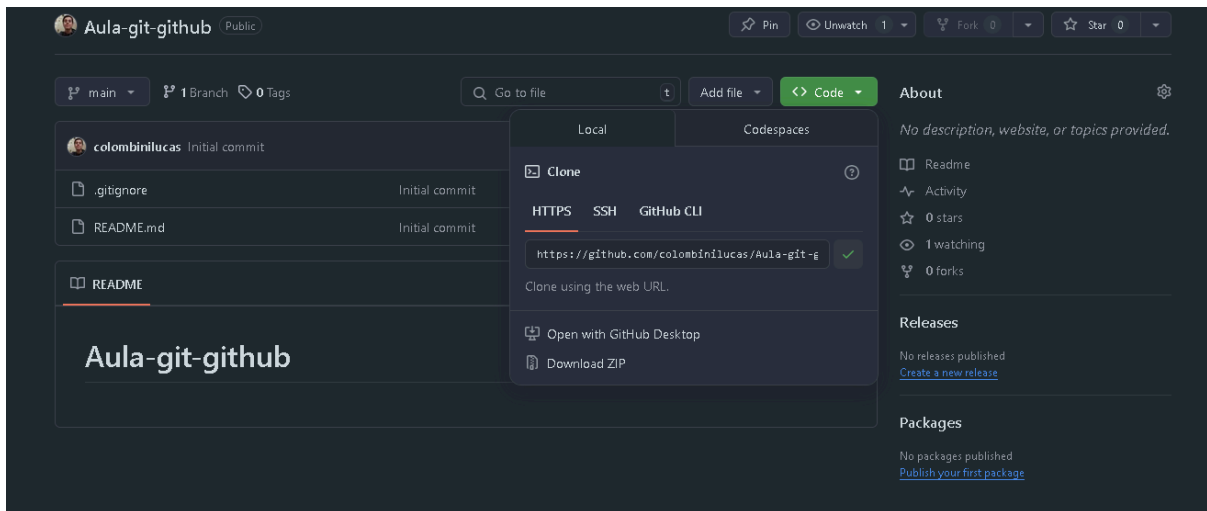
The image shows the GitHub sign-in page. At the top center is the GitHub logo. Below it, the text "Sign in to GitHub" is displayed. The main form has two input fields: "Username or email address" with the value "colombinilucas31@gmail.com" and "Password" with masked characters. A "Forgot password?" link is next to the password field. Below the inputs is a green "Sign in" button. Underneath the button is a link for "Sign in with a passkey" and a link for "New to GitHub? Create an account". At the bottom of the page, there are links for "Terms", "Privacy", "Docs", "Contact GitHub Support", "Manage cookies", and "Do not share my personal information".

2. Criando um novo Repositório



The image shows the "Create a new repository" page on GitHub. The title "Create a new repository" is at the top. Below it, a paragraph explains that a repository contains all project files and revision history, with a link to "Import a repository". A note states "Required fields are marked with an asterisk (*)". The "Owner" field shows a dropdown with "colombinilucas" selected. The "Repository name" field contains "Aula-git-github" and has a green checkmark indicating it is available. Below this, a suggestion for repository names is shown: "Great repository names are short and memorable. Need inspiration? How about **sturdy-spoon** ?". The "Description (optional)" field is empty. The "Visibility" section has two options: "Public" (selected) and "Private". The "Initialize this repository with:" section has a checked box for "Add a README file". The "Add .gitignore" section has a dropdown showing "gitignore template: Java". The "Choose a license" section has a dropdown showing "License: None". At the bottom, there is a note: "You are creating a public repository in your personal account." and a green "Create repository" button.

3. Clonando o Repositório Aula-gi-github



```
MINGW64:/c/Users/Lucas

Lucas@lucas MINGW64 ~
$ git clone https://github.com/colombinilucas/Aula-git-github.git
Cloning into 'Aula-git-github'...
remote: Enumerating objects: 4, done.
remote: Counting objects: 100% (4/4), done.
remote: Compressing objects: 100% (3/3), done.
remote: Total 4 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)
Receiving objects: 100% (4/4), done.

Lucas@lucas MINGW64 ~
$
```

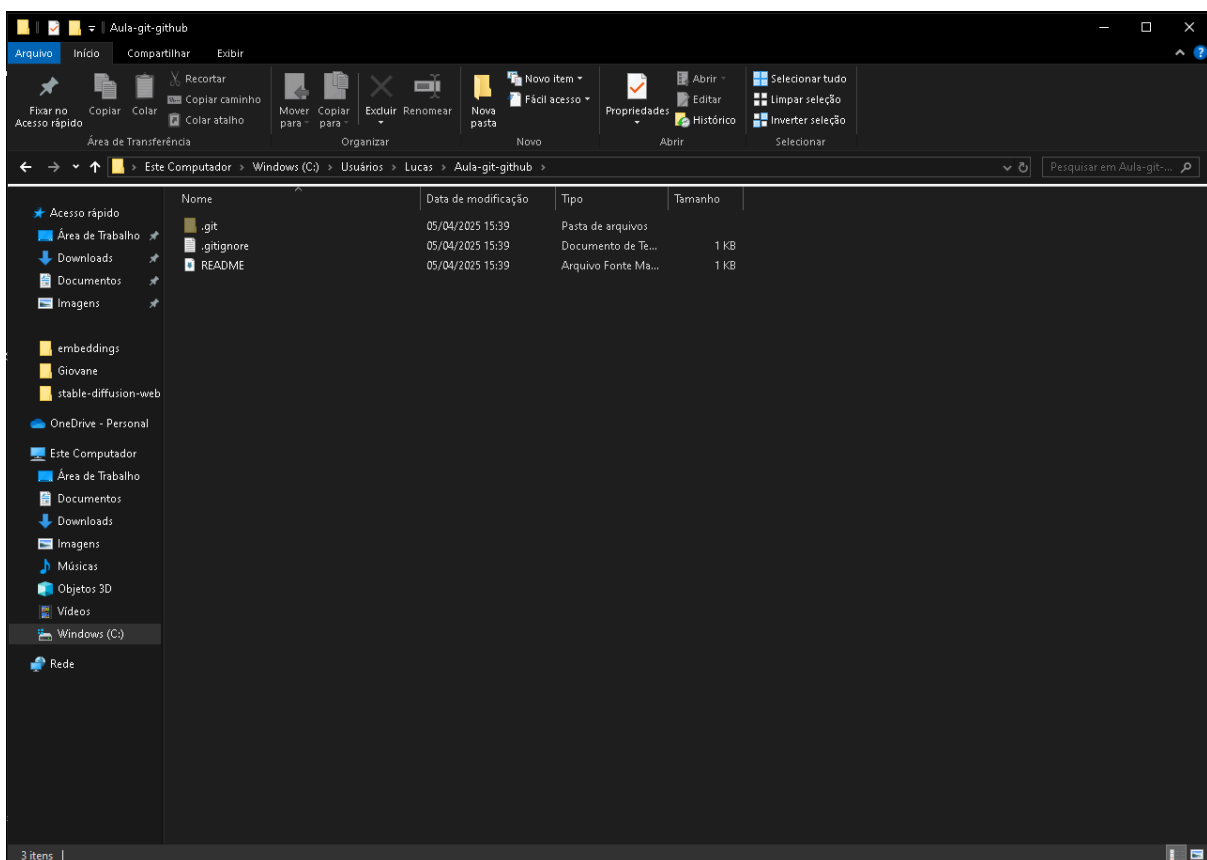
4. Mudando de diretório cd Aula-git-github e detalhamento do conteúdo do diretório

```
MINGW64~/c/Users/Lucas/Aula-git-github

Lucas@lucas MINGW64 ~
$ git clone https://github.com/colombinilucas/Aula-git-github.git
Cloning into 'Aula-git-github'...
remote: Enumerating objects: 4, done.
remote: Counting objects: 100% (4/4), done.
remote: Compressing objects: 100% (3/3), done.
remote: Total 4 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)
Receiving objects: 100% (4/4), done.

Lucas@lucas MINGW64 ~
$ cd Aula-git-github

Lucas@lucas MINGW64 ~/Aula-git-github (main)
$ |
```



5. Git Status exibe estado do diretório de trabalho e da área de preparação

```
MINGW64/c/Users/Lucas/Aula-git-github

Lucas@lucas MINGW64 ~
$ git clone https://github.com/colombinilucas/Aula-git-github.git
Cloning into 'Aula-git-github'...
remote: Enumerating objects: 4, done.
remote: Counting objects: 100% (4/4), done.
remote: Compressing objects: 100% (3/3), done.
remote: Total 4 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)
Receiving objects: 100% (4/4), done.

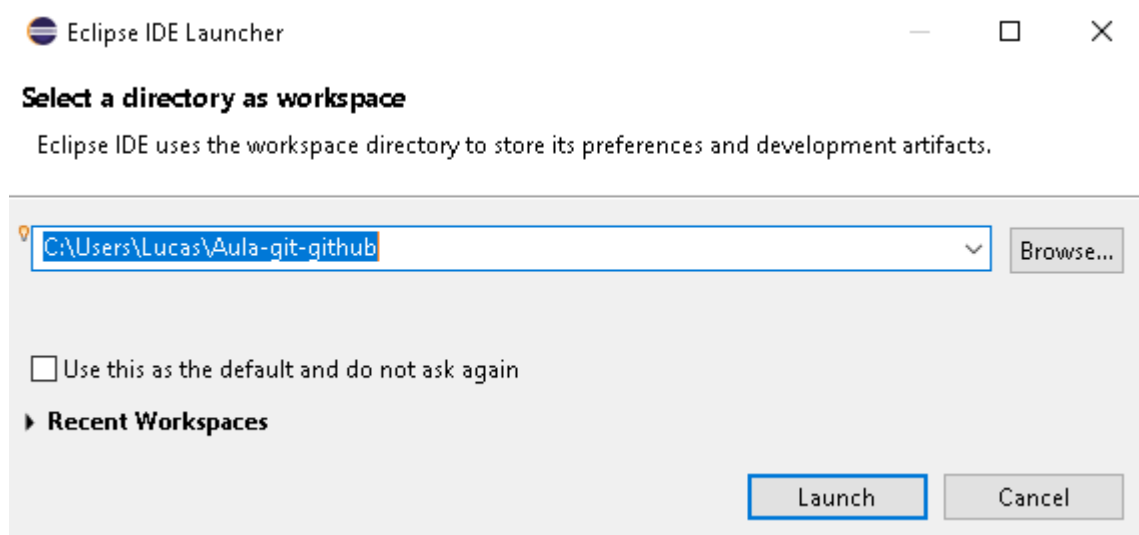
Lucas@lucas MINGW64 ~
$ cd Aula-git-github

Lucas@lucas MINGW64 ~/Aula-git-github (main)
$ git status
On branch main
Your branch is up to date with 'origin/main'.

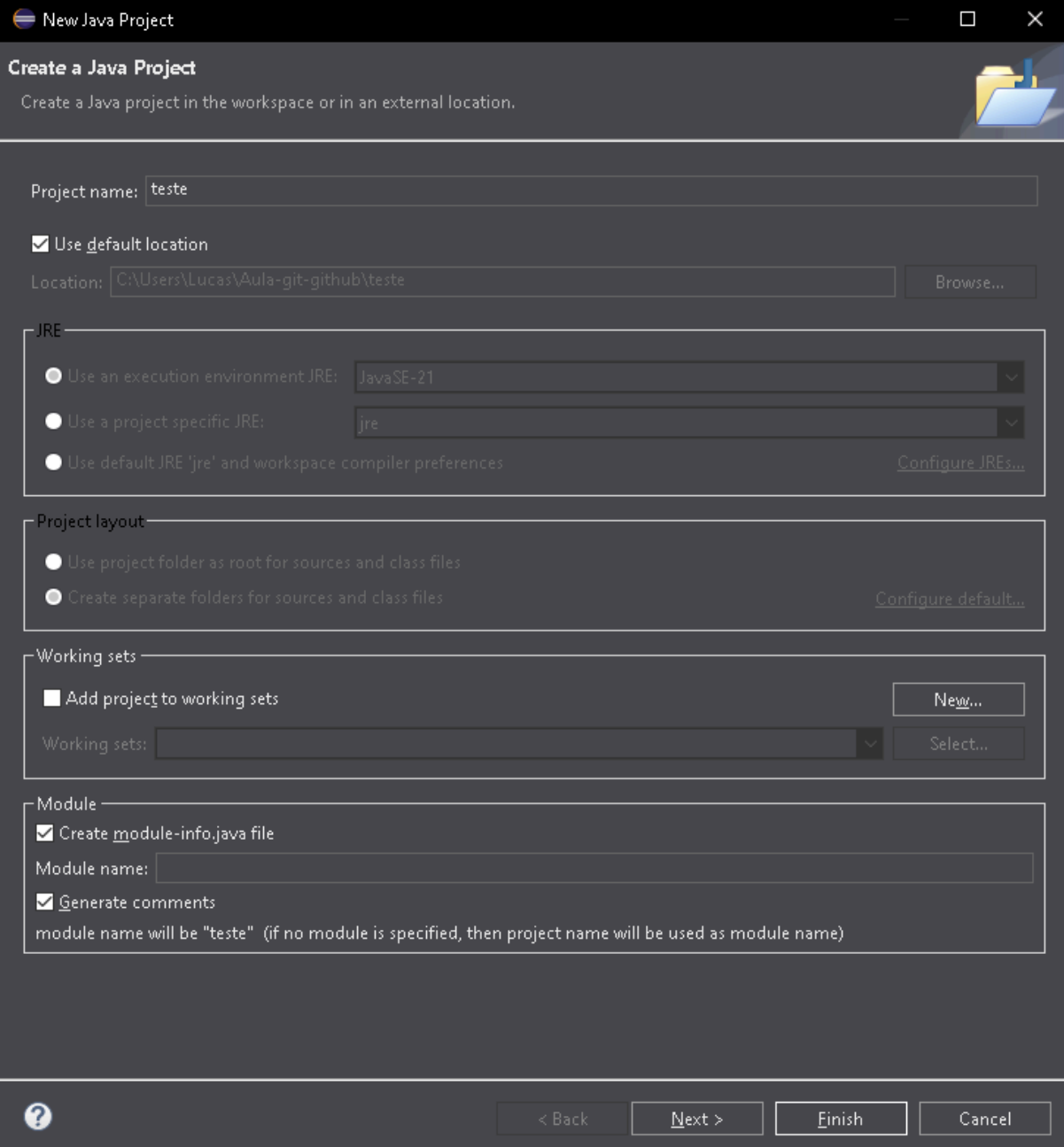
nothing to commit, working tree clean

Lucas@lucas MINGW64 ~/Aula-git-github (main)
$ |
```

6. Escolhendo o workspace na IDE Eclipse para Aula-gi-git-hub



7. Criando um projeto Java



New Java Project

Create a Java project in the workspace or in an external location.

Project name:

☒ Use default location

Location:

JRE

☒ Use an execution environment JRE:

☐ Use a project specific JRE:

☐ Use default JRE 'jre' and workspace compiler preferences [Configure JREs...](#)

Project layout

☒ Use project folder as root for sources and class files

☐ Create separate folders for sources and class files [Configure default...](#)

Working sets

☐ Add project to working sets

Working sets:

Module

☒ Create module-info.java file

Module name:

☒ Generate comments

module name will be "teste" (if no module is specified, then project name will be used as module name)

8. Criando Classe Teste

New Java Class

Create a new Java class.

Source folder:

Package:

☐ Enclosing type:

Name:

Modifiers: ☐ public ☐ package ☐ private ☐ protected
☐ abstract ☐ final ☐ static
☐ none ☐ sealed ☐ non-sealed ☐ final

Superclass:

Interfaces:

Which method stubs would you like to create?

☒ `public static void main(String[] args)`

☐ Constructors from superclass

☒ Inherited abstract methods

Do you want to add comments? (Configure templates and default value [here](#))

☐ Generate comments

9. Exibindo o status do diretório com a nova classe criada.

```
MINGW64:/c/Users/Lucas/Aula-git-github
ntuser.dat.LOG2
ntuser.ini

Lucas@lucas MINGW64 ~
$ cd Aula-git-github

Lucas@lucas MINGW64 ~/Aula-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:   .gitignore

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

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

Lucas@lucas MINGW64 ~/Aula-git-github (main)
$
```

10. Git add adicionando /Teste arquivos ou alterações no diretório de trabalho para a área de preparação do Git e exibindo os Status após as alterações

```
MINGW64:/c/Users/Lucas/Aula-git-github

Lucas@lucas MINGW64 ~/Aula-git-github (main)
$ git add teste/

Lucas@lucas MINGW64 ~/Aula-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)
        new file:   teste/.classpath
        new file:   teste/.project
        new file:   teste/.settings/org.eclipse.core.resources.prefs
        new file:   teste/.settings/org.eclipse.jdt.core.prefs
        new file:   teste/src/module-info.java
        new file:   teste/src/teste/Teste.java

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:   .gitignore

Lucas@lucas MINGW64 ~/Aula-git-github (main)
```


11. **Git commit -m** salvar as mudanças que você já adicionou com o **git add**. A flag **-m** significa “message” (mensagem), serve pra escrever uma mensagem descrevendo o que foi feito nesse commit.

```
MINGW64:/c/Users/Lucas/Aula-git-github

new file:   teste/.project
new file:   teste/.settings/org.eclipse.core.resources.prefs
new file:   teste/.settings/org.eclipse.jdt.core.prefs
new file:   teste/src/module-info.java
new file:   teste/src/teste/Teste.java

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:   .gitignore

Lucas@lucas MINGW64 ~/Aula-git-github (main)
$ git commit -m "Primeiro commit do projeto: classe Java Teste vazia."
[main e744802] Primeiro commit do projeto: classe Java Teste vazia.
 6 files changed, 58 insertions(+)
 create mode 100644 teste/.classpath
 create mode 100644 teste/.project
 create mode 100644 teste/.settings/org.eclipse.core.resources.prefs
 create mode 100644 teste/.settings/org.eclipse.jdt.core.prefs
 create mode 100644 teste/src/module-info.java
 create mode 100644 teste/src/teste/Teste.java

Lucas@lucas MINGW64 ~/Aula-git-github (main)
$
```

12. **git push** é usado para enviar seus commits locais para um repositório remoto

```
MINGW64:/c/Users/Lucas/Aula-git-github

Lucas@lucas MINGW64 ~/Aula-git-github (main)
$ git commit -m "Primeiro commit do projeto: classe Java Teste vazia."
[main e744802] Primeiro commit do projeto: classe Java Teste vazia.
 6 files changed, 58 insertions(+)
 create mode 100644 teste/.classpath
 create mode 100644 teste/.project
 create mode 100644 teste/.settings/org.eclipse.core.resources.prefs
 create mode 100644 teste/.settings/org.eclipse.jdt.core.prefs
 create mode 100644 teste/src/module-info.java
 create mode 100644 teste/src/teste/Teste.java

Lucas@lucas MINGW64 ~/Aula-git-github (main)
$ git push
Enumerating objects: 13, done.
Counting objects: 100% (13/13), done.
Delta compression using up to 4 threads
Compressing objects: 100% (10/10), done.
Writing objects: 100% (12/12), 1.52 KiB | 390.00 KiB/s, done.
Total 12 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)
To https://github.com/colombini/lucas/Aula-git-github.git
   3f40835..e744802  main -> main

Lucas@lucas MINGW64 ~/Aula-git-github (main)
$
```

13. Exibindo os Status após o Git push

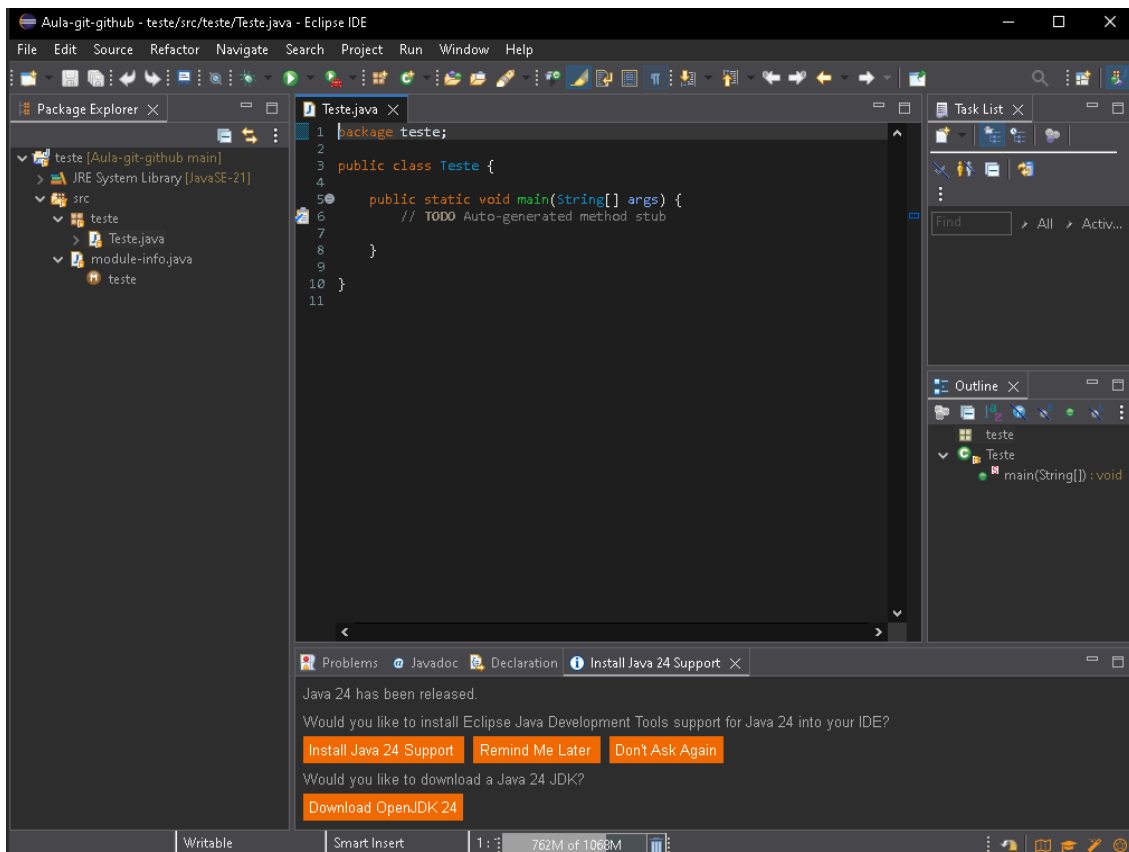
```
MINGW64:/c/Users/Lucas/Aula-git-github
Lucas@lucas MINGW64 ~/Aula-git-github (main)
$ git push
Enumerating objects: 13, done.
Counting objects: 100% (13/13), done.
Delta compression using up to 4 threads
Compressing objects: 100% (10/10), done.
Writing objects: 100% (12/12), 1.52 KiB | 390.00 KiB/s, done.
Total 12 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)
To https://github.com/colombinilucas/Aula-git-github.git
   3f40835..e744802  main -> main

Lucas@lucas MINGW64 ~/Aula-git-github (main)
$ git status
On branch main
Your branch is up to date with 'origin/main'.

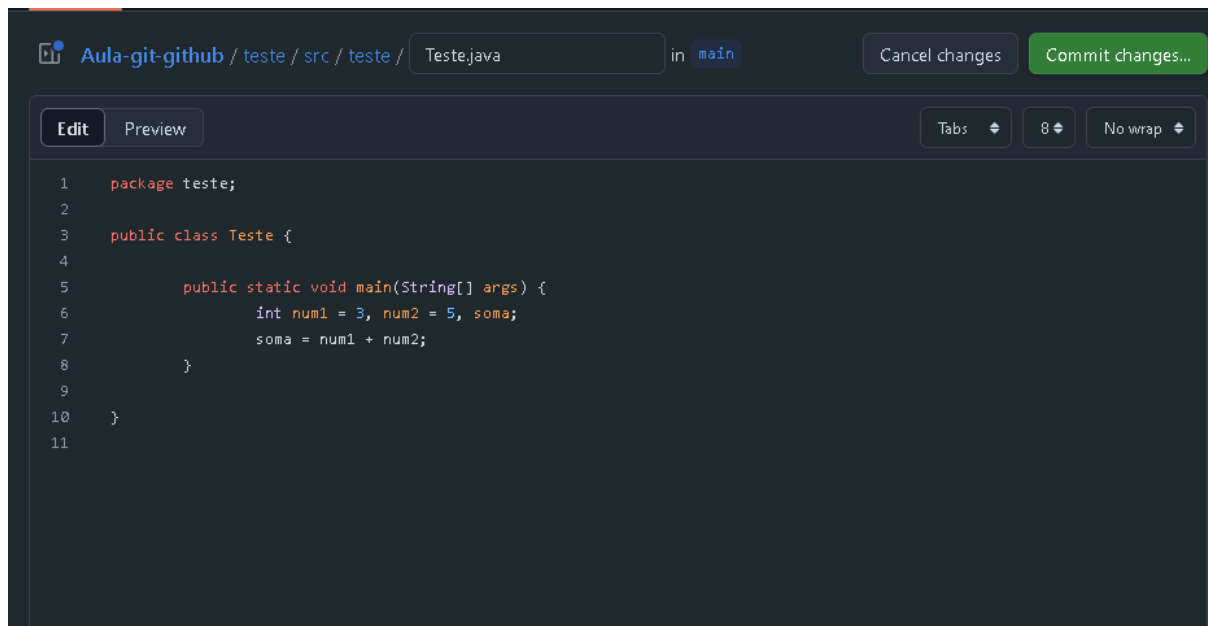
Changes not staged for commit:
  (use "git add <file>..." to update what will be committed)
  (use "git restore <file>..." to discard changes in working directory)
        modified:   .gitignore

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

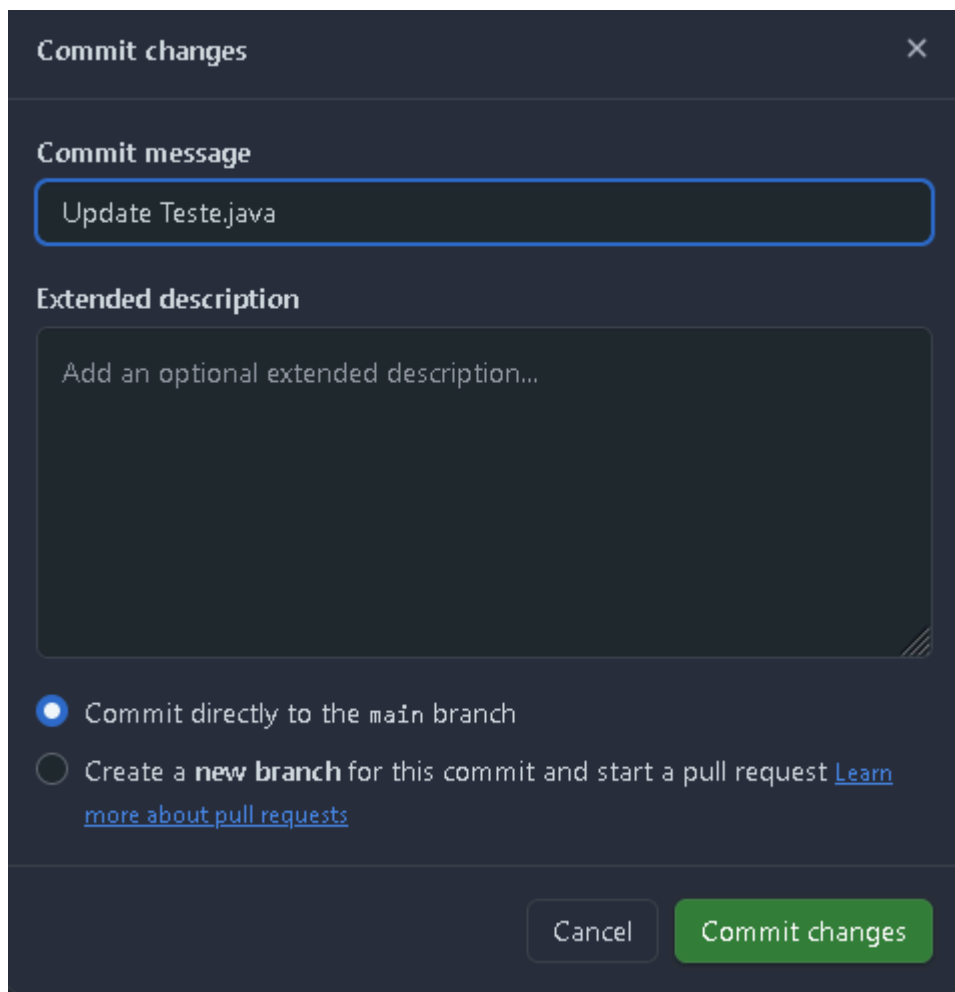
Lucas@lucas MINGW64 ~/Aula-git-github (main)
$
```



14. Realizando alterações no repositório remoto e comitando.



```
1 package teste;
2
3 public class Teste {
4
5     public static void main(String[] args) {
6         int num1 = 3, num2 = 5, soma;
7         soma = num1 + num2;
8     }
9
10 }
11
```



Commit changes ✕

Commit message

Update Teste.java

Extended description

Add an optional extended description...

☒ Commit directly to the main branch

☐ Create a **new branch** for this commit and start a pull request [Learn more about pull requests](#)

Cancel Commit changes

15. `git fetch` é usado para buscar todas as atualizações do repositório remoto, mas sem aplicar essas mudanças no seu código ainda. E exibindo os status do diretório.

```
MINGW64:/c/Users/Lucas/Aula-git-github

Lucas@lucas MINGW64 ~/Aula-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:   .gitignore

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

Lucas@lucas MINGW64 ~/Aula-git-github (main)
$ git fetch
remote: Enumerating objects: 11, done.
remote: Counting objects: 100% (11/11), done.
remote: Compressing objects: 100% (5/5), done.
remote: Total 6 (delta 2), reused 0 (delta 0), pack-reused 0 (from 0)
Unpacking objects: 100% (6/6), 1.19 KiB | 45.00 KiB/s, done.
From https://github.com/colombinilucas/Aula-git-github
   e744802..3bdb4cb  main       -> origin/main

Lucas@lucas MINGW64 ~/Aula-git-github (main)
$ |
```

```
MINGW64:/c/Users/Lucas/Aula-git-github

Lucas@lucas MINGW64 ~/Aula-git-github (main)
$ git fetch
remote: Enumerating objects: 1, done.
remote: Counting objects: 100% (1/1), done.
remote: Total 1 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)
Unpacking objects: 100% (1/1), 843 bytes | 210.00 KiB/s, done.
From https://github.com/colombinilucas/Aula-git-github
   3bdb4cb..42f46a2  main       -> origin/main

Lucas@lucas MINGW64 ~/Aula-git-github (main)
$ git status
On branch main
Your branch is behind 'origin/main' by 2 commits, and can be fast-forwarded.
  (use "git pull" to update your local branch)

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:   .gitignore

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

Lucas@lucas MINGW64 ~/Aula-git-github (main)
$
```

16. git pull Baixa as mudanças do repositório remoto e já junta (mergeia) com o seu código local.

```
MINGW64/c/Users/Lucas/Aula-git-github
From https://github.com/colombinilucas/Aula-git-github
 3bdb4cb..42f46a2  main      -> origin/main

Lucas@lucas MINGW64 ~/Aula-git-github (main)
$ git status
On branch main
Your branch is behind 'origin/main' by 2 commits, and can be fast-forwarded.
(use "git pull" to update your local branch)

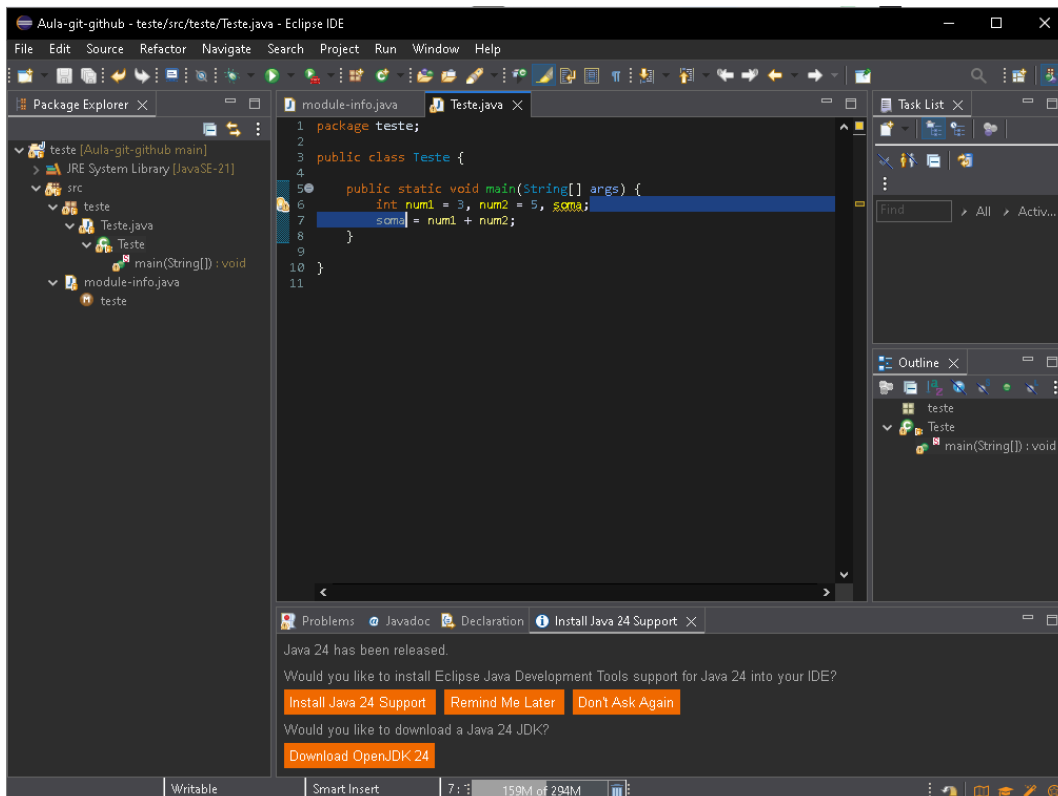
Changes not staged for commit:
  (use "git add <file>..." to update what will be committed)
  (use "git restore <file>..." to discard changes in working directory)
        modified:   .gitignore

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

Lucas@lucas MINGW64 ~/Aula-git-github (main)
$ git pull
Updating e744802..42f46a2
Fast-forward
 teste/src/teste/Teste.java | 4 ++--
 1 file changed, 2 insertions(+), 2 deletions(-)

Lucas@lucas MINGW64 ~/Aula-git-github (main)
$ |
```

17. Visualizando as alterações realizadas pelo git pull na IDE



18. **git pull** Baixa as mudanças do repositório remoto e já junta (mergeia) com o seu código local.

```
MINGW64:/c/Users/Lucas/Aula-git-github
modified:  .gitignore

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

Lucas@lucas MINGW64 ~/Aula-git-github (main)
$ git pull
Updating e744802..42f46a2
Fast-forward
 teste/src/teste/Teste.java | 4 ++--
 1 file changed, 2 insertions(+), 2 deletions(-)

Lucas@lucas MINGW64 ~/Aula-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:   .gitignore

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

Lucas@lucas MINGW64 ~/Aula-git-github (main)
$ |
```

19. Realizando mais uma alteração no repositório remoto, agora printando o resultado soma.

```
Code Blame
Raw Copy Download Edit

1 package teste;
2
3 public class Teste {
4
5     public static void main(String[] args) {
6         int num1 = 3, num2 = 5, soma;
7         soma = num1 + num2;
8         System.out.println(soma);
9     }
10
11 }
```

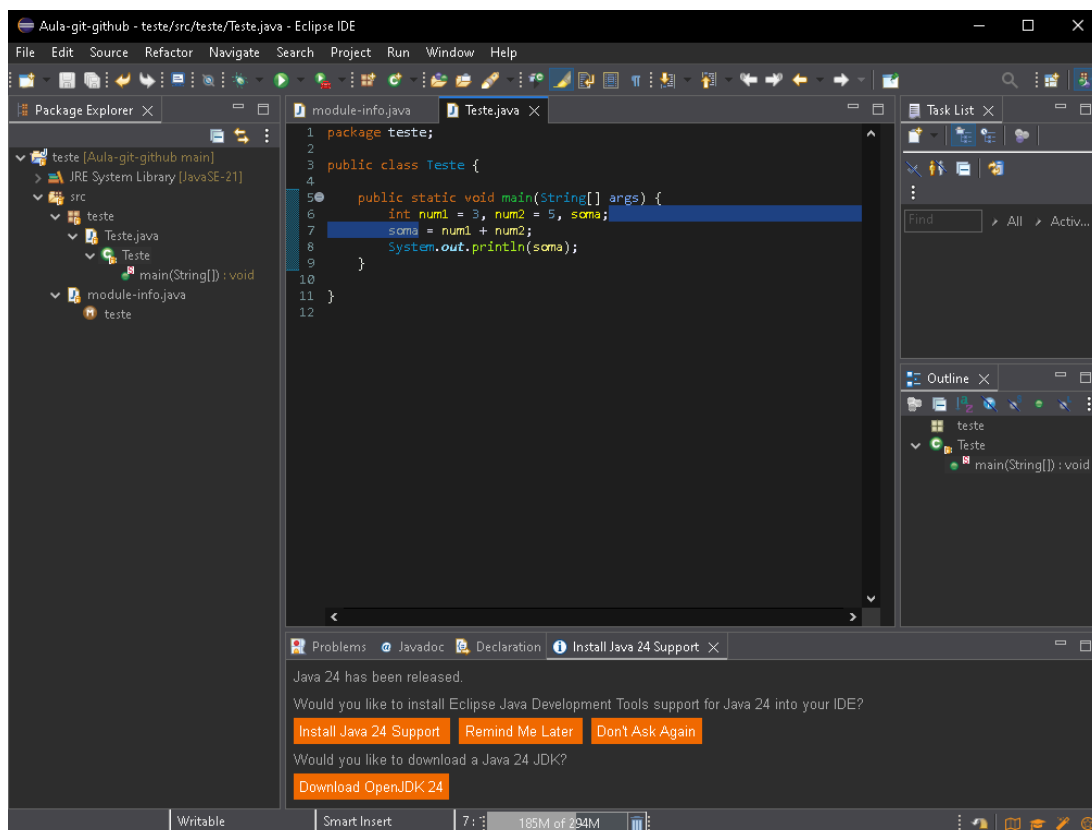
20. git pull Baixando as mudanças do repositório remoto e juntando com o seu código local.

```
MINGW64:/c/Users/Lucas/Aula-git-github

Lucas@lucas MINGW64 ~/Aula-git-github (main)
$ hit pull
bash: hit: command not found

Lucas@lucas MINGW64 ~/Aula-git-github (main)
$ git pull
remote: Enumerating objects: 11, done.
remote: Counting objects: 100% (11/11), done.
remote: Compressing objects: 100% (5/5), done.
remote: Total 6 (delta 3), reused 0 (delta 0), pack-reused 0 (from 0)
Unpacking objects: 100% (6/6), 1.11 KiB | 42.00 KiB/s, done.
From https://github.com/colombinilucas/Aula-git-github
   430b019..73b66ed  main       -> origin/main
Updating 430b019..73b66ed
Fast-forward
 teste/src/teste/Teste.java | 2 +-
 1 file changed, 1 insertion(+), 1 deletion(-)

Lucas@lucas MINGW64 ~/Aula-git-github (main)
$ |
```



```
MINGW64:/c/Users/Lucas/Aula-git-github
remote: Counting objects: 100% (11/11), done.
remote: Compressing objects: 100% (5/5), done.
remote: Total 6 (delta 3), reused 0 (delta 0), pack-reused 0 (from 0)
Unpacking objects: 100% (6/6), 1.11 KiB | 42.00 KiB/s, done.
From https://github.com/colombinilucas/Aula-git-github
   430b019..73b66ed  main       -> origin/main
Updating 430b019..73b66ed
Fast-forward
 teste/src/teste/Teste.java | 2 +-
 1 file changed, 1 insertion(+), 1 deletion(-)

Lucas@lucas MINGW64 ~/Aula-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:   .gitignore

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

Lucas@lucas MINGW64 ~/Aula-git-github (main)
$ |
```