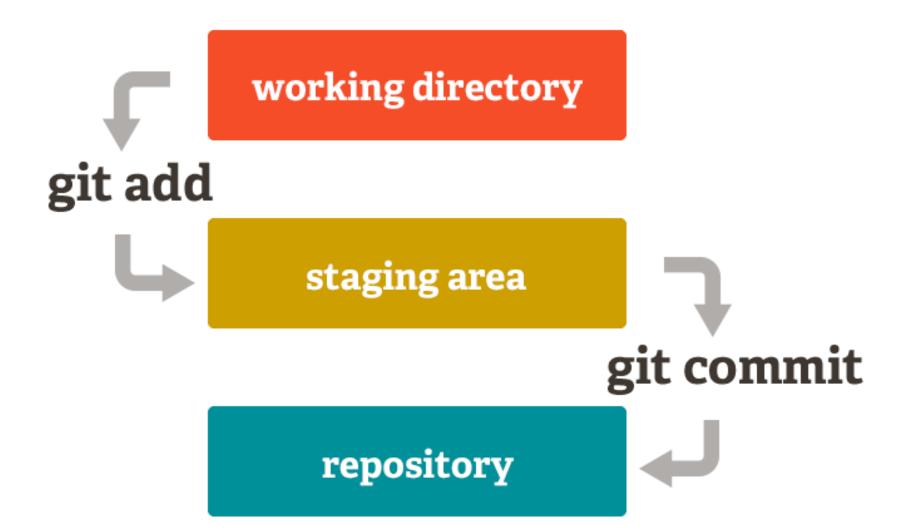


INTRODUÇÃO

- Git (gíria cabeça dura)
- Linus Tovarlds (2005)
- Sistema de controle de versão distribuído
- Registra todas as modificações feitas sobre um artefato ou um conjunto de artefatos ao longo do tempo (máquina do tempo)
- Possibilita voltar no tempo e resgatar uma versão anterior
- Possibilita que várias pessoas trabalhem ao mesmo tempo sobre um mesmo projeto sem conflito



INTRODUÇÃO





INSTALAÇÃO E CONFIGURAÇÃO

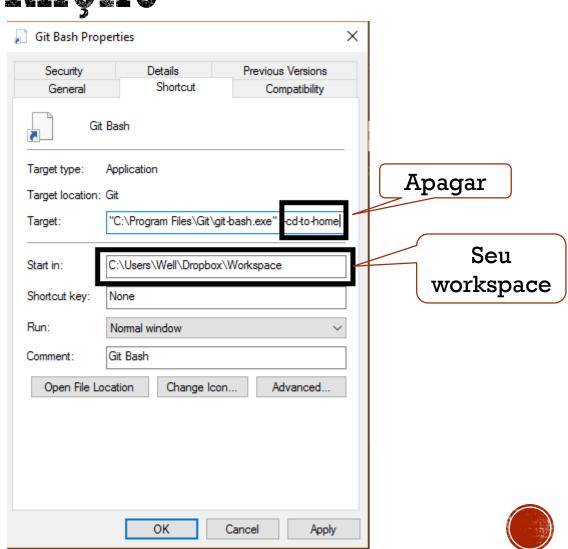
- git-scm.com/downloads
- Iniciar o terminal Git Bash

```
MINGW64:/c/Users/Well
/ell@Wellington MINGW64 ∼
 git config --global user.name "Wellington Moreira"
Well@Wellington MINGW64 ~
$ git config --global user.email "wmoliveira1@gmail.com"
Vell@Wellington MINGW64 ~
```



INSTALAÇÃO E CONFIGURAÇÃO

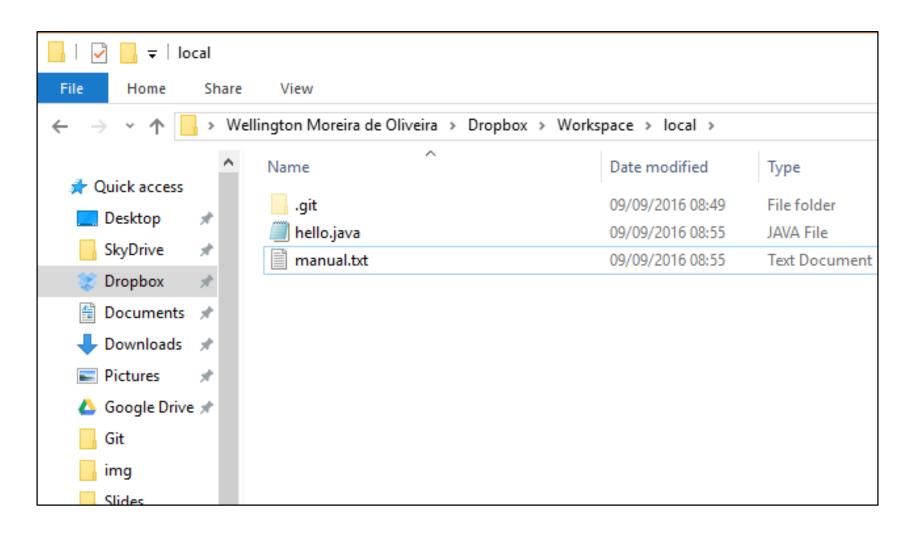
- Criar uma pasta "workspace" e incluir nela um diretório para seu projeto
- Configure o Git Bash para abrir diretamente neste diretório (no Windows clique com o botão direito em cima do atalho Git Bash)
- *cd* entra numa pasta
- cd .. Volta uma pasta
- *clear* limpa os comandos da tela



INICIAR O GIT NUM DIRETÓRIO

```
MINGW64:/c/Users/Well/Dropbox/Workspace/local
                                                                                X
well@Wellington MINGW64 ~/Dropbox/Workspace
$ cd local
well@wellington MINGW64 ~/Dropbox/Workspace/local
$ git init
Initialized empty Git repository in C:/Users/Well/Dropbox/Workspace/local/.git/
Well@Wellington MINGW64 ~/Dropbox/Workspace/local (master)
$ git status
On branch master
Initial commit
nothing to commit (create/copy files and use "git add" to track)
well@wellington MINGW64 ~/Dropbox/Workspace/local (master)
```

INCLUINDO ARQUIVOS NO DIRETÓRIO





INCLUINDO ARQUIVOS NO DIRETÓRIO

```
MINGW64:/c/Users/Well/Dropbox/Workspace/local
                                                                                X
Vell@Wellington MINGW64 ~/Dropbox/Workspace/local (master)
 git status
On branch master
Initial commit
Untracked files:
  (use "git add <file>..." to include in what will be committed)
nothing added to commit but untracked files present (use "git add" to track)
well@Wellington MINGW64 ~/Dropbox/Workspace/local (master)
```

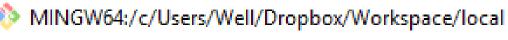


GERENCIANDO ARQUIVOS COM GIT ADD

```
MINGW64:/c/Users/Well/Dropbox/Workspace/local
Well@Wellington MINGW64 ~/Dropbox/Workspace/local (master)
$ git add hello.java
Well@Wellington MINGW64 ~/Dropbox/Workspace/local (master)
$ git add *.txt
Well@Wellington MINGW64 ~/Dropbox/Workspace/local (master)
$ git add .
Well@Wellington MINGW64 ~/Dropbox/Workspace/local (master)
```



GERENCIANDO ARQUIVOS COM GIT ADD



```
well@Wellington MINGW64 ~/Dropbox/Workspace/local (master)
 git status
On branch master
Initial commit
Changes to be committed:
 (use "git rm --cached <file>..." to unstage)
       new file: hello.java
       new file: manual.txt
/ell@Wellington MINGW64 ~/Dropbox/Workspace/local (master)
```



X

DELETANDO ARQUIVOS COM GIT RM



MINGW64:/c/Users/Well/Dropbox/Workspace/local

```
Well@Wellington MINGW64 ~/Dropbox/Workspace/local (master)
$ git status
On branch master
Changes not staged for commit:
  (use "git add/rm <file>..." to update what will be committed)
  (use "git checkout -- <file>..." to discard changes in working directory)
        modified: hello.java
deleted: manual.txt
no changes added to commit (use "git add" and/or "git commit -a")
well@Wellington MINGW64 ~/Dropbox/Workspace/local (master)
 git rm manual.txt
rm 'manual.txt'
well@Wellington MINGW64 ~/Dropbox/Workspace/local (master)
```



X

CONFIRMANDO E ENVIANDO AS MUDANÇAS PARA O REPOSITÓRIO - COMMIT

```
MINGW64:/c/Users/Well/Dropbox/Workspace/local
Well@Wellington MINGW64 ~/Dropbox/Workspace/local (master)
$ git commit -m "first commit"
[master (root-commit) ab46082] first commit
2 files changed, 0 insertions(+), 0 deletions(-)
create mode 100644 hello.java
 create mode 100644 manual.txt
 well@Wellington MINGW64 ~/Dropbox/Workspace/local (master)
$ git status
On branch master
nothing to commit, working tree clean
 well@wellington MINGW64 ~/Dropbox/Workspace/local (master)
```



IGNORANDO ARQUIVOS NO GIT

- Criar um arquivo .gitignore
- Criar um arquivo teste.txt
- Incluir o teste.txt no arquivo .gitignore
- Pode ser colocado coringas e pastas inteiras
- Todos arquivos incluídos neste arquivo serão ignorados pelo GIT (não irá aparecer como not staged com o git status)



COMITANDO DIRETO SEM ADD

MINGW64:/c/Users/Well/Dropbox/Workspace/local

```
Well@Wellington MINGW64 ~/Dropbox/Workspace/local (master)
 git status
On branch master
Changes not staged for commit:
 (use "git add <file>..." to update what will be committed)
  (use "git checkout -- <file>..." to discard changes in working directory)
no changes added to commit (use "git add" and/or "git commit -a")
Well@Wellington MINGW64 ~/Dropbox/Workspace/local (master)
§ git commit -a -m "Alteração no arquivo manual"
[master 968f0b0] Alteraçao no arquivo manual
1 file changed, 1 insertion(+)
well@Wellington MINGW64 ~/Dropbox/Workspace/local (master)
```

X

VISUALIZANDO ALTERAÇÕES (NOT STAGED) - DIFF

Retirado vermelho

Adicionado verde

```
MINGW64:/c/Users/Well/Dropbox/Workspace/local
                                                                                              X
Well@Wellington MINGW64 ~/Dropbox/Workspace/local (master)
  git diff
diff --git a/manual.txt b/manual.txt
index 4734ad1..8f8bc26 100644
    a/manual.txt
+++ b/manual.txt
aa -1 +1 aa
+Uma adi<mark><E7><E3></mark>o diferente com uma remo<mark><E7><E3></mark>o anterior do Ola
Well@Wellington MINGW64 ~/Dropbox/Workspace/local (master)
```



VISUALIZANDO ALTERAÇÕES (STAGED) - DIFF

```
MINGW64:/c/Users/Well/Dropbox/Workspace/local
 Well@Wellington MINGW64 ~/Dropbox/Workspace/local (master)
$ git diff --staged
diff --git a/manual.txt b/manual.txt
index 4734ad1..8f8bc26 100644
--- a/manual.txt
+++ b/manual.txt
aa -1 +1 aa
+Uma adi<mark><E7><E3></mark>o diferente com uma remo<mark><E7><E3></mark>o anterior do Ola
Well@Wellington MINGW64 ~/Dropbox/Workspace/local (master)
```



VISUALIZANDO LOG DE COMMITS

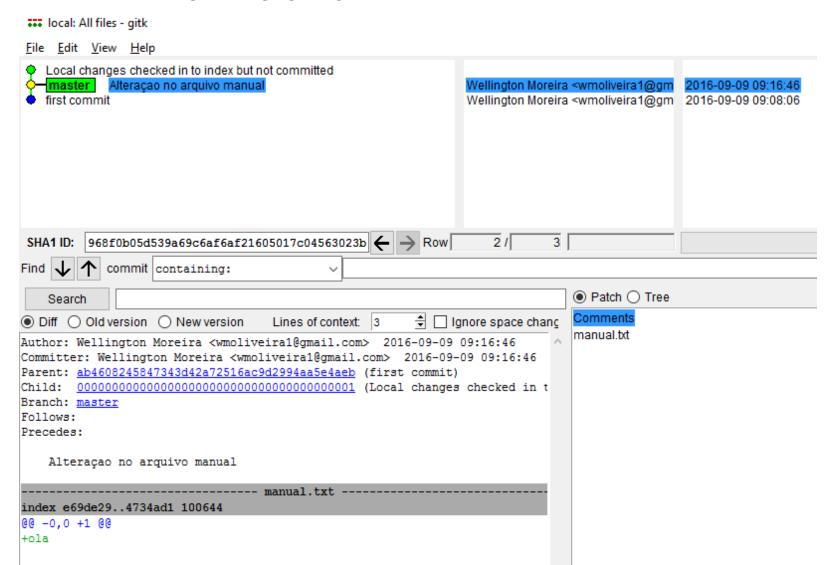
```
MINGW64:/c/Users/Well/Dropbox/Workspace/local
                                                                               X
well@Wellington MINGW64 ~/Dropbox/Workspace/local (master)
$ git log
commit 968f0b05d539a69c6af6af21605017c04563023b
Author: Wellington Moreira <wmoliveira1@gmail.com>
Date: Fri Sep 9 09:16:46 2016 -0300
    Alteração no arquivo manual
commit ab4608245847343d42a72516ac9d2994aa5e4aeb
Author: Wellington Moreira <wmoliveira1@gmail.com>
Date: Fri Sep 9 09:08:06 2016 -0300
   first commit
Well@Wellington MINGW64 ~/Dropbox/Workspace/local (master)
```

VISUALIZANDO LOG DE COMMITS

- Paginação de log trazendo os mais novos primeiros git log -p
- Trazer somente o último alterado
 git log -p l
- Visualizando numa interface gráfica gitk
- Visualizando só o hash dos commts com a mensagem git log -pretty=oneline



VISUALIZANDO LOG GITK





DESFAZENDO AÇÕES

 Desfazer uma alteração num arquivo não "stagiado" (volta para o último commit)

git checkout -- <nome do arquivo>

Tirar o arquivo da stage area

git reset Head <nome do arquivo>

- Editar/Reverter um commit errado sem criar outro
 - Passos: > alterar o arquivo > git add arquivo >

git commit --amend -m "Minha msg"



- Nome simbólico para uma revisão
- Utilizadas para fazer marcações de um estado do sistema
- Normalmente utilizadas para rotular versões
- Atalho para reverter o status para uma tag (hora e data específica)
- Consulta mais rápida informando a tag
- Não mudam com novos commits (como é o caso dos branches)
- Criando uma tag anotada para o commit atual e anterior git tag -a v1.0 -m "Versão 1.0"
 git tag -a v1.0 hash -m "Versão 0.0"



```
MINGW64:/c/Users/Well/Dropbox/Workspace/local
 well@wellington MINGW64 ~/Dropbox/Workspace/local (master)
   git log --pretty=oneline
298b665dc8ffa7c86ee2e553cef073723f00384 Retirada de arquivo
cbc8b160ec0e27f734d19a47e299692efa566289 alternado novamente
968f0b05d539a69c6af6af21605017c04563023b Alteração no arquivo manual
ab4608245847343d42a72516ac9d2994aa5e4aeb first commit
Well@Wellington MINGW64 ~/Dropbox/Workspace/local (master)
$ git tag -a v0.0 ab4608245847343d42a72516ac9d2994aa5e4aeb -m "Primeira versão"
Well@Wellington MINGW64 ~/Dropbox/Workspace/local (master)
$ git tag
v0.0
| | Well@Wellington MINGW64 ~/Dropbox/Workspace/local (master)
| $ git tag -a v1.0 -m "Versão 1.0 Retirada de arquivo"
 Well@Wellington MINGW64 ~/Dropbox/Workspace/local (master)
$ git tag
v0.0
v1.0
 Well@Wellington MINGW64 ~/Dropbox/Workspace/local (master)
```



- Mostrar a taggit taggit show v1.0
- Voltar para uma versão anterior git checkout v0.0 git checkout máster
- Deletar uma taggit tag -d v0.0



```
MINGW64:/c/Users/Well/Dropbox/Workspace/local
                                                                               ×
 /ell@Wellington MINGW64 ~/Dropbox/Workspace/local (master)
$ git show v0.0
 ag v0.0
Tagger: Wellington Moreira <wmoliveira1@gmail.com>
Date: Wed Sep 28 11:18:29 2016 -0300
Primeira versão
commit ab4608245847343d42a72516ac9d2994aa5e4aeb
Author: Wellington Moreira <wmoliveira1@gmail.com>
Date: Fri Sep 9 09:08:06 2016 -0300
    first commit
diff --git a/hello.java b/hello.java
new file mode 100644
index 0000000..e69de29
diff --git a/manual.txt b/manual.txt
new file mode 100644
index 0000000..e69de29
well@Wellington MINGW64 ~/Dropbox/Workspace/local (master)
$ git checkout v0.0
Note: checking out 'v0.0'.
You are in 'detached HEAD' state. You can look around, make experimental
changes and commit them, and you can discard any commits you make in this
state without impacting any branches by performing another checkout.
If you want to create a new branch to retain commits you create, you may
do so (now or later) by using -b with the checkout command again. Example:
  git checkout -b <new-branch-name>
HEAD is now at ab46082... first commit
 /ell@Wellington MINGW64 ~/Dropbox/Workspace/local ((v0.0))
```



BRANCH

- Nome simbólico para linhas de desenvolvimento
- Ramificação/Segmentação do sistema
- Criação, checkout, e deleção git branch Teste git checkout Teste git checkout -b Teste git branch -d Teste
- Isolamento de linhas/frentes diferentes de desenvolvimento
- Inclusão de novas funcionalidades
- Correção do sistema
- Ambientes de desenvolvimento (Teste, Homologação Produção)



BRANCH

```
MINGW64:/c/Users/Well/Dropbox/Workspace/local
Well@Wellington MINGW64 ~/Dropbox/Workspace/local ((v0.0))
$ git branch Teste
Well@Wellington MINGW64 ~/Dropbox/Workspace/local ((v0.0))
 git checkout Teste
Switched to branch 'Teste'
Well@Wellington MINGW64 ~/Dropbox/Workspace/local (Teste)
$ git checkout master
Switched to branch 'master'
Well@Wellington MINGW64 ~/Dropbox/Workspace/local (master)
$ git checkout -b Teste
fatal: A branch named 'Teste' already exists.
Well@Wellington MINGW64 ~/Dropbox/Workspace/local (master)
```



MERGE

- Mesclagem de branches
- Você deve estar no ambiente destino. Ex.: master git merge Teste
- Conflito entre os arquivos
 - Fazer um merge manual (alterar o arquivo)
 - Comitar



MERGE

```
MINGW64:/c/Users/Well/Dropbox/Workspace/local
 1 file changed, 1 insertion(+), 1 deletion(-)
Well@Wellington MINGW64 ~/Dropbox/Workspace/local (master)
$ git checkout Teste
Switched to branch 'Teste'
Well@Wellington MINGW64 ~/Dropbox/Workspace/local (Teste)
$ git commit -a -m 'Testando merge no branch Teste'
[Teste 9565f1c] Testando merge no branch Teste
1 file changed, 2 insertions(+)
Well@Wellington MINGW64 ~/Dropbox/Workspace/local (Teste)
$ git checkout master
Switched to branch 'master'
Well@Wellington MINGW64 ~/Dropbox/Workspace/local (master)
$ git merge Teste
Auto-merging hello.java
CONFLICT (content): Merge conflict in hello.java
Automatic merge failed; fix conflicts and then commit the result.
Well@Wellington MINGW64 ~/Dropbox/Workspace/local (master MERGING)
$ git commit -a -m 'Merge manual'
[master 7557392] Merge manual
Well@Wellington MINGW64 ~/Dropbox/Workspace/local (master)
```

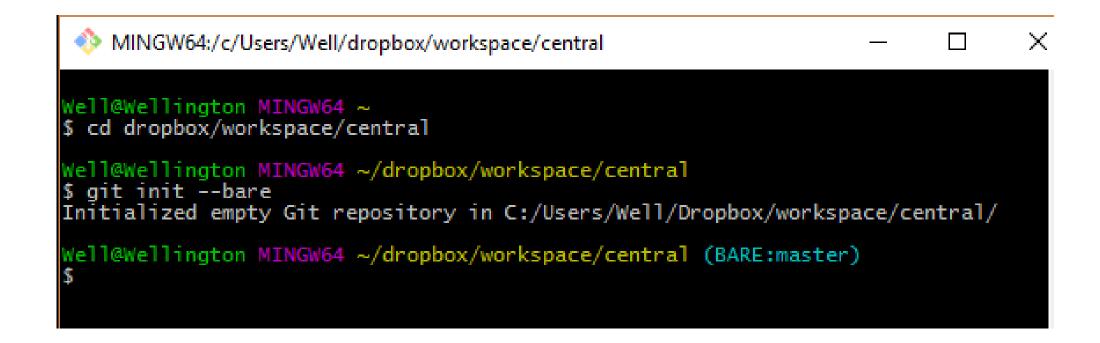


REPOSITÓRIO CENTRAL

- Criar uma pasta e compartilhar na rede
- Iniciar um repositório Git central e clonar para o repositório local
- cd diretórioRepositorioCentral
- git init --bare
- cd diretórioRepositorioLocal
- git clone diretórioLocal novaPasta



INICIALIZANDO O REPOSITÓRIO CENTRAL





CLONANDO O REPOSITÓRIO CENTRAL



ADICIONANDO NOVOS ARQUIVOS (LOCAL)

```
MINGW64:/c/Users/Well/dropbox/workspace/Projeto1
                                                                                         ×
well@Wellington MINGW64 ~/dropbox/workspace
$ cd Projeto1
well@wellington MINGW64 ~/dropbox/workspace/Projeto1 (master)
$ git status
On branch master
Initial commit
Untracked files:
  (use "git add <file>..." to include in what will be committed)
nothing added to commit but untracked files present (use "git add" to track)
well@wellington MINGW64 ~/dropbox/workspace/Projeto1 (master)
$ git add .
well@Wellington MINGW64 ~/dropbox/workspace/Projeto1 (master)
$ git commit -m "Commit inicial"
[master (root-commit) aee3df5] Commit inicial
3 files changed, 0 insertions(+), 0 deletions(-)
 create mode 100644 Cadastro.java
 create mode 100644 Conexao.java
 create mode 100644 Principal.java
```

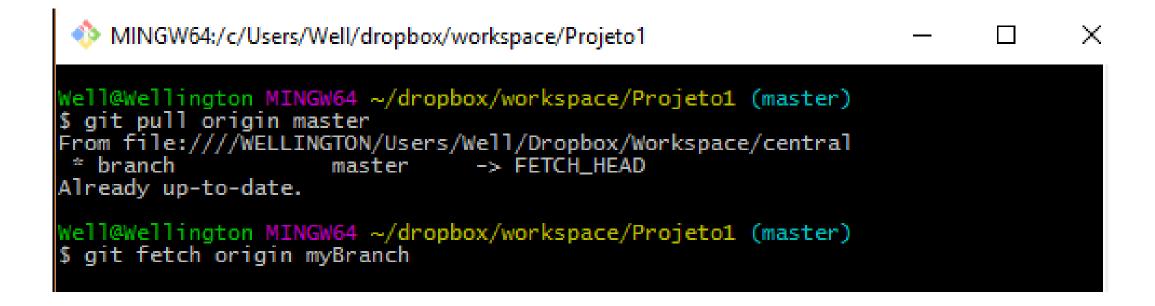


ADICIONANDO NO REPOSITÓRIO CENTRAL (PUSH)

```
MINGW64:/c/Users/Well/dropbox/workspace/Projeto1
 Well@Wellington MINGW64 ~/dropbox/workspace/Projeto1 (master)
   git remote
oragan
 Well@Wellington MINGW64 ~/dropbox/workspace/Projeto1 (master)
$ git push origin master
Counting objects: 3, done.
Delta compression using up to 4 threads.
Compressing objects: 100% (2/2), done.
Writing objects: 100% (3/3), 248 bytes | 0 bytes/s, done.
Total 3 (delta 0), reused 0 (delta 0)
To file:////WELLINGTON/Users/Well/Dropbox/Workspace/central
     [new branch] master -> master
```



ATUALIZANDO O PROJETO LOCAL (PULL E FETCH)





CONECTAR COM GITHUB

- Verificar o caminho da chave SSH gerada no Git Bash
- Abrir o arquivo .pub com o bloco de notas
- Copiar todo o conteúdo do arquivo
- Levar para sua conta do GitHub
- Account Settings > SSH Keys > Add Key
- A partir daí é possível ter acesso ao Git Hub a partir do Git Bash local



CONECTAR COM GITHUB (GERAR CHAVE SSH)

```
MINGW64:/c/Users/Well/dropbox/workspace/Projeto1
Well@Wellington MINGW64 ~/dropbox/workspace/Projeto1 (master)
$ ssh-keygen
Generating public/private rsa key pair.
Enter file in which to save the key (/c/Users/Well/.ssh/id_rsa):
Created directory '/c/Users/Well/.ssh'.
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /c/Users/Well/.ssh/id_rsa.
Your public key has been saved in /c/Users/Well/.ssh/id_rsa.pub.
The key fingerprint is:
SHA256:ixwU4tR2/hT+bfeP+Occw3gDZuOZsNpPjw9cNJbJpGQ Well@Wellington
The key's randomart image is:
+---[RSA 2048]----+
                 Eo.
     0 .0.. . 0+=00
      ...0 . .00=*+
         . o o..X.
         . So o o.o=
                    +0
                  00*
 ----[SHA256]----+
 Well@Wellington MINGW64 ~/dropbox/workspace/Projeto1 (master)
```

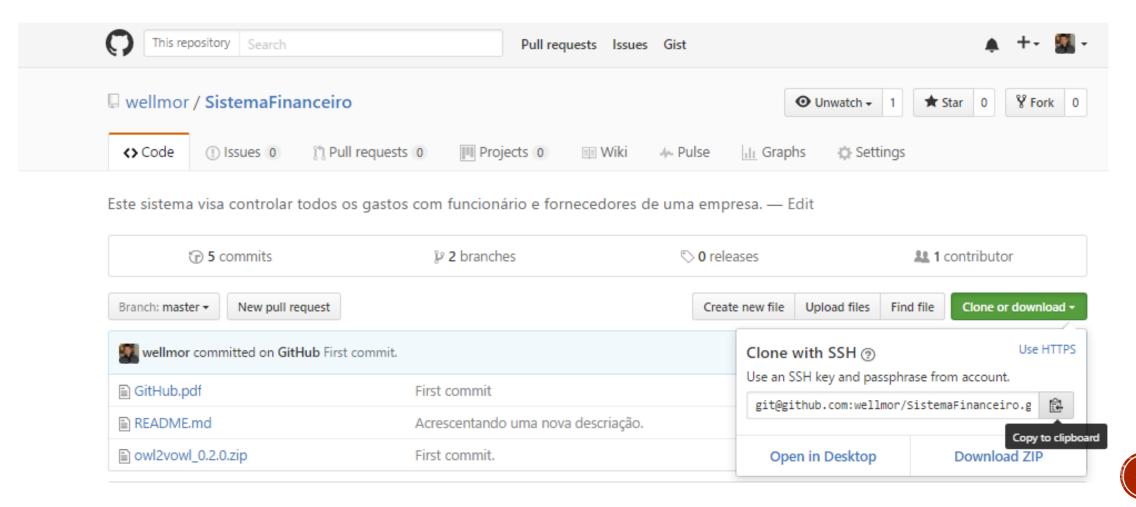


CONECTAR COM O GITHUB (ADICIONAR CHAVE)

ersonal settings	SSH keys New SSH
rofile	There are no SSH keys with access to your account.
ccount	Title
mails	Minha Chave
lotifications	Key
illing	ssh-rsa AAAAB3NzaC1yc2EAAAADAQABAAABAQDNeHppC/+9tlqHSs2HWPj6PhG07wh3xsBu32GUWuHDgNJbVw6yCIr Y6NFTSSXnVRUkO/UI2L0H2POGSv2g4478iKVyMZdWbkdYl0963ArCFKpfJ3NX9H6BakHXY1ydWFIVe8atRDNB9B /RYO915528RIWEoKnvFFFPmP/+wK2YaEuFHcMUZOWGrKicQJ6GKjdkRYIytWjWnkR+4GCx1/koGJYoTMKL1xrtt8v XqVQKqwa5UdkcMaVTFB9iz2okrJsUve/Z3OfOyWe7/aarDz7t87tkDoE3DlFJ5n7g/u6t8QTJIUxqoufSzstwfGLKaGu/ 8WjpID9GXxbcnj2Xabf Well@Wellington
SH and GPG keys	
ecurity	
locked users	
epositories	
rganizations	
aved replies	Add SSH key



CLONAR REPOSITÓRIO SSH GITHUB (COPIAR ENDEREÇO SSH OU HTTP)



CLONAR REPOSITÓRIO SSH GITHUB (COLAR ENDEREÇO SSH OU HTTP)

```
MINGW64:/c/Users/Well/dropbox/workspace
                                                                                                 X
Well@Wellington MINGW64 ~/dropbox/workspace
$ git clone git@github.com:wellmor/SistemaFinanceiro.git SistemaFinanceiro
Cloning into 'SistemaFinanceiro'...
ssh: connect to host github.com port 22: Connection refused
fatal: Could not read from remote repository.
Please make sure you have the correct access rights
and the repository exists.
Well@Wellington MINGW64 ~/dropbox/workspace
$ git clone https://github.com/wellmor/SistemaFinanceiro.git SistemaFinanceiro
Cloning into 'SistemaFinanceiro'...
remote: Counting objects: 13, done.
remote: Total 13 (delta 0), reused 0 (delta 0), pack-reused 13
Unpacking objects: 100% (13/13), done.
```



SINCRONIZANDO COM O REPOSITÓRIO CENTRAL (GIT HUB)

- Mesmo comando para adicionar ao repositório central de rede
 - git push origin master
- Mesmo comando para atualizar com repositório central de rede
 - git pull origin master



SINCRONIZANDO COM O REPOSITÓRIO CENTRAL

(GIT HUB)

```
MINGW64:/c/Users/Well/dropbox/workspace/SistemaFinanceiro
Well@Wellington MINGW64 ~/dropbox/workspace
$ cd SistemaFinanceiro
Well@Wellington MINGW64 ~/dropbox/workspace/SistemaFinanceiro (master)
$ git status
On branch master
Your branch is up-to-date with 'origin/master'.
Untracked files:
  (use "git add <file>..." to include in what will be committed)
nothing added to commit but untracked files present (use "git add" to track)
Well@Wellington MINGW64 ~/dropbox/workspace/SistemaFinanceiro (master)
$ git add .
Well@Wellington MINGW64 ~/dropbox/workspace/SistemaFinanceiro (master)
$ git commit -m "Adicionando o manual"
[master 680700b] Adicionando o manual
1 file changed, 0 insertions(+), 0 deletions(-)
 create mode 100644 manual.txt
Well@Wellington MINGW64 ~/dropbox/workspace/SistemaFinanceiro (master)
$ git push origin master
Counting objects: 3, done.
Delta compression using up to 4 threads.
Compressing objects: 100% (2/2), done.
Writing objects: 100% (3/3), 283 bytes | 0 bytes/s, done.
Total 3 (delta 1), reused 0 (delta 0)
remote: Resolving deltas: 100% (1/1), completed with 1 local objects.
To https://github.com/wellmor/SistemaFinanceiro.git
   f0cc3a9..680700b master -> master
```



EXERCÍCIO 1

- Configure o seu nome e email para o GIT
- Crie seu workspace e inclua uma pasta com de um projeto
- Faça o GIT gerenciar esta pasta
- Crie dois arquivos .java e dois .txt
- Verifique o status e inclua os arquivos no GIT
- Verifique novamente o status e confirme os arquivos para o repositório
- Cria mais um arquivo txt e retire do gerência do GIT
- Edite um commit já realizado



EXERCÍCIO 2

- Crie duas tags para o seu sistema indicando uma versão inicial e a atual
- Crie um branch com o seu nome e faça o checkout do branch master para o o novo branch
- Crie novos arquivos e faça o commit para este novo branch
- Faça o merge deste novo branch com o master
- Modifique um mesmo arquivo nos dois braches (master e o outro)
- Teste fazer o merge automático e faça o merge manual em seguida



EXERCÍCIO 3

- Crie uma nova pasta para ser o seu repositório central
- Clone o repositório central para um projeto local
- Adicione novos arquivos no projeto local 1
- Faça um push detes arquivos
- Clone o respositório central para um segundo projeto local
- Altere um arquivo do segundo projeto
- Faça um pull para o primeiro projeto para sincroniza-lo com o central

