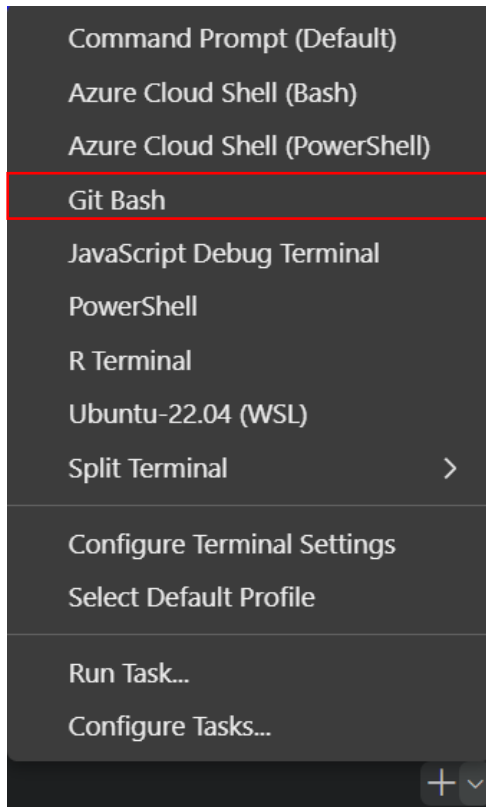


Git – Getting Started

Getting Started

Switch to Git Bash

- Switch via Dropdown



Getting Started

Version

- `git --version`

```
bertg@BERT2022 MINGW64 /c/Projects/DataAnalyst_Ultimate (main)  
$ git --version  
git version 2.44.0.windows.1
```

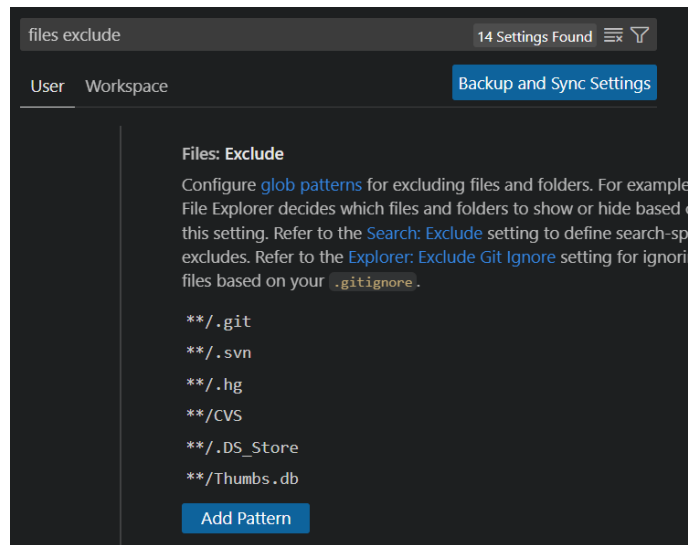
Getting Started

Initialisation

- **git init** - initialize empty git repository

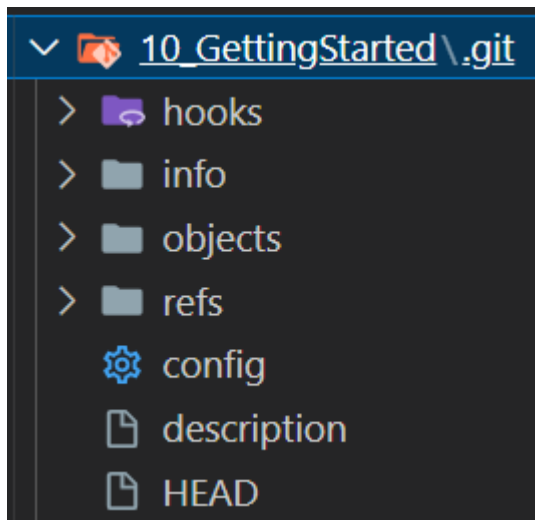
```
bertg@BERT2022 MINGW64 /c/Projects/DataAnalyst_Ultimate/20_Git/10_GettingStarted (main)
$ git init
Initialized empty Git repository in C:/Projects/DataAnalyst_Ultimate/20_Git/10_GettingStarted/.git/
```

- creates subfolder .git
- folder by default invisible → make visible by
 - File – Preferences – Setting
 - search for „files exclude“
 - remove entry for ****/.git**



Getting Started

subfolder .git



Getting Started

Configuration

- show configuration...`git config --list` or `git config -l`
- set user name...`git config --global user.name "Berti Gollnick"`
- set email...`git config --global user.email "bert.gollnick@posteo.net"`

--global...affects system
wide config

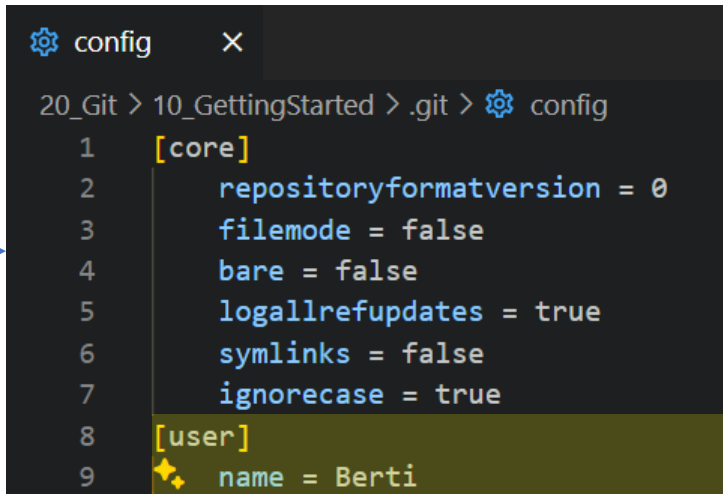
--local...only this
project

```
bertg@BERT2022 MINGW64 /c/Projects/DataAnalyst_Ultimate/20_Git/10_GettingStarted (master)
$ git config --list
diff.astextplain.textconv=astextplain
filter.lfs.clean=git-lfs clean -- %f
filter.lfs.smudge=git-lfs smudge -- %f
filter.lfs.process=git-lfs filter-process
filter.lfs.required=true
http.sslbackend=openssl
http.sslcainfo=C:/Program Files/Git/mingw64/etc/ssl/certs/ca-bundle.crt
core.autocrlf=true
core.fscache=true
core.symlinks=false
pull.rebase=false
credential.helper=manager
credential.https://dev.azure.com.usehttppath=true
init.defaultbranch=master
user.email=bert.gollnick@posteo.net
user.name=bert.gollnick
```

Getting Started

.gitconfig

```
bertg@BERT2022 MINGW64 /c/Projects/DataAnalyst_Ultimate/20_Git/10_GettingStarted (master)  
$ git config --local user.name "Berti"
```



config X

20_Git > 10_GettingStarted > .git > config

```
1 [core]  
2     repositoryformatversion = 0  
3     filemode = false  
4     bare = false  
5     logallrefupdates = true  
6     symlinks = false  
7     ignorecase = true  
8 [user]  
9     ✨ name = Berti
```

.git/config

Getting Started

Git Status

- git status...practical info on current branch, commits, or nothing to commit

```
$ git status
On branch master

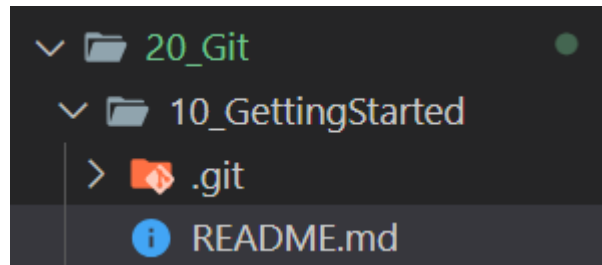
No commits yet

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


Getting Started

New File

- create a file „README.md“ in working folder (10_GettingStarted)



```
bertg@BERT2022 MINGW64 /c/Projects/DataAnalyst_Ultimate/20_Git/10_GettingStarted (master)
$ dir
README.md
```

```
$ git status
On branch master

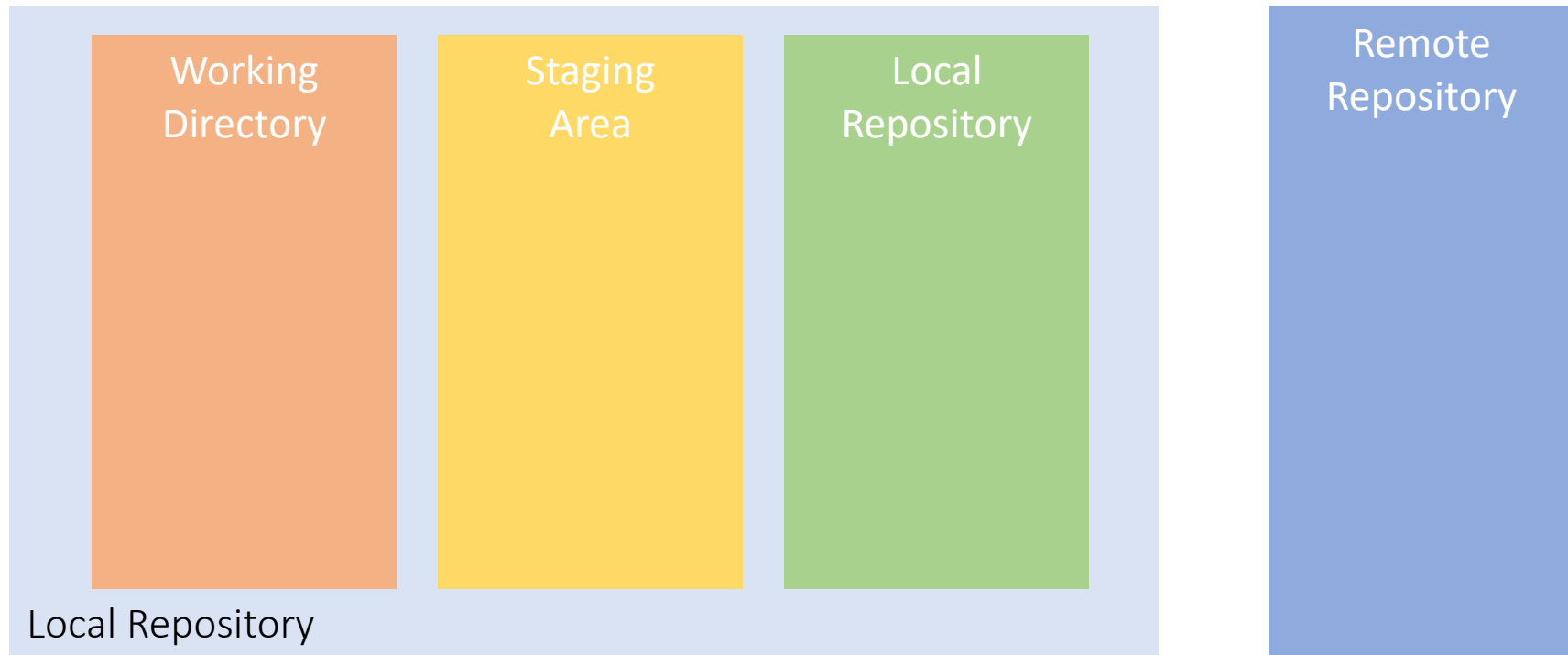
No commits yet

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

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

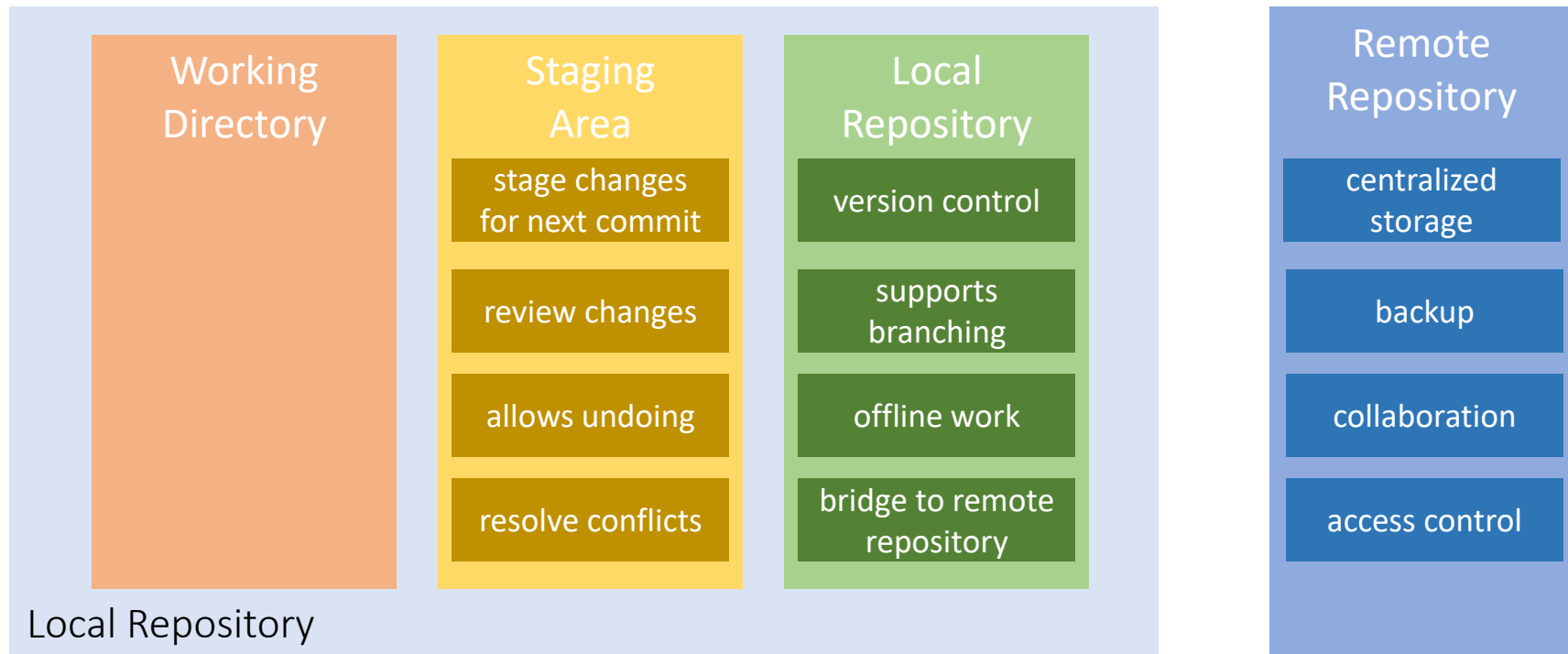
Getting Started

Repositories



Getting Started

Repositories: Purpose



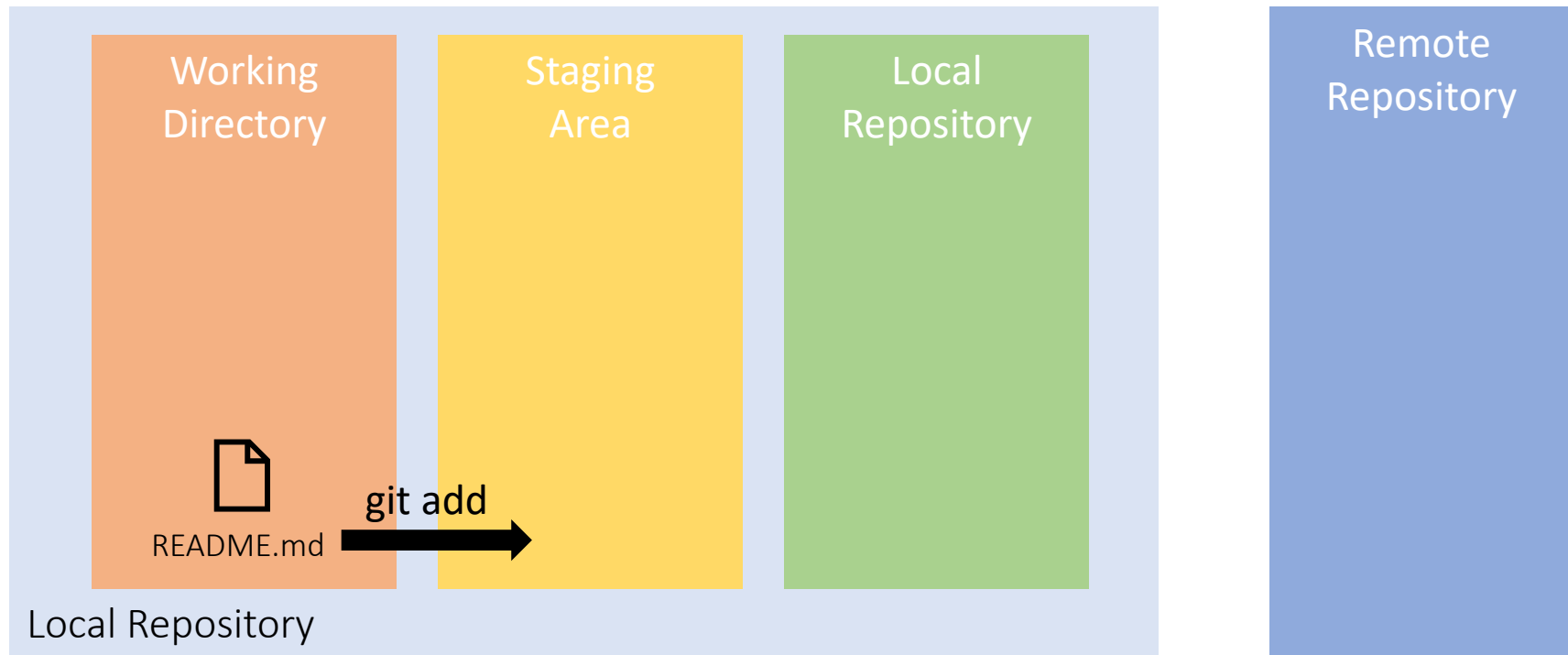
GitHub



Gitlab

Getting Started

Staging



Getting Started

git add

```
bertg@BERT2022 MINGW64 /c/Projects/DataAnalyst_Ultimate/20_Git/10_GettingStarted (master)
$ git add README.md

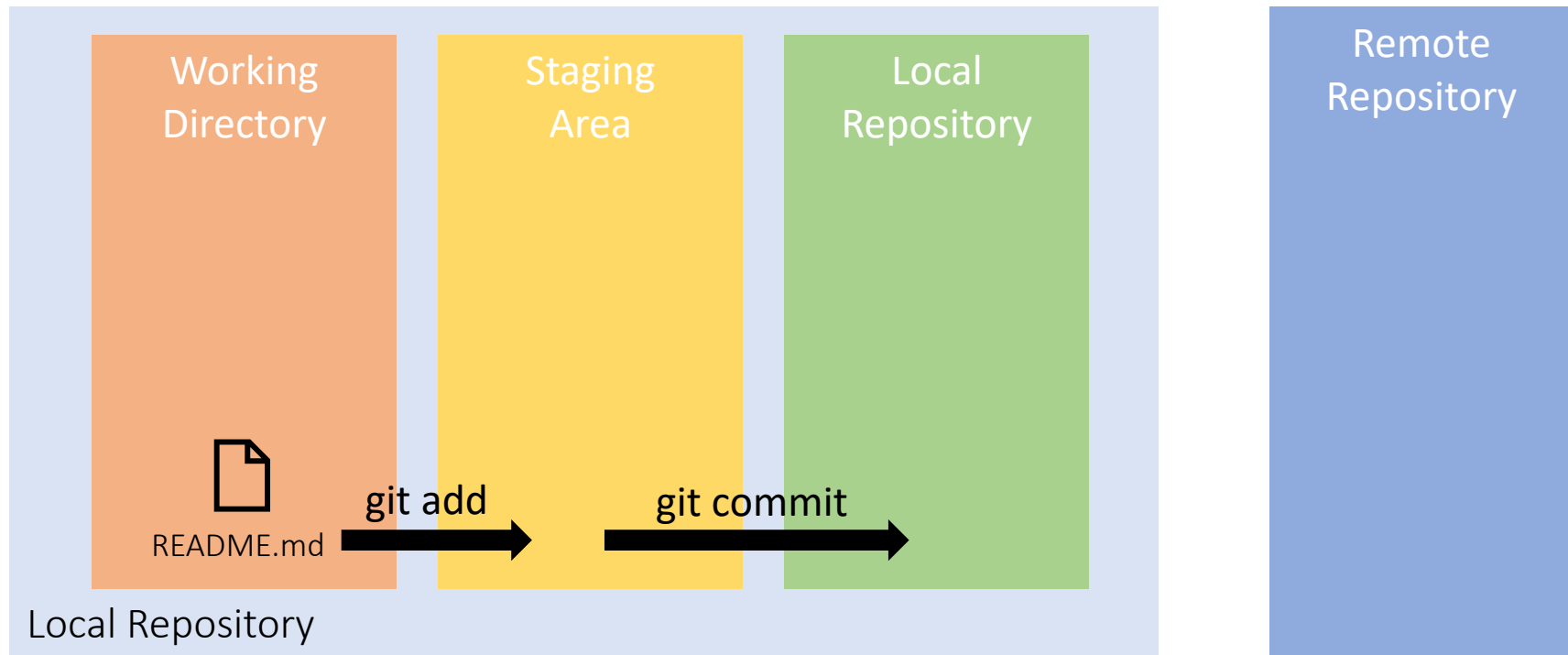
bertg@BERT2022 MINGW64 /c/Projects/DataAnalyst_Ultimate/20_Git/10_GettingStarted (master)
$ git status
On branch master

No commits yet

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

Getting Started

Committing



Getting Started

Committing

- `git commit -m „commit message“`
- try to come up with useful message

```
bertg@BERT2022 MINGW64 /c/Projects/DataAnalyst_Ultimate/20_Git/10_GettingStarted (master)
$ git commit -m "README added"
[master (root-commit) 3b3060f] README added
1 file changed, 0 insertions(+), 0 deletions(-)
create mode 100644 README.md

bertg@BERT2022 MINGW64 /c/Projects/DataAnalyst_Ultimate/20_Git/10_GettingStarted (master)
$ git status
On branch master
nothing to commit, working tree clean
```


Getting Started

Log

- `git log`...provides history of previous commits

```
bertg@BERT2022 MINGW64 /c/Projects/DataAnalyst_Ultimate/20_Git/10_GettingStarted (master)
$ git log
commit 3b3060f0edcfaec513f7f8ff70b19a022504a065 (HEAD -> master)
Author: Berti <bert.gollnick@posteo.net>
Date:   Wed Apr 24 08:38:00 2024 +0200

    README added
```



Git ID

- every commit has unique ID (SHA-1 checksum based on changes)
- also author and date shown

Getting Started

git diff

- typical workflow:
 - file adapted
 - needs to be added to repo
- **git diff**...shows content changes
- only relevant parts are shown
- - indicates removed line
- + indicates added line

```
20_Git > 10_GettingStarted > ⓘ README.md > abc # Getting Started
1  # Getting Started
2
3
```

```
bertg@BERT2022 MINGW64 /c/Projects/DataAnalyst_Ultimate/20_Git/10_GettingStarted (master)
$ git diff
diff --git a/README.md b/README.md
index e69de29..31dd91e 100644
--- a/README.md
+++ b/README.md
@@ -0,0 +1,2 @@
+# Getting Started
+
```

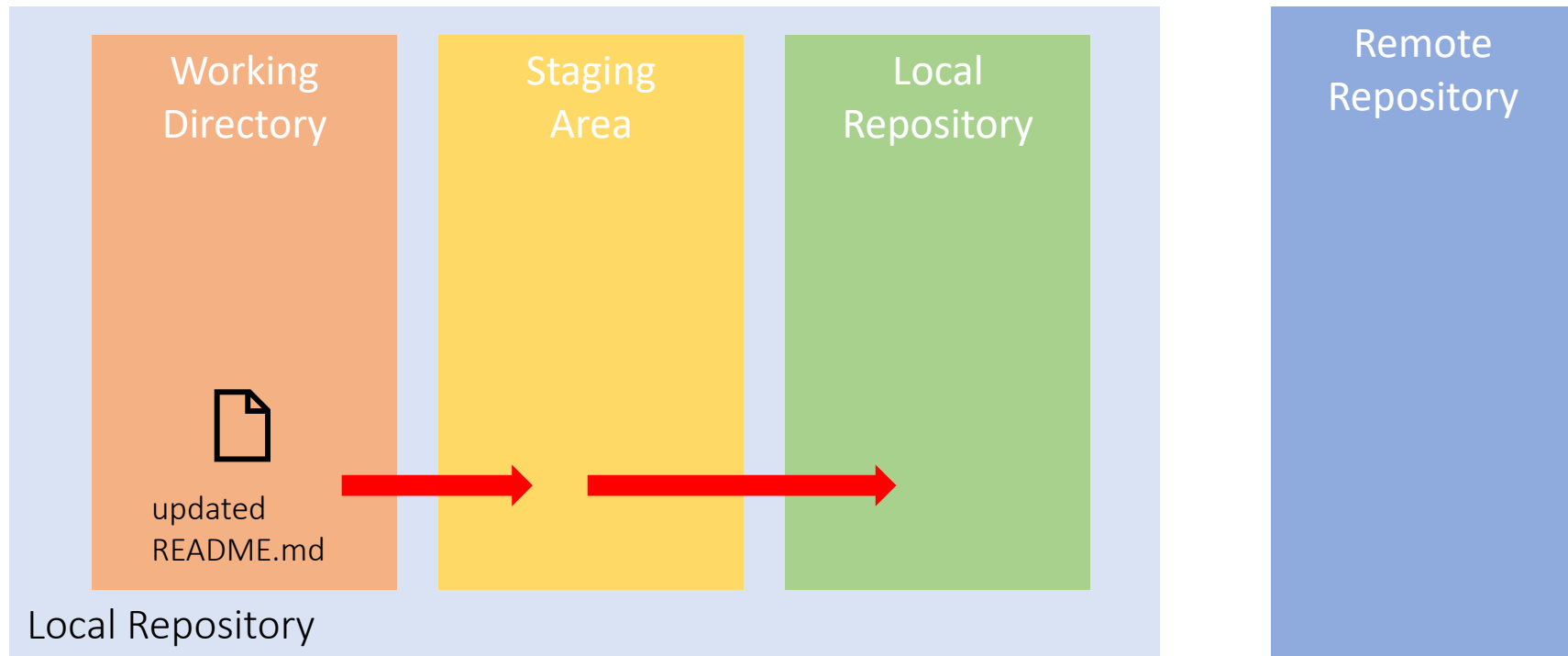
Getting Started

Exercise

- add modified file to staging area
- create another commit to the local repository

Getting Started

Exercise



- TASK: add modified file to staging area
- create another commit to the local repository, check previous commits

Getting Started

Solution

```
bertg@BERT2022 MINGW64 /c/Projects/DataAnalyst_Ultimate/20_Git/10_GettingStarted (master)
$ git status
On branch master
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.md

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

bertg@BERT2022 MINGW64 /c/Projects/DataAnalyst_Ultimate/20_Git/10_GettingStarted (master)
$ git add README.md

bertg@BERT2022 MINGW64 /c/Projects/DataAnalyst_Ultimate/20_Git/10_GettingStarted (master)
$ git commit -m "README adapted and headline included"
[master 72d59f7] README adapted and headline included
1 file changed, 2 insertions(+)
```

Getting Started

Solution

```
bertg@BERT2022 MINGW64 /c/Projects/DataAnalyst_Ultimate/20_Git/10_GettingStarted (master)
$ git log
commit 72d59f79f60150b1acec7544e5a94a472ec852b7 (HEAD -> master)
Author: Berti <bert.gollnick@posteo.net>
Date:   Wed Apr 24 08:48:49 2024 +0200

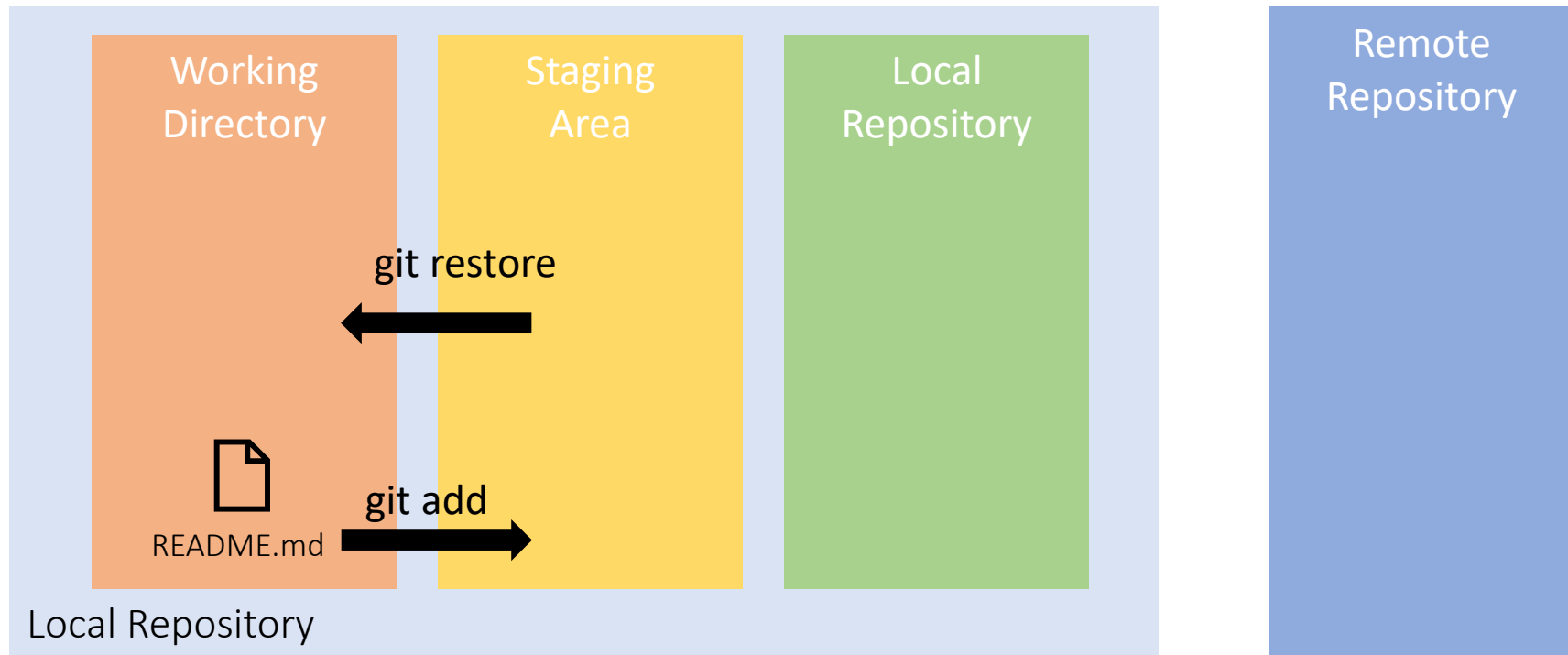
    README adapted and headline included

commit 3b3060f0edcfaec513f7f8ff70b19a022504a065
Author: Berti <bert.gollnick@posteo.net>
Date:   Wed Apr 24 08:38:00 2024 +0200

    README added
```

Getting Started

Removing from Staging Area



Getting Started

Removing from Staging Area

- add some more content to our file and save it

```
1  # Getting Started
2
3  ## What is Git?
4
5  Git is a distributed version control system that allows you to track changes in your codebase. It is a tool that
   helps you to manage your codebase and collaborate with other developers.
6
```

- git add
- git restore --staged README.md

Getting Started

Commit Best Practices

Messages

- describe content in best way
- never use „final“, „current“, „correction“
- use commit style guide, e.g.
 - add feature X
 - fix bug Y

Commits

- commit often...small changes
- commit related changes...bundle things
- separate unrelated changes



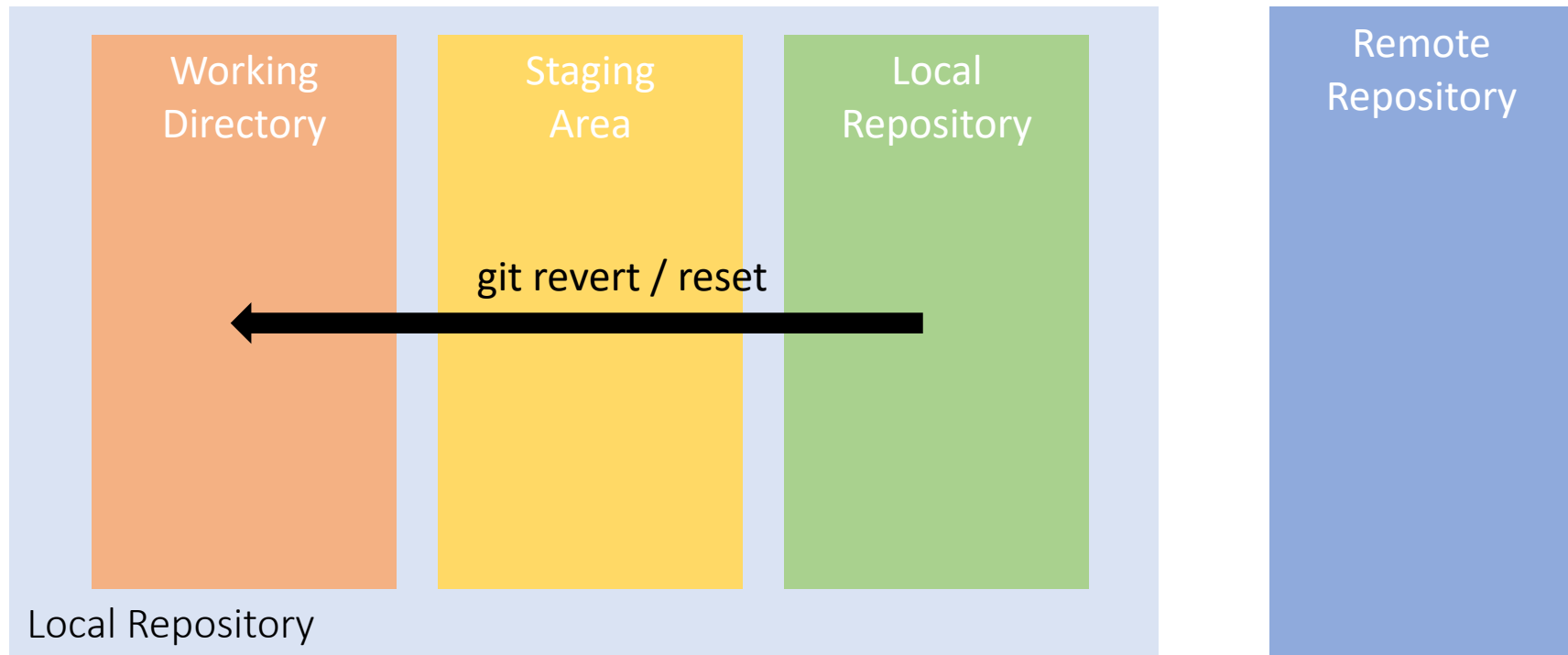
	COMMENT	DATE
○	CREATED MAIN LOOP & TIMING CONTROL	14 HOURS AGO
○	ENABLED CONFIG FILE PARSING	9 HOURS AGO
○	MISC BUGFIXES	5 HOURS AGO
○	CODE ADDITIONS/EDITS	4 HOURS AGO
○	MORE CODE	4 HOURS AGO
○	HERE HAVE CODE	4 HOURS AGO
○	AAAAAAA	3 HOURS AGO
○	ADKFJSLKDFJSDKLFJ	3 HOURS AGO
○	MY HANDS ARE TYPING WORDS	2 HOURS AGO
○	HAAAAAAAAAANDS	2 HOURS AGO

AS A PROJECT DRAGS ON, MY GIT COMMIT
MESSAGES GET LESS AND LESS INFORMATIVE.

Source: <https://xkcd.com/1296/>

Getting Started

Roll Back Commits



Getting Started

Removing Commits

- `git reset HEAD~2`...rolls back last two commits

```
$ git log
commit 21b88e266b77cd9fb8531efacefb8d8850c0a9b8 (HEAD -> master)
Author: Berti <bert.gollnick@posteo.net>
Date:   Wed Apr 24 16:00:12 2024 +0200

    chapter Why use Git? added

commit 72d59f79f60150b1ac7544e5a94a472ec852b7
Author: Berti <bert.gollnick@posteo.net>
Date:   Wed Apr 24 08:48:49 2024 +0200

    README adapted and headline included

commit 3b3060f0edcfaec513f7f8ff70b19a022504a065
Author: Berti <bert.gollnick@posteo.net>
Date:   Wed Apr 24 08:38:00 2024 +0200

    README added
```

`git reset HEAD~2`



```
$ git log
commit 3b3060f0edcfaec513f7f8ff70b19a022504a065 (HEAD -> master)
Author: Berti <bert.gollnick@posteo.net>
Date:   Wed Apr 24 08:38:00 2024 +0200

    README added
```

Getting Started

Storing Commits in Remote Repository


- Create new repo on GitHub

Create a new repository

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


Required fields are marked with an asterisk ().*

Owner *

 DataScienceHamburg ▾


Repository name *

testrepo


 testrepo is available.

Great repository names are short and memorable. Need inspiration? How about **cuddly-spork** ?

Description (optional)

☐  **Public**

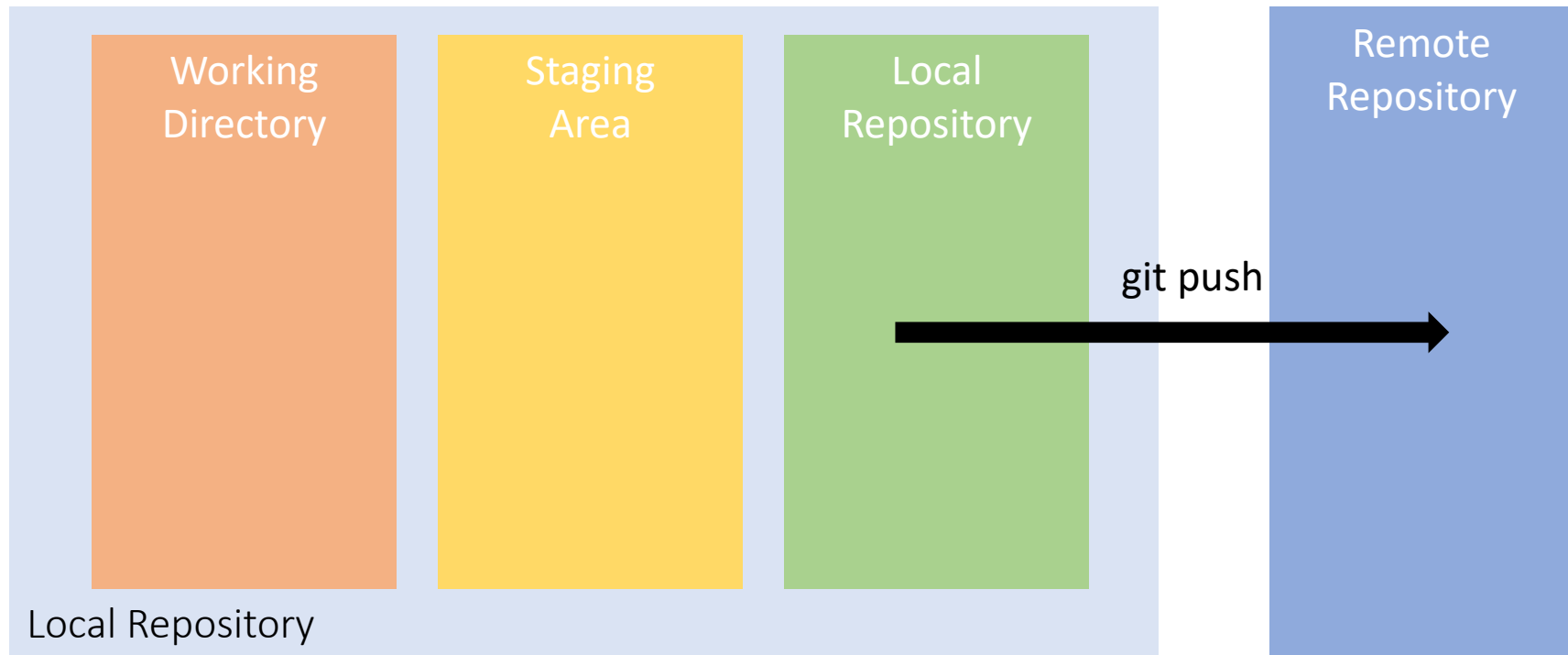
Anyone on the internet can see this repository. You choose who can commit.

☒  **Private**

You choose who can see and commit to this repository.

Getting Started

Storing Commits in Remote Repository



Getting Started

Storing Commits in Remote Repository

- Link local repo to remote repository

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

```
git remote add origin https://github.com/DataScienceHamburg/testrepo.git
git branch -M main
git push -u origin main
```

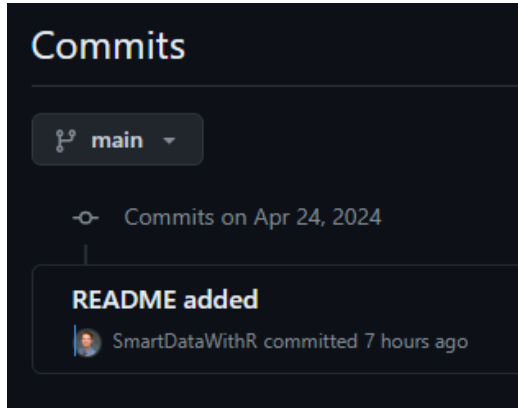
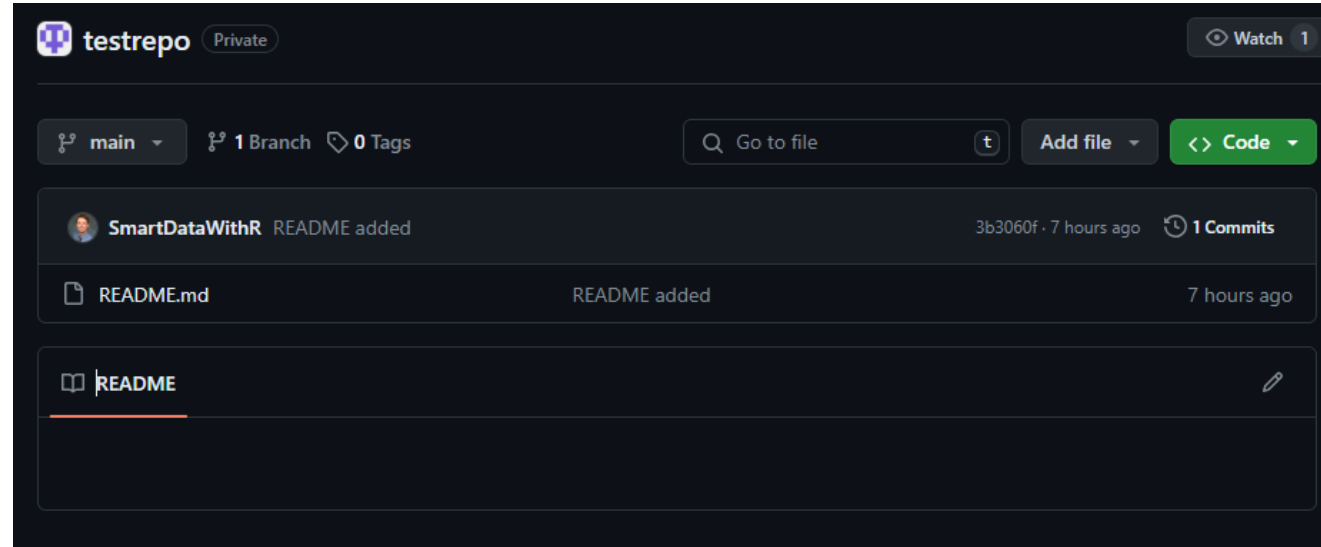
```
bertg@BERT2022 MINGW64 /c/Projects/DataAnalyst_Ultimate/20_Git/10_GettingStarted (master)
$ git branch -M main

bertg@BERT2022 MINGW64 /c/Projects/DataAnalyst_Ultimate/20_Git/10_GettingStarted (main)
$ git push origin main
Enumerating objects: 3, done.
Counting objects: 100% (3/3), done.
Writing objects: 100% (3/3), 218 bytes | 109.00 KiB/s, done.
Total 3 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)
To https://github.com/DataScienceHamburg/testrepo.git
 * [new branch]      main -> main
```

Getting Started

Storing Commits in Remote Repository

- checking result online



Getting Started

Storing Commits in Remote Repository

