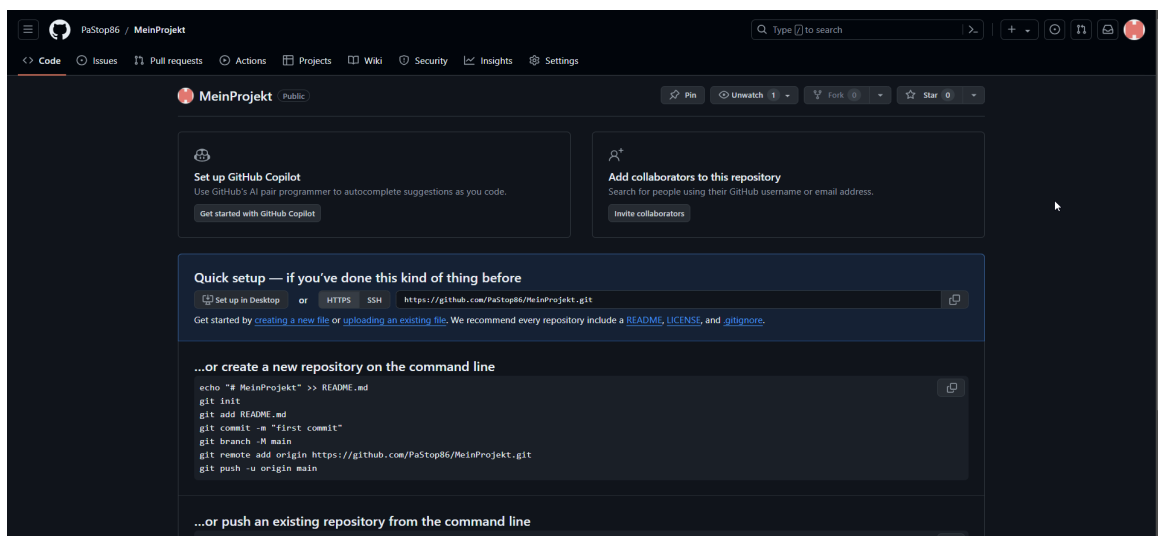
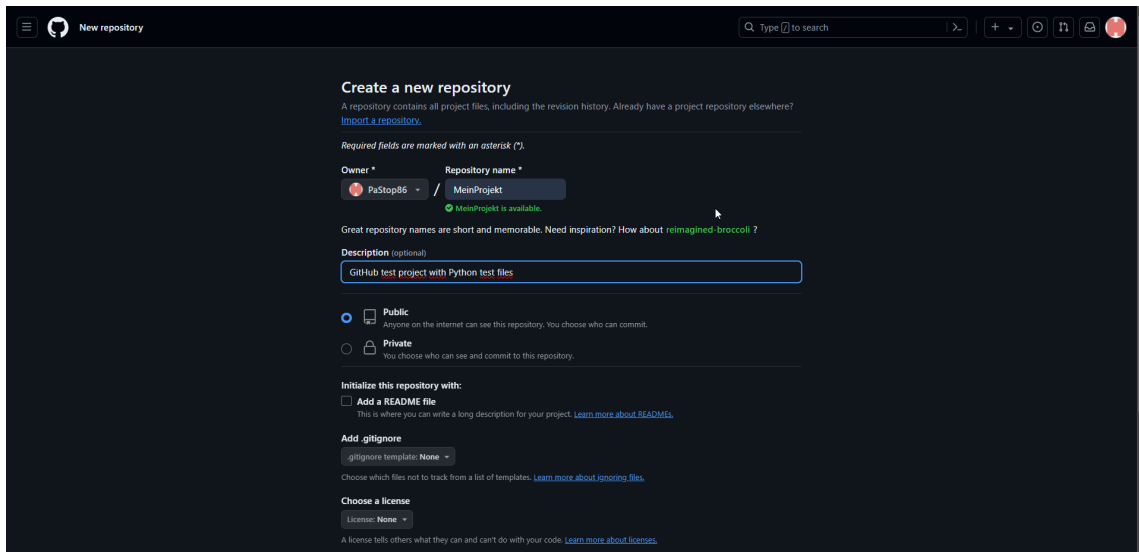


##Readme: Mein Testprojekt – Dokumentation
##Autor: P.Stoppel
##Datum: 24.05.2024

1. Repository „MeinProjekt“ in GitHub anlegen:



2. SSH Schlüssel in der GitBash erstellen:

```
$ ssh-keygen -t rsa -b 4096 -C "censored.email@mail.de"
Generating public/private rsa key pair.
Enter file in which to save the key (/c/Users/ps-dev/.ssh/id_rsa):
Created directory '/c/Users/ps-dev/.ssh'.
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /c/Users/ps-dev/.ssh/id_rsa
Your public key has been saved in /c/Users/ps-dev/.ssh/id_rsa.pub
...
```

3. Lokales Repository Verzeichnis erstellen

```
ps-dev@DESKTOP-8xyz3 MINGW64 /c/weiterBildung/1-Git-Github/Übungen/Prüfung
$ mkdir MeinProjekt-Repo
```

```
ps-dev@DESKTOP-8xyz3 MINGW64 /c/weiterBildung/1-Git-Github/Übungen/Prüfung
$ cd MeinProjekt-Repo/
```

4. Github Repository in das angelegte Verzeichnis clonen (hatte über Ssh nicht funktioniert-> somit den clone über https URL durchgeführt)

```
ps-dev@DESKTOP-8xyz3 MINGW64 /c/weiterBildung/.../MeinProjekt-Repo
$ git clone git@github.com:PaStop86/MeinProjekt.git
Cloning into 'MeinProjekt'...
Connection reset by 140.82.121.3 port 22
fatal: Could not read from remote repository.
```

Please make sure you have the correct access rights
and the repository exists.

```
ps-dev@DESKTOP-8xyz3 MINGW64 /c/weiterBildung/.../MeinProjekt-Repo
$ git clone https://github.com/PaStop86/MeinProjekt.git
Cloning into 'MeinProjekt'...
warning: You appear to have cloned an empty repository.
```

5. Ins lokale Repository Verzeichnis navigieren und die Autor Email und Namen über den Git Config Befehl setzen(Email-Adresse wurden aus Datenschutzgründen zensiert)

```
ps-dev@DESKTOP-8xyz3 MINGW64 /c/weiterBildung/.../MeinProjekt-Repo
$ cd MeinProjekt/

ps-dev@DESKTOP-8xyz3 MINGW64 /c/weiterBildung/.../MeinProjekt-Repo/MeinProjekt (main)
$ git config user.name "P.Stopp"

ps-dev@DESKTOP-8xyz3 MINGW64 /c/weiterBildung/.../MeinProjekt-Repo/MeinProjekt (main)
$ git config user.email "censored.email@mail.de"
```

6. Datei „main.py“ anlegen und mit Inhalt befüllen

```
ps-dev@DESKTOP-8xyz3 MINGW64 /c/weiterBildung/.../MeinProjekt-Repo/MeinProjekt (main)
$ touch main.py

ps-dev@DESKTOP-8xyz3 MINGW64 /c/weiterBildung/.../MeinProjekt-Repo/MeinProjekt (main)
$ echo "## Main Python file" >> main.py
```

7. Datei „main.py“ mit git add dem Staging-Bereich hinzufügen

```
ps-dev@DESKTOP-8xyz3 MINGW64 /c/weiterBildung/.../MeinProjekt-Repo/MeinProjekt (main)
$ git add main.py
warning: in the working copy of 'main.py', LF will be replaced by CRLF the next time Git
touches it
```

8. Initialen Commit der Datei „main.py“ mit git Commit durchführen

```
ps-dev@DESKTOP-8xyz3 MINGW64 /c/weiterBildung/.../MeinProjekt (main)
$ git commit -m "Initialer Commit"
[main (root-commit) 5f8178a] Initialer Commit
1 file changed, 1 insertion(+)
create mode 100644 main.py
```

9. Erstellung des Branch „feature“

```
ps-dev@DESKTOP-8xyz3 MINGW64 /c/weiterBildung/.../MeinProjekt (main)
$ git checkout -b feature
Switched to a new branch 'feature'
```

10. Anlegen und erstellen des „utils“ Verzeichnisses sowie der enthaltenen Datei database.py (Durchführung im 'feature' Branch)

```
ps-dev@DESKTOP-8xyz3 MINGW64 /c/weiterBildung/.../MeinProjekt (feature)
$ mkdir utils

ps-dev@DESKTOP-8xyz3 MINGW64 /c/weiterBildung/.../MeinProjekt (feature)
$ cd utils/

ps-dev@DESKTOP-8xyz3 MINGW64 /c/weiterBildung/.../MeinProjekt/utils (feature)
$ touch database.py

ps-dev@DESKTOP-8xyz3 MINGW64 /c/weiterBildung/.../MeinProjekt/utils (feature)
$ echo "##Test python database file" >> database.py
```

```
ps-dev@DESKTOP-8xyz3 MINGW64 /c/weiterBildung/.../MeinProjekt/utils (feature)
$ cd ..
```

11. Datei „database.py“ mit git add dem Staging-Bereich hinzufügen (Durchführung im 'feature' Branch)

```
ps-dev@DESKTOP-8xyz3 MINGW64 /c/weiterBildung/.../MeinProjekt (feature)
$ git add utils/database.py
```

12. Datei „database.py“ mit git Commit hinzufügen

```
ps-dev@DESKTOP-8xyz3 MINGW64 /c/weiterBildung/.../MeinProjekt (feature)
$ git commit -m "Neue Funktion hinzugefügt"
[feature 5e65bc8] Neue Funktion hinzugefügt
1 file changed, 1 insertion(+)
create mode 100644 utils/database.py
```

13. Zurück in den 'main' Branch wechseln

```
ps-dev@DESKTOP-8xyz3 MINGW64 /c/weiterBildung/.../MeinProjekt (feature)
$ git checkout main
Switched to branch 'main'
Your branch is based on 'origin/main', but the upstream is gone.
(use "git branch --unset-upstream" to fixup)
```

14. Inhalt von Datei 'main.py' im 'main' Branch verändern

```
ps-dev@DESKTOP-8xyz3 MINGW64 /c/weiterBildung/.../MeinProjekt (main)
$ echo "\n##Die Datei wurde veraendert" >> main.py
```

15. Veränderte Datei 'main.py' im 'main' Branch in den Staging bereich hinzufügen und einen Commit durchführen

```
ps-dev@DESKTOP-8UJP8K3 MINGW64 /c/weiterBildung/1-Git-Github/Übungen/Prüfung/MeinProjekt-Repo/MeinProjekt (main)
$ git add main.py
warning: in the working copy of 'main.py', LF will be replaced by CRLF the next time Git touches it
```

```
ps-dev@DESKTOP-8xyz3 MINGW64 /c/weiterBildung/.../MeinProjekt (main)
$ git commit -m "Hauptdatei aktualisiert"
[main d47284c] Hauptdatei aktualisiert
1 file changed, 1 insertion(+)

```

16. Den 'feature' Branch in den 'main' Branch über den Merge-Befehl zusammenführen

```
ps-dev@DESKTOP-8xyz3 MINGW64 /c/weiterBildung/.../MeinProjekt (main)
$ git merge feature
Merge made by the 'ort' strategy.
utils/database.py | 1 +
1 file changed, 1 insertion(+)
create mode 100644 utils/database.py
```

16.1 Merge-Konflikt im automatisch geöffneten Text-Editor Fenster bestätigen und Merge final ausführen

```
C: > WeiterBildung > 1-Git-Github > Übungen > Prüfung > MeinProjekt-Repo > MeinProjekt > .git > MERGE_MSG
1 Merge branch 'feature'
2 # Please enter a commit message to explain why this merge is necessary,
3 # especially if it merges an updated upstream into a topic branch.
4 #
5 # Lines starting with '#' will be ignored, and an empty message aborts
6 # the commit.
7
```

17. Finale Überprüfung des Repository Status, ob Merge erfolgreich war → nothing to commit, working tree clean

```
ps-dev@DESKTOP-8xyz3 MINGW64 /c/weiterBildung/.../MeinProjekt-Repo/MeinProjekt (main)
$ git status
On branch main
Your branch is based on 'origin/main', but the upstream is gone.
  (use "git branch --unset-upstream" to fixup)

nothing to commit, working tree clean
```