CoGrammar

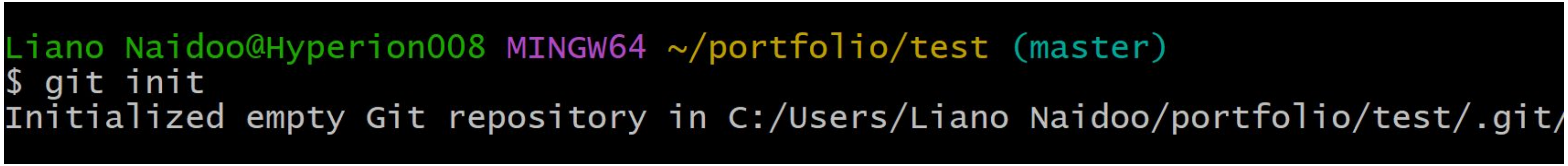
Tutorial: *Introduction to Git*

SKILLS FOR LIFE
SKILLS BOOTCAMPS

Department for Education

Initialising a git repository

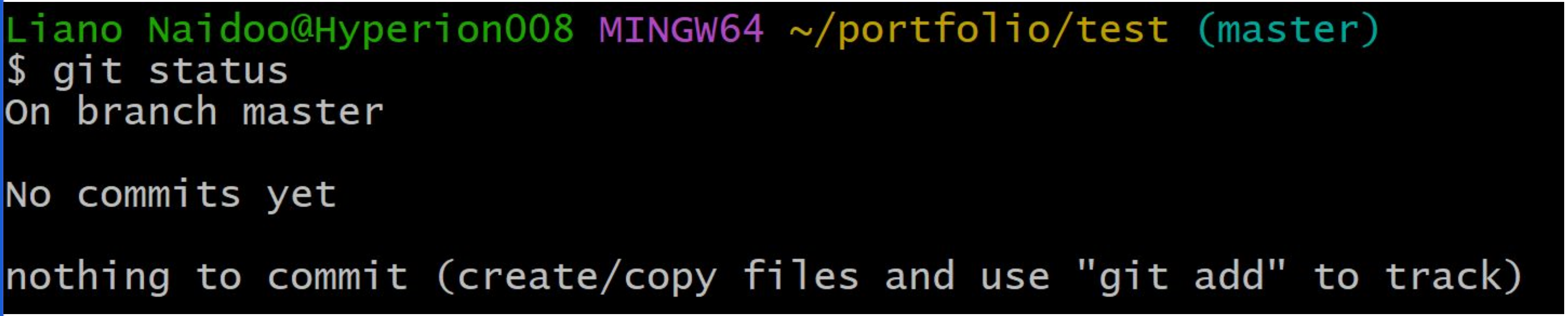
★ To initialise a git repository in our desired root folder, run the git init command.



```
Liano Naidoo@Hyperion008 MINGW64 ~/portfolio/test (master)
$ git init
Initialized empty Git repository in C:/Users/Liano Naidoo/portfolio/test/.git/
```

Git Status

★ After initialisation, we can use git status command to check what git is watching.



```
Liano Naidoo@Hyperion008 MINGW64 ~/portfolio/test (master)
$ git status
On branch master

No commits yet

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

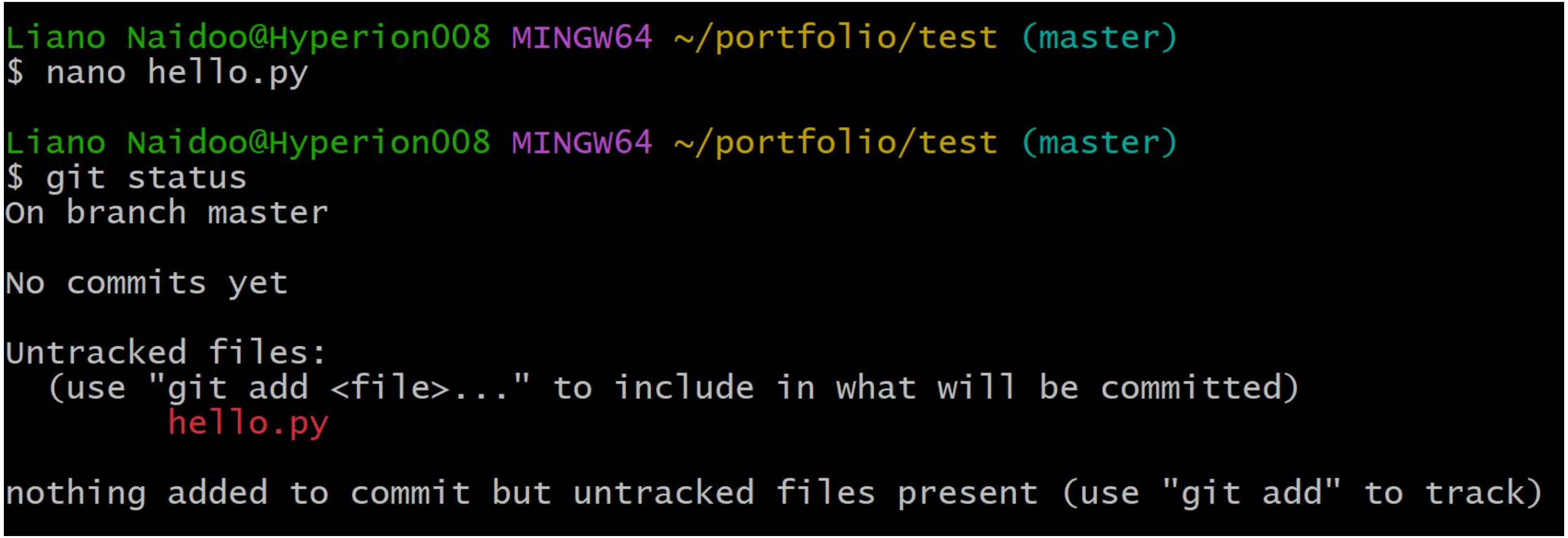
Adding a file to the Staging Environment

★ To add a file to the staging environment, we use the git add command.

★ If we rerun the git status command, we'll see that git has added the file to the staging environment.

CoGrammar

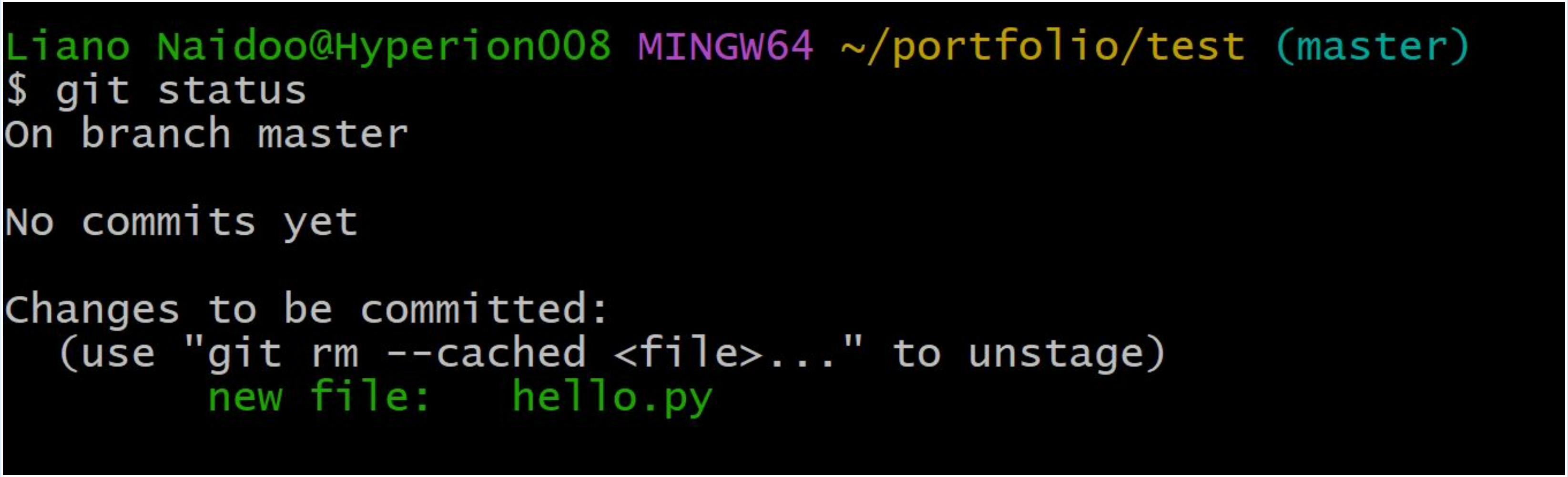
Adding a file to the Staging Environment



```
Liano Naidoo@Hyperion008 MINGW64 ~/portfolio/test (master)
$ nano hello.py

Liano Naidoo@Hyperion008 MINGW64 ~/portfolio/test (master)
$ git status
On branch master

No commits yet

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

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

CoGrammar

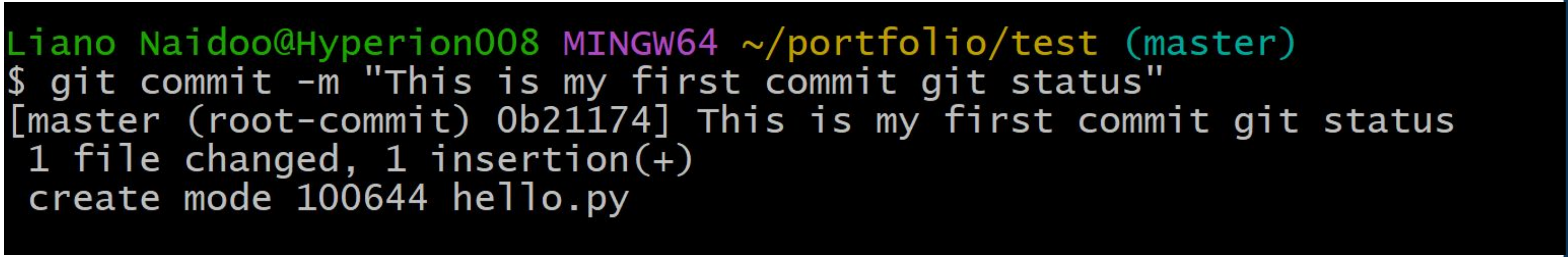
Adding a file to the Staging Environment



```
Liano Naidoo@Hyperion008 MINGW64 ~/portfolio/test (master)
$ git status
On branch master

No commits yet

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

Creating a Commit

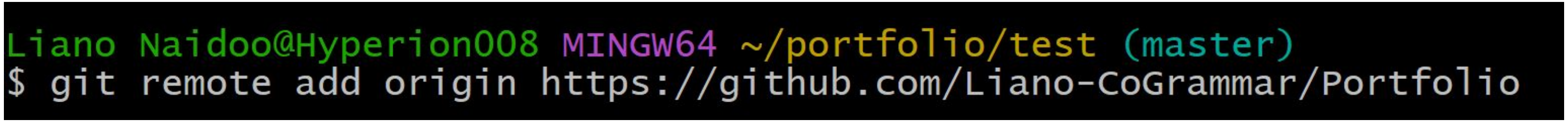
- To create a commit, run the git commit -m “message about commit” command.



```
Liano Naidoo@Hyperion008 MINGW64 ~/portfolio/test (master)
$ git commit -m "This is my first commit git status"
[master (root-commit) 0b21174] This is my first commit git status
 1 file changed, 1 insertion(+)
 create mode 100644 hello.py
```

Adding Remote Repository

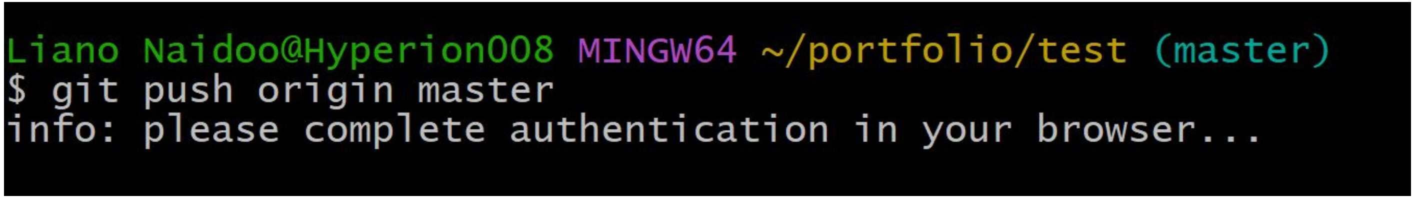
★ To create a commit, run the git commit -m “message about commit” command.



```
Liano Naidoo@Hyperion008 MINGW64 ~/portfolio/test (master)
$ git remote add origin https://github.com/Liano-CoGrammar/Portfolio
```

Pushing to the remote repository

- To create a commit, run the git commit -m “message about commit” command.



```
Liano Naidoo@Hyperion008 MINGW64 ~/portfolio/test (master)
$ git push origin master
info: please complete authentication in your browser...
```

Creating a new Branch

★ We create different branches for the purpose of if we’re creating a new feature to our project, but we do not want this new code to make changes to our main project.

★ Branches allow us to move back and forth between ‘states’ of a project

Creating a new Branch

★ E.g if we want to add a new page to a website, we can create a new branch just for that page without affecting the main part of the project.

★ We can merge our changes from our branch to the main branch.

★ When creating a new branch, Git keeps track of which commit your branch ‘branched’ off of, so git knows the history behind all files.

Branching

★ Let’s say we’re on the master branch and want to create a new branch to develop a new feature.

★ Run git checkout -b “branch name”

★ This command will automatically create a new branch and then ‘check you out’ on it.

★ We can then use the git branch command to confirm that your branch is created.

CoGrammar

Thank you for joining

CoGrammar