

**TU856/858-2 Object Oriented Programming with Python**  
**Week 1 - Tutorial**  
**Dr. Bianca Schoen-Phelan**

```
0:$ mkdir Project1.0
```

```
1:$ cd Project1.0/
```

```
2:$ vi hello_world.py
```

```
3:$ git init
```

```
Initialized empty Git repository in  
/Users/bianca.schoenphelan/Documents/gitExamples/Project1.0/.git/
```

```
4:$ git add hello_world.py
```

```
5:$ git status
```

```
On branch master
```

```
No commits yet
```

```
Changes to be committed:
```

```
(use "git rm --cached <file>..." to unstage)
```

```
new file:   hello_world.py
```

```
6:$ git commit -m "Original File"
```

```
[master (root-commit) ace9ada] Original File
```

```
Committer: Bianca SchoenPhelan
```

```
<bianca.schoenphelan@soc-mbp13-bsp.lan>
```

```
Your name and email address were configured automatically based  
on your username and hostname. Please check that they are  
accurate.
```

```
You can suppress this message by setting them explicitly. Run the  
following command and follow the instructions in your editor to  
edit
```

```
your configuration file:
```

```
git config --global --edit
```

```
After doing this, you may fix the identity used for this commit  
with:
```

```
git commit --amend --reset-author
```

```
1 file changed, 2 insertions(+)  
create mode 100644 hello_world.py
```

```
7:$ git log --oneline
```

```
ace9ada (HEAD -> master) Original File
```

```
8:$ git checkout -b feature_1
```

```
Switched to a new branch 'feature_1'
```

```
9:$ git branch
```

## TU856/858-2 Object Oriented Programming with Python

### Week 1 - Tutorial

Dr. Bianca Schoen-Phelan

```
* feature_1
```

```
    master
```

```
10:$ cat hello_world.py
```

```
print('Hello world')
```

```
11:$ vi hello_world.py
```

```
12:$ git add hello_world.py
```

```
13:$ git commit -m "First change in feature_1"
```

```
[feature_1 22c5d4b] First change in feature_1
```

```
Committer: Bianca SchoenPhelan
```

```
<bianca.schoenphelan@soc-mbp13-bsp.lan>
```

Your name and email address were configured automatically based on your username and hostname. Please check that they are accurate.

You can suppress this message by setting them explicitly. Run the following command and follow the instructions in your editor to edit

your configuration file:

```
git config --global --edit
```

After doing this, you may fix the identity used for this commit with:

```
git commit --amend --reset-author
```

```
1 file changed, 1 insertion(+)
```

```
14:$ git log --oneline
```

```
22c5d4b (HEAD -> feature_1) First change in feature_1
```

```
ace9ada (master) Original File
```

```
15:$ vi hello_world.py
```

```
16:$ git add hello_world.py
```

```
17:$ git commit -m "Second change in feature_1"
```

```
[feature_1 198a797] Second change in feature_1
```

```
Committer: Bianca SchoenPhelan
```

```
<bianca.schoenphelan@soc-mbp13-bsp.lan>
```

Your name and email address were configured automatically based on your username and hostname. Please check that they are accurate.

You can suppress this message by setting them explicitly. Run the following command and follow the instructions in your editor to edit

your configuration file:

```
git config --global --edit
```

## TU856/858-2 Object Oriented Programming with Python

### Week 1 - Tutorial

#### Dr. Bianca Schoen-Phelan

After doing this, you may fix the identity used for this commit with:

```
git commit --amend --reset-author
```

```
1 file changed, 1 insertion(+)
```

```
18:$ git log --oneline
```

```
198a797 (HEAD -> feature_1) Second change in feature_1
```

```
22c5d4b First change in feature_1
```

```
ace9ada (master) Original File
```

```
19:$ cat hello_world.py
```

```
print('Hello world')
```

```
print('First change')
```

```
print('Second change')
```

```
20:$ git checkout 22c5d4b
```

```
Note: checking out '22c5d4b'.
```

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 22c5d4b First change in feature_1
```

```
21:$ cat hello_world.py
```

```
print('Hello world')
```

```
print('First change')
```

```
22:$ git branch
```

```
* (HEAD detached at 22c5d4b)
```

```
feature_1
```

```
master
```

```
23:$ git log --oneline
```

```
22c5d4b (HEAD) First change in feature_1
```

```
ace9ada (master) Original File
```

## TU856/858-2 Object Oriented Programming with Python

### Week 1 - Tutorial

Dr. Bianca Schoen-Phelan

**24:\$ git checkout feature\_1**

Previous HEAD position was 22c5d4b First change in feature\_1  
Switched to branch 'feature\_1'

**25:\$ git log --oneline**

198a797 (HEAD -> feature\_1) Second change in feature\_1  
22c5d4b First change in feature\_1  
ace9ada (master) Original File

**26:\$ cat hello\_world.py**

```
print('Hello world')  
print('First change')  
print('Second change')
```

**27:\$ git revert HEAD**

```
[feature_1 d899b2b] Revert "Second change in feature_1"  
Committer: Bianca SchoenPhelan  
<bianca.schoenphelan@soc-mbp13-bsp.lan>  
Your name and email address were configured automatically based  
on your username and hostname. Please check that they are  
accurate.  
You can suppress this message by setting them explicitly. Run the  
following command and follow the instructions in your editor to  
edit  
your configuration file:
```

```
git config --global --edit
```

After doing this, you may fix the identity used for this commit with:

```
git commit --amend --reset-author
```

```
1 file changed, 1 deletion(-)
```

**28:\$ git log --oneline**

```
d899b2b (HEAD -> feature_1) Revert "Second change in feature_1"  
198a797 Second change in feature_1  
22c5d4b First change in feature_1  
ace9ada (master) Original File
```

**29:\$ cat hello\_world.py**

```
print('Hello world')  
print('First change')
```

**30:\$ git branch**

```
* feature_1  
master
```

**31:\$ git checkout master**

## TU856/858-2 Object Oriented Programming with Python

### Week 1 - Tutorial

Dr. Bianca Schoen-Phelan

Switched to branch 'master'

**32:\$ git branch**

```
feature_1  
* master
```

**33:\$ cat hello\_world.py**

```
print('Hello world')
```

**34:\$ git merge feature\_1**

Updating ace9ada..d899b2b

Fast-forward

```
hello_world.py | 1 +  
1 file changed, 1 insertion(+)
```

**35:\$ cat hello\_world.py**

```
print('Hello world')
```

```
print('First change')
```

**36:\$ git branch -d feature\_1**

Deleted branch feature\_1 (was d899b2b).

**37:\$ git branch**

```
* master  
$
```