

```
In [ ]: # Initialize Otter
import otter
grader = otter.Notebook("test_assignment.ipynb")
```

**Q1** Clone the repository <https://github.com/zpqrtnk/test-repo.git>

```
In [1]: !git clone https://github.com/zpqrtnk/test-repo.git # SOLUTION
```

```
Cloning into 'test-repo'...
remote: Enumerating objects: 1687, done.
remote: Counting objects: 100% (14/14), done.
remote: Compressing objects: 100% (12/12), done.
remote: Total 1687 (delta 2), reused 10 (delta 0), pack-reused 1673 (from 1)
Receiving objects: 100% (1687/1687), 251.45 KiB | 2.89 MiB/s, done.
Resolving deltas: 100% (877/877), done.
```

```
In [2]: # DO NOT CHANGE THE BELOW CODE - RUN THIS CELL
import os
os.chdir('test-repo')

import subprocess
def run_shell_command(command):
    process = subprocess.Popen(command, stdout=subprocess.PIPE, stderr=subprocess.PIPE, shell=True)
    output, error = process.communicate()

    if process.returncode != 0:
        print(f"Error: {error.decode().strip()}")
        return None

    return output.decode().strip()

initial_hash = run_shell_command('git rev-parse HEAD')
os.chdir('..')
```

```
In [ ]: grader.check("q1")
```

**Question 2** Add a new file named “new\_file.txt” and commit it

```
In [4]: # DO NOT CHANGE BELOW CODE - RUN THIS CELL
os.chdir("test-repo")
```

```
In [5]: # BEGIN SOLUTION
!echo "New empty file" > new_file.txt
```

```
!git add new_file.txt
!git commit -m "Added first commit"
# END SOLUTION
```

```
[master 59181be] Added first commit
1 file changed, 1 insertion(+)
create mode 100644 new_file.txt
```

```
In [6]: # DO NOT CHANGE BELOW CODE - RUN THIS CELL
os.chdir("..")
```

```
In [ ]: grader.check("q2")
```

## 0.1 Submission

Make sure you have run all cells in your notebook in order before running the cell below, so that all images/graphs appear in the output. The cell below will generate a zip file for you to submit. **Please save before exporting!**

These are some submission instructions.

```
In [ ]: # Save your notebook first, then run this cell to export your submission.
grader.export(run_tests=True)
```