Programming Practical 2 - git

1 Setup

- To set your username and password:

```
git config --global user.name "Username"
git config --global user.name "email@domain.com"
```

- To set the editor

```
git config --global core.editor "nano -w"
```

- To check settings

```
git config --list
```

2 Creating a repository

- To initialise a repository

```
git init
```

- To check the status of the project

```
git status
```

3 Tracking changes

- To add a directory to staging

```
git add <file>
```

- To commit the staging area to the repository

```
git commit -m "Commit comment"
```

- To see what has been done recently

```
git log
```

- To see what has been changed since the last commit

```
git diff
```

- To see what has been changed and added to the staging area since the last commit

```
git diff --staged
```

- To limit the log size:

```
git log -1
```

  To display a more compressed version of the log

```
git log --oneline
```

4 Exploring history

- HEAD can be used to refer to the most recent commit, so to get the changes in the most recent commit:

  ```
  git diff HEAD <file>
  ```

- To refer to commits before the head, use a after the head, so to refer to the penultimate commit:

  ```
  git diff HEAD˜1 <file>
  ```

- To see the changes, as well as the commit message use git show, for example

  ```
  git show HEAD˜2 <file>
  ```

- To refer to specific commits, use the hex string provided when using git log –oneline

  ```
  git diff f22b25e <file>
  ```

- To change things back to how they were use git checkout

  ```
  git checkout f22b25e <file>
  ```