≡   >▭_ (/learn)   ⚙   📋

Setting up a Bare Repository

Set up a repository that will help you understand the concept of "bare repositories."

We'll cover the following   ⌃

- Configuration
- The --bare flag
- config file

Configuration

In this example, you're going to repeat the last section's work. But this time, you're going to be able to push your commits at the end. This will be achieved by having a bare repository that your two imagined users will pull from and push to.
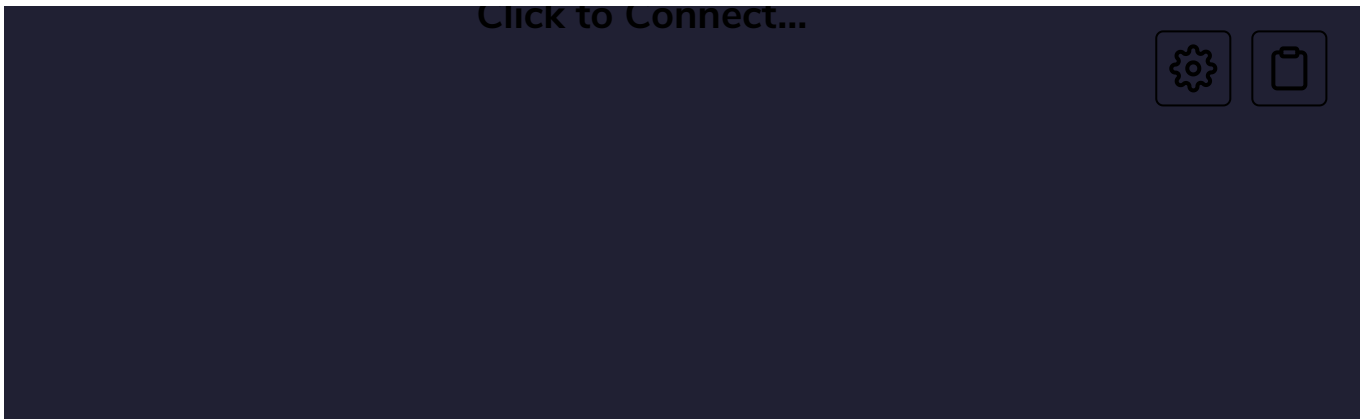
Type this in:

```
1    mkdir lgthw_bare_repo
2    cd lgthw_bare_repo
3    git init --bare
```

Terminal 1   ⟳

Terminal   ⌃

Click to Connect...

Did you spot the difference?

The `--bare` flag

This time, you created a Git repository with the `--bare` flag. Have a look around the folder that you are in now. What's the main difference you notice between this and a "normal" repository?

`config` file

One difference I want to point out is in the `config` file:

```
4    cat config
```

Diff this with a normal Git repository to see the difference, and then research what the differences mean.

You can't do much with this repository directly, as there's nowhere to put your content. Now type this, and think about what you're doing:

```
5    cd ..
6    git clone lgthw_bare_repo lgthw_bare_repo_alice
7    git clone lgthw_bare_repo lgthw_bare_repo_bob
```

Terminal 1

Terminal

You've created two clones of the original bare repository. Each of these will represent the work of a separate user working on a project: Alice and Bob.

```
8    cd lgthw_bare_repo_alice
9    git remote -v
10   touch afile
11   git add afile
12   git commit -m 'Initial'
13   for i in {1..10}
14   > do
15   > echo $i >> afile
16   > git commit -am "commit:${i}"
17   > done
18   git log --oneline
19   git push
```

Terminal 1   ↻

Terminal   ⌃

You've made ten commits as Alice. You pushed them to the remote (bare) repository, ready to be picked up by Bob.
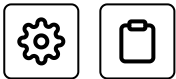
Now type this:

```
20  cd ../lgthw_bare_repo_bob
21  git log --oneline
```

You should be able to explain why that failed. What do you need to do to get Alice's change in your log?

```
22  git fetch origin
23  git merge origin/master
24  git log --oneline
```

Now all three repositories are in sync with the ten commits in all of them.

⚙️ 📋

← **Back**

Next →

Introduction: Bare Repositories

Squashes the Commits

✅ Mark as Completed

⚠️ Report an
Issue

❓ Ask a Question
(https://discuss.educative.io/tag/setting-up-a-bare-repository__bare-
repositories__learn-git-the-hard-way)