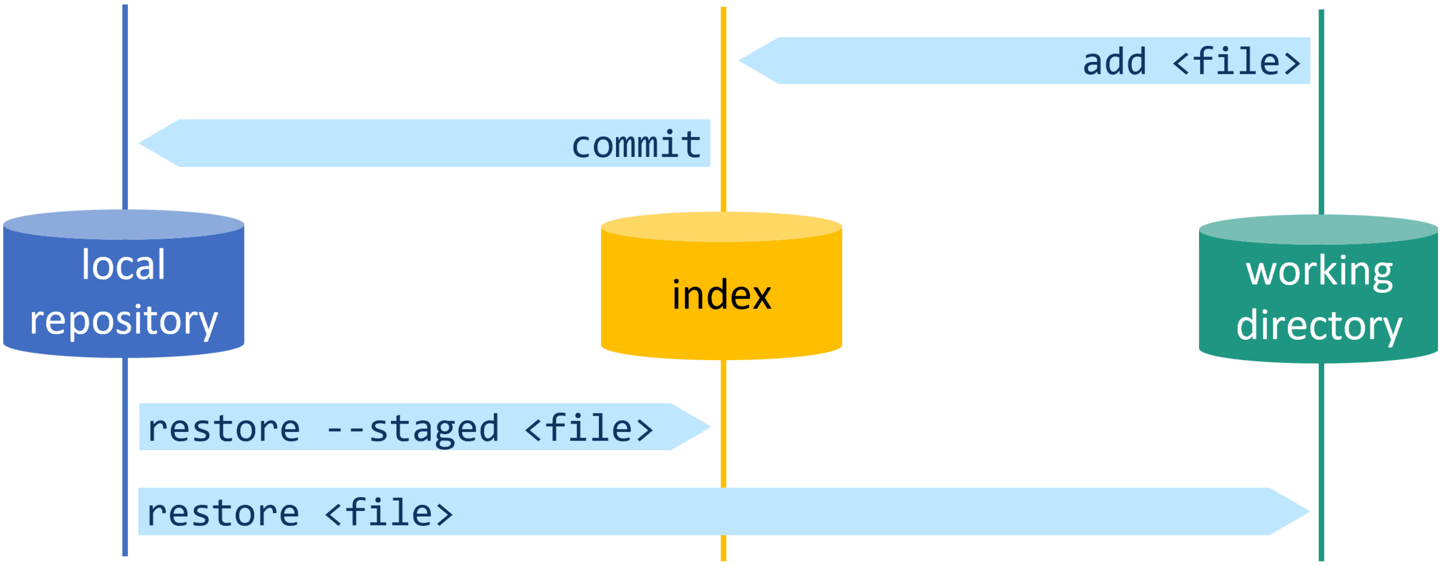
Lab 2 - Basic Snapshotting

Last updated by | kees.vanloenen | Jun 29, 2025 at 11:37 AM GMT+2

Lab 2 - Basic Snapshotting

Let's add some files and commit them in the repository created in the previous step.



(we'll practise `restore` in next lab)

Exercise 1

1. In the directory *`demo`* create a new text file.
2. Check the status (`status`).
3. Add this file to the staging area (`add`).
4. Check the status (again). You'll use this command quite often.
5. Now remove the change from the staging area using the suggested command.
6. The file is still in the working directory but no longer in the staging area. Check the status.
7. Add the file to the staging area again (`add`).
8. Commit the added file (`commit`).
9. Check the status.
10. Watch the log (`log`).

Exercise 2

1. Change the contents of the file.
2. Check the status. Why is the displayed command to discard the changes in the working directory *different* compared to the command we performed in exercise 1?
3. Now undo the change (`restore`).
4. Check the status.
5. Change the contents of the file (again).
6. Try to commit without adding the file to the staging area (`commit`).
7. What happened? What did you expect?

Exercise 3

1. Add the changed file to the staging area.
2. Check the status.
3. Commit the changes.
4. Check the status.
5. Watch the log.

Exercise 4

1. Create a second text file.
2. Check the status.
3. Ensure the file is committed in the repository.
4. Watch the 3 commits.

Exercise 5

If time permits.

1. Watch the most recent commit:

   ```
   git log -1
   ```

2. To further inspect this commit object, run command below. Replace the shown number 'c796a7c' with the first 7 characters of the SHA1 hash of your commit:

   ```
   git cat-file -p c796a7c
   ```

   - In this object a `tree` and a `parent` hash are given. Can you find the `tree` object in the file system?
3. First try to predict what's in the `tree` object!

   Afterwards, open the tree object.

   ```
   git cat-file -p 3ecd4ca
   ```

4. Feel free to inspect the content of a `blob` object.