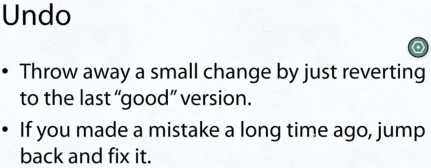
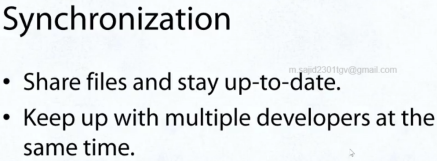
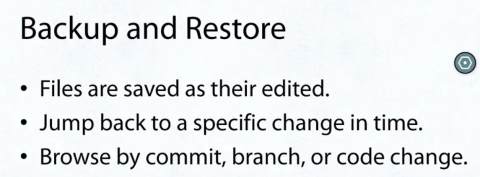
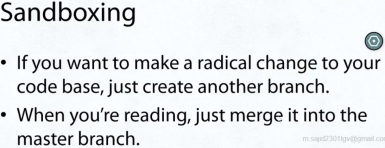
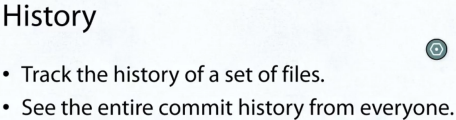
**What is version control?**





**What is Git?**

* Git is a distributed version control system.
* It allows for easy collaboration and version control.
* It was created in2005 by Linux Torvald for development on Linux. Today, it is one of the most popular versioning systems out there.

**How does Git work?**

**Repositories**

* It holds the history of all changes.
* In a general sense, the repo is a directory.
* We use Git through a CLI (Windows Command Prompt, Terminal, etc)

**How would I use Git?**

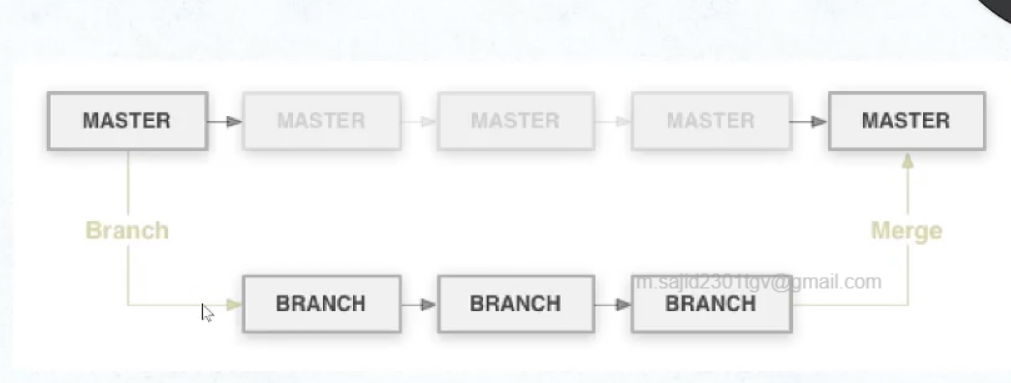
* To get the code and history of any repository, you clone it.
* git clone git://github.com/eturk/jack
* To initialize a new repo: git init
* To see the history of any repository, you see the log of it. Command: git log
* Every change in the history of a project is represented by a commit. Each commit has a SHA-1 ID.

**Branches**

* A branch is a different version of the same project.
* You use a branch for keeping some code separate from the main branch (usually called “master”).
* For example, I use the branch “stable” to differentiate between my development (“master”) and my code that is ready for use (“stable”).
* You can see the current branch you are using. Command: git branch
* And change the current branch. Command: git checkout foobar

**Merging**

* Once you have isolated a branch, you will want to incorporate those changes back into the main branch. You want to select the branch you want to merge into and specify the branch you are merging from.
* Commands: git checkout stable -> git merge master



**Branches and Merging**

* When you merge, Git will show you a diff.
* A diff is the difference between the current code and the code you want to merge in. Command: git diff
* When the same block of code is edited in two branches, Git doesn’t know how to merge.
* Instead, it will give us a “merge conflict” error and insert markers in the file where the difference is.

Git Setup