CS 240: GitHub Demo Transcript

[00:00:00] **INSTRUCTOR:** So, now that I've got my GitHub account set up and I've created my repository and I've cloned the repository to my local development machine, now I'm ready to do some work. Now, the next thing I'm going to do here is because my repository is essentially empty, I'm going to copy in the chess starter files that we've been using into this project as well.

[00:00:22] So, I'm going to unzip my chess.zip file. And so now, all of a sudden, my project has lots of files. Now, if I go back to GitHub and I refresh my browser, you'll see that it's still empty, right?

Start visual description. Instructor types the following on the command line: unzip ../../chess.zip. End visual description.

[00:00:49] So the files that I just added are copied or are just local on my own machine. So now, if I go back, I'm going to add all those files to my local copy of the repository, so I'm going to stage everything except what gitignore ignores.

[00:01:10] So now, if I do git status, all those files have been staged. Now, I can do a local commit. Do my commit git status. OK, so we're all good there. So I've committed those changes to my local copy of the repository.

Start visual description. Instructor types the following on the command line:

git status

git commit -m "Initial chess commit"

End visual description.

[00:01:34] But if I go back to GitHub and refresh it, still nothing. That's because I haven't done a push. So the next step here would be to do a git push. And so it just took all the changes that I've committed locally and it pushed them up into GitHub.

Start visual description. Instructor types the following on the command line: git push. End visual description.

[00:01:57] So now, if I go back to the GitHub interface, my refresh. Now all my files are all there. So just keep in mind that there's a couple of different copies of this repository, right?

Start visual description. Screen shows the chess-demo repository on GitHub. End visual description.

[00:02:12] There's the one in GitHub, the one that's in the cloud. There's the one that you have locally on your machine. If you're working with other people, they also have local copies of the repository on their computers.

[00:02:23] And so really, you're all working individually until you do pushes and pulls. And that's where the sharing and the collaboration comes in. Now I've been using the command line to do my git pushes and git pulls.

[00:02:37] If we go back to IntelliJ, let's try to open the new project that we just created. You can also do pushes and pulls in IntelliJ. So if you go up to the Git menu, you see that you can do a push.

[00:03:03] You can do a pull. If I've made changes to the files, let's just make a little change here to the readme. Add some new content. Save it. And we can go back to the commit tab. We can see that we made some changes here.

[00:03:24] I'm going to go ahead and commit those. My commit messages are pretty lame because I'm not doing real work. Now, you'll notice here, having made a local change, I can just commit it locally, or I can do a commit and a push.

Start visual description. Instructor makes a change in one of the files in the chess project and commits the change to the git repository. End visual description.

[00:03:43] Now, you have a choice. You may not want to push your changes up to GitHub immediately. You might just want to commit them locally. But I'm going to commit and push in one shot.

[00:04:04] And so now, if I go up to GitHub and look at the readme file, hopefully there's some new content. Ah, there it is, new content. So everything we've been doing at the command line you can also do through IntelliJ.