CS 240: IntelliJ Git Integration Transcript

[00:00:00] **INSTRUCTOR:** So far, we've been using git from the command line. So we've opened a terminal. We've used the git command line to do various operations. Typically, you won't use git in that fashion. Sometimes you will.

[00:00:15] Sometimes you won't. And it's important to know how to use the git command line. But oftentimes, you'll want to just use your IDE or your integrated development environment to do git operations. So most modern IDEs are integrated with git.

[00:00:31] And so they understand that your project is a git repository. And all the commands that we did through the command line, you can actually do through the graphical interface of your IDE. For example, here we have our chess project again.

[00:00:45] And you'll notice over here on the left, well, actually, in the bottom right, you can kind of tell that this is a git project. This little icon down here kind of tells you that this is a git project.

[00:00:58] But more than that, if you go over here on the left-hand side, you can see this tab called commit. So commit is a word we're familiar with. And so if you open that tab, this is essentially a graphical interface through which you can stage your files to be committed and actually commit them.

[00:01:16] And so if you see here, there's a couple of changes that we already made to this repository. We deleted the .gitignore file. We renamed the notes.md file to be mynotes.md. And so you can see that those changes are listed here.

Start visual description. Instructor clicks on the commit tab on the left hand side of the screen to show what changes have been made to the git repository. End visual description.

[00:01:32] So IntelliJ is aware of all those changes we're making to the repository. And let's say I want to commit those changes. All I would have to do is select the changes that I want to commit to the repository.

[00:01:48] And then here, I can type in a commit message. And then all I would have to do is click commit, and that would both stage those changes and commit them to the repository in one shot. And so it's a really simple interface.

Start visual description. Instructor shows how to add a message when committing changes to a repository. End visual description.

[00:02:12] Just check the boxes next to the files that you want to include in your commit, type in your commit message, and then click the commit button. Now there's other things you can do here as well. You can also roll back changes.

[00:02:28] So maybe I deleted my gitignore file, and now I really regret that because that was not very smart. And so I'm going to just right click on that change, and I'm going to say roll back. And you can see that it rolled that change back by checking out the .gitignore file from my repository.

Start visual description. Instructor right clicks on the gitignore file and selects rollback to revert back to an old version. End visual description.

[00:02:48] So basically just did a git checkout for me. And I still have this change here. So that's useful. There's also a menu up here named Git, and there's a whole bunch of other Git commands that you can do that are more advanced than what we've talked about so far, but just about anything you can do with Git, or at least many things you can you can do through the IntelliJ graphical interface.

[00:03:15] The most important one is this commit tab where you can select your changes to commit and commit them. Now earlier we talked about how if you have a Git project that it's really important that you're careful when you rename files or move files between folders or delete files.

[00:03:33] So of course all those operations can be done through the IntelliJ interface. So for example, if I click on a file and I want to, let's say we want to move it. So I'm going to take that file and I'm going to move it into another folder.

[00:03:57] I'm going to move it to a folder called Shared. And so now if I go to my shared folder, I've got that pom.xml file. Now I could have done that in a terminal or in my file explorer but I did it through IntelliJ.

Start visual description. Instructor shows how to move a git repository. End visual description.

[00:04:23] Now because IntelliJ is aware of Git, it actually underneath the covers actually does the correct Git commands so that these file operations or Git is made aware of those things. So now if I go back to the commit tab, you can see that it says, oh, we've added a new file named pom.xml in the shared folder and we've removed a file here that used to be there.

[00:04:50] And so Git is aware of those changes because IntelliJ does all the right things underneath to make sure that it's aware. So it is safe to do file operations through your IDE.