CS 240: Creating Gson Project Dependency Transcript

[00:00:01] In the chess project, we're going to use the Gson serialize and deserialize uh to, to transfer objects across the network.

Start visual description. The professor demonstrates how to use Gson to serialize and deserialize objects in the chess project, explaining the process of transferring objects across the network. End visual description.

[00:00:11] And so we're going to make and we're also going to use gong to store chess games in the database.

[00:00:17] And so we're going to need to learn how to create a dependency on the Gson library in our chess project.

Start visual description. The professor shows how to create a dependency on the Gson library in the chess project, emphasizing the importance of adding external libraries to the project. End visual description.

[00:00:24] As you've hopefully learned by now, anytime you use an external library in a project, you need to create a dependency inside your project so that your ID E and tell in this case, knows that you're using that library in, in your project.

[00:00:39] And so if we're going to use Gson, we need to add a Gson dependency to our project. Now, there's different ways to do that within IntelliJ in your chest project, you're probably going to want to use the second approach here because the, the code that we provide for the chest project um uses a maven project.

[00:01:02] Now, we're not going to talk about what that exactly means for now, but just understand that the project that you're using for chess, even though it's an Intel project is using another tool underneath called maven that manages dependencies for it. And so, um that means that in your project in your chess project, there's a file named Palm dot XML pom dot XML.

[00:01:23] And what you're going to need to do to use DEA in your project is go to your palm dot XML file, which is in the root folder of the project and open it up and inside that file, you'll, you'll find a an XML tag called dependencies.

Start visual description. The professor navigates to the pom.xml file in the root folder of the project and explains how to add the Gson dependency to the dependencies list. End visual description.

[00:01:41] And so what you want to do is you want to add this dependency here that's shown on this slide to the dependencies list in your, in your palm dot XML file.

[00:01:51] And then if, if you add that, then the next time you build the project, what it will do is it will download um the Gong library for you and, and import that into your project so that you can use um all the GGO classes.

[00:02:05] So once you've added this dependency to your project, then you can go ahead and import the various Gong classes in your code using java import statements.

[00:02:14] So if you're, if you've got a, a Maven project, which you do, um this is how you would add the, the Gson dependency.

[00:02:22] Now, if you're not using a Maven project in Intel, it's just a regular native Intel project.

[00:02:27] The way you would add dependencies is, is this first approach here and I'll demonstrate what that looks like here.

[00:02:33] So if I have a regular Intel project. What I would do to add the Gson dependency is I would go to the file menu and look for a menu item named project structure.

[00:02:46] Now it turns out that this project structure, menu item is really important in Intel J because it's a place where you can go to modify the structure of your project in a variety of ways.

[00:02:58] But one aspect of project structure that you can change here is the dependencies or the external libraries that you're using.

[00:03:04] So I'm going to select project structure and in project structure, there's different sections of settings here, but I'm going to go to the modules section.

Start visual description. The professor selects the project structure menu in IntelliJ, demonstrating how to modify the structure of the project and add external libraries. End visual description.

[00:03:13] Cause when you create a dependency, you're going to create it on a module.

[00:03:20] And once I've selected modules, then I would select the module.

[00:03:25] Now most projects have multiple modules, and you'll learn about modules later uh when we get farther into the chess project.

[00:03:34] But um this project only has one module named code.

[00:03:38] And so I'm going to select the code module.

[00:03:41] And then over here, I'm going to go to the dependencies tab because there's, there's several tabs here, go to the dependencies tab and then these are the various libraries that my project is currently using.

[00:03:51] Now, you can see here, I've already got a dependency on Gson. So, I actually don't need to add that. But let's pretend like that's, that's not already there.

[00:03:59] So to add the GSA dependency, I'm going to come up here and hit the plus button and I'm going to go to the library option and I'm going to select from Maven.

[00:04:10] Now Maven is, as we described earlier, a tool for managing dependencies in Java projects, but it's also an online repository of Java libraries.

Start visual description. The professor searches for the Gson library in the Maven repository, selects the appropriate version, and adds it to the project, explaining the process of managing dependencies in Java projects. End visual description.

[00:04:20] So even though this project is not a Maven project, I still want to take advantage of the Maven library repository on the internet.

[00:04:28] So that's why I'm going to say from maven here.

[00:04:30] So I want to download the Gson library from that, that website essentially.

[00:04:35] And then to find the Gson library on there, I'm just going to search for Gson and hit the search button and then it brings up a bunch of matches here and I'm going to go to the one that I know I want, which is the uh the real Gson library here, which is from Google.

[00:04:59] So I'm going to use Dean 10 or maybe it's got a 10.1 here.

[00:05:05] I use 10.1.

[00:05:08] And so once I've selected that library, then I would just say, OK.

[00:05:12] And then that would go ahead and add the library to my project.

[00:05:16] If this library has any dependencies of its own, it would recursively download those libraries as well.

[00:05:21] And so all I have to do is hit, OK? And I get exactly what I want.

[00:05:24] And so that's how to add a dependency if you just have a regular intelligent project.

[00:05:33] Now, on the slide here. It does reference another um It tells you also how to add a dependency. If you, if you're using Gray O Great O is another tool similar to Maven that manages dependencies.

[00:05:47] If you have a Great O project, then you would want to paste um this line down here at the very bottom into your, your build dot Gradel file.

[00:05:55] But uh you're probably not using Gradel in this class. So, you're probably going to do one of the first two.

[00:06:00] But the project we provide is actually a maven project.

[00:06:03] So you'll probably do number two.