Dependencies Tab

Use the **Dependencies** tab of the Module page to configure the module dependencies.

Item	Description
Module SDK	Select the module SDK. (To associate the project SDK with the module, select **Project SDK**. Note that if you change the project SDK later, the module SDK will change accordingly.)
	If the desired SDK is not present in the list, click **New** and select the necessary SDK type. Then, in the dialog that opens, select the SDK home directory and click **OK**.
	To view or edit the name and contents of the selected SDK, click **Edit**. (The SDK page will open.)

Item	Description
Table of dependencies	This table shows all dependencies of the selected module.

Export		If this check box is selected, a library is included in the compilation classpath of the dependent modules, as well as the classpath of the module which it is attached to. If this option is not selected, a library is only included in the compilation classpath of a module which it is attached to. A library is always included in the runtime classpath regardless of the state of the **Export** check box.
Scope		This check box lets you control the classpath 🔗 for your module sources and test sources at the build and run phases. ■ **Compile** The dependency is included in the classpath for your sources and test sources at the compilation and run phases. ■ **Test** The dependency is included in the classpath only for your test sources at the compilation and run phases. ■ **Runtime** The dependency is included in the classpath for your sources and test sources but only at the run phase. ■ **Provided** For your sources, the dependency is included in the classpath only at the compilation phase. This is useful when there is a container (e.g. a web container 🔗 of an application server) that provides the corresponding dependency at runtime. For your test sources, however, the dependency is included in the classpath both at the build and run phases.

Item			Description
	✚	Alt+Insert	Use this icon or shortcut to add a library or a module to the list of the module dependencies. Select one of the following options:
			▪ **Jars or directories.** Select this option to create a new module (Java) library. In the dialog that opens, select the necessary files and folders. These may be individual `.class`, `.java`, `.jar` and `.zip` files, and the directories containing such files.
			▪ **Library.** Select this option to use one or more of the existing project or global libraries. (If necessary, you can also choose to create a new module, project or global library.)
			> The following is available only in the Ultimate edition of IntelliJ IDEA
			You can also select the libraries belonging to the application servers registered in IntelliJ IDEA. (To register an application server, use the Application Servers page of the **Settings** dialog.)
			▪ **Module Dependency.** Select this option to specify the modules the current module should depend on. See Configuring Module Dependencies and Libraries for details.
Dependency storage format			Use this drop-down list to select the format for storing dependencies (as an IntelliJ IDEA module, or as Eclipse project). This option is helpful for the teams that use different development tools.
	━	Alt+Delete	Use this icon or shortcut to remove the selected item or items from the list of dependencies.
	⬆⬇	Alt+Up or Alt+Down	Use these icons or shortcuts to move the selected item one line up or down in the list of dependencies. See Processing Order.
	🖉	F4	Use this icon to open the Configure Library dialog to edit the library.

Context menu

Item	Description
Edit	Choose this command to configure the selected module or project library.
Remove	Choose this command to delete the selected dependency from the list.
Navigate	Choose this command to open the selected archive or SDK in the corresponding pages of the Project Structure dialog box.
Find Usages	Choose this command to show the list of modules where the selected dependency is used. Clicking one of the modules from this list results in opening this module in the Module Page of the Project Structure dialog box.
Analyze this dependency	Choose this command to perform analysis and display results in the Dependency Viewer.
Move to project libraries Move to global libraries	Choose one of these commands to move the selected dependency to the list of project or global libraries accordingly. After applying, these commands disappear from the menu, and **Copy to module libraries** command becomes available.

Item	Description
Copy to module libraries	Once a library is moved to the project or global libraries, it can be copied to the module libraries.

See Also

Concepts:

■ Dependencies

Procedures:

■ Configuring Module Dependencies and Libraries

Reference:

■ SDKs

■ Application Servers

■ Configure Library Dialog

■ Project Library and Global Library Pages

Web Resources:

■ Developer Community ⊕