

Synchronizing and Sharing Settings

In this section:

- [Synchronizing settings](#)
- [Prerequisite](#)
- [Basics](#)
- [What is synched?](#)
- [Connecting to IntelliJ Configuration Server](#)
- [Configuring login to IntelliJ Configuration Server](#)
- [Sharing settings](#)

Synchronizing settings

Prerequisite

Before you start working with IntelliJ Configuration Server, make sure that **IntelliJ Configuration Server** plugin is downloaded and enabled, as described in the section [Installing, Updating and Uninstalling Plugins](#).

Basics

IntelliJ IDEA makes it possible to synchronize settings across several different IntelliJ IDEA installations, or auto-configure new installations, using settings from the server.

Once logged in, the settings are synchronized with the server based on "updated" timestamp stored in each settings group. The most recent ones are used and are either sent to server or applied locally from the server.

IntelliJ IDEA always works with settings stored locally. So, the local settings are used, if, at some moment, the server is not available. On the following connection all updated settings will be pushed to the server.

When logged in to the server, once some settings are updated locally inside IntelliJ IDEA, they are subsequently sent to the server. But local settings are only updated on IntelliJ IDEA startup.

The server stores only one set of settings for each user.

What is synched?




IntelliJ Configuration Server stores almost all of the IDE and project settings, except for some platform-specific ones, and those containing local paths. This includes code style settings, keymaps, fonts and colors scheme, inspection profiles, and others.

Connecting to IntelliJ configuration server

You can connect to the *IntelliJ Configuration Server* in various ways:

- During the *first* IntelliJ IDEA startup after installing the plugin, through the [Login to IntelliJ Configuration Server](#) dialog box.
- At any time [during a session](#) through the **Login to IntelliJ Configuration Server** dialog box.
- During IntelliJ IDEA startup according to the previously configured [settings](#):
 - Silently
 - Through the [Login to IntelliJ Configuration Server](#) dialog box


To connect to the IntelliJ configuration server during a session

1. At any time during a session, click the **IntelliJ Configuration Server Status** button ( when connected to the server, or , when disconnected) on the Status bar.
2. In the dialog box that opens, click the **Login** button.
3. Specify the login and password of your JetBrains Account or create a [JetBrains Account](#)  if you do not have it.

To connect to the IntelliJ Configuration Server through a Proxy server, select the **Use HTTP Proxy** check box and specify the Proxy server parameters in the dialog box controls. See a detailed description of controls in the [Login to IntelliJ Configuration Server](#) dialog box reference.

Configuring login to IntelliJ configuration server

To configure login to the server during the next IntelliJ IDEA startup

1. Click the **IntelliJ Configuration Server Status** button  on the Status bar.
2. In the dialog box that opens, specify whether you want to log in to the IntelliJ Configuration Server during the next startup and how you want to do it. The following options are available:
 - Show [login](#) dialog box
 - Login silently
 - Do not login

Select the desired option and click **OK**.

The specified configuration settings will be applied during the next IntelliJ IDEA startup.

Sharing settings

For sharing project settings, `.idea` project configuration directory should be shared via [version control](#).

If you decide to share IntelliJ IDEA project files with the other developers, follow these guidelines.

Directory based project format (`.idea` directory) is used by all the recent IntelliJ IDEA versions by default. Here is what you need to share:

- All the files under `.idea` directory in the project root, except the `workspace.xml` and `tasks.xml` files, which store user-specific settings.
- All the `.iml` module files that can be located in different module directories.

Be careful about sharing the following:

- Android artifacts that produce a signed build (will contain keystore passwords)
- `dataSources.ids`, `datasources.xml` (can contain database passwords)

You may consider not to share the following:

- `gradle.xml`.
- XML files under `.idea/libraries` in case they are generated from Gradle project.
- `user dictionaries` folder (to avoid conflicts if another developer has the same name).

For the legacy project format (`.ipr/.iml/.iws` files), share the project `.ipr` file and all the `.iml` module files, don't share the `.iws` file, since it stores user specific settings.

See Also

Concepts:

- [Project](#)

Procedures:

- [Version Control with IntelliJ IDEA](#)

Reference:

- [Project and IDE Settings](#)
- [Directories Used by IntelliJ IDEA to Store Settings, Caches, Plugins and Logs](#)
- [Login to IntelliJ Configuration Server Dialog](#)

External Links:

- [Support: How to manage projects under Version Control Systems](#) 

Web Resources:

- [Developer Community](#) 