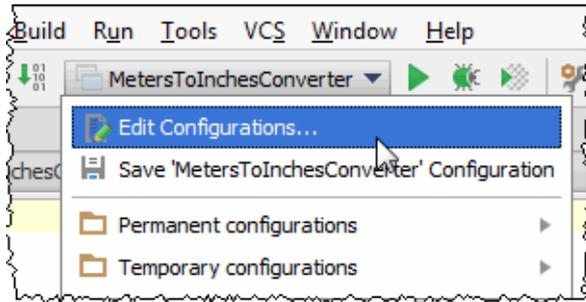


Creating and Editing Run/Debug Configurations

The available [run/debug configurations](#) are displayed in the drop-down list in the Run area of the main toolbar:



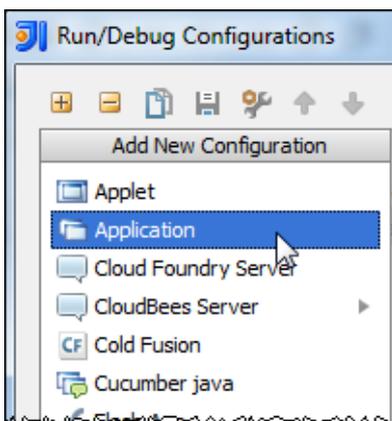
IntelliJ IDEA provides the [Run/Debug Configuration](#) dialog box as a tool for handling run/debug configurations: create configuration profiles or change the default ones.

IntelliJ IDEA suggests the following ways to create a run/debug configuration:

- [Create a run configuration manually](#) on the base of the default one, using the [Run/Debug Configuration](#) dialog box.
- [Save a temporary run configuration](#).
- Specify the [target application server during module creation](#). IntelliJ IDEA will generate a run/debug configuration of the corresponding type.

To create a run/debug configuration

1. Open the [Run/Debug Configuration](#) dialog box by doing one of the following:
 - On the main menu, choose **Run | Edit Configurations**.
 - Click the run/debug configurations selector on the main toolbar.
 - Press **Shift+Alt+F10**, then press **0** to display the **Edit Configuration** dialog box or select the configuration from the pop-up window and press **F4**.
2. In the [Run/Debug Configuration](#) dialog box, click **+** on the toolbar or press **Alt+Insert**. The drop-down list shows the default run/debug configurations. Select the desired configuration type.



The fields that appear in the right pane, display the default settings for the selected configuration type.

If you want to change the default run/debug configuration settings, expand the **Defaults** node, select the desired configuration type, and modify it as required.

3. For the new run/debug configuration:

- Specify its name in the **Name** text box. This name will be shown in the list of the available run/debug configurations.
- Specify whether you want to make IntelliJ IDEA check existence of the instances of the same run/debug configuration. If you want to make sure that only one instance of the run/debug configuration is currently executed, select the check box **Single instance only**. In this case, a confirmation dialog box will show up every time you try to launch run/debug configuration, when one instance of the same type is still running.

If you click **OK** in the confirmation dialog box, the first instance of the runner will be stopped, and the next one will take its place.

If this check box is not selected, you can launch as many instances of the runner as required. So doing, each runner will start in its own tab of the Run tool window.

To use an existing configuration as a pattern, create its copy by clicking the **Copy** button  on the toolbar, then change it as required.

4. In the **Before launch** section, define whether you want to compile the modified sources, and run an Ant or Maven script.
5. In the **Configuration** tab, [specify the class](#) that contains the `main()` method, VM options, program arguments, working directory and other configuration-specific settings.
6. In the **Logs** tab, specify the options that control [output logs](#) produced while running or debugging the application.

In particular, specify whether IntelliJ IDEA will display the [standard output and standard error output](#) to the console.
7. For the various servers, set up the deployment and startup/connection options.
8. For the applications and tests, click the **Code Coverage** tab (for example, [in applications run/debug configuration](#)), and specify the options that define [code coverage measurement](#) for testing purposes.
9. Specify additional parameters depending on the configuration type. Refer to the descriptions of run/debug configuration parameters below the [Run/Debug Configurations](#) section.
10. Apply the changes and close the dialog box.

See Also

Concepts:

- [Run/Debug Configuration](#)
- [Artifact](#)

Procedures:

- [Setting Configuration Options](#)
- [Configuring Code Coverage Measurement](#)
- [Setting Log Options](#)
- [Configuring Artifacts](#)

Reference:

- [Run/Debug Configurations](#)

Web Resources:

- [Developer Community](#) 

