

Enabling JSF Support

To support the [JavaServer Faces framework](#) (JSF), IntelliJ IDEA provides:

- The Java EE: Java Server Faces [plugin](#). This plugin is bundled with the IDE and enabled by default. However, it's always worth making sure that this plugin is still enabled before you start developing a JSF application.
- A dedicated [facet](#) type (JSF). A JSF facet provides facilities for developing JSF applications, and lets you set up an appropriate [module](#) structure as well as corresponding deployment descriptors and configuration files. It also lets you download JSF implementation libraries (e.g. [Apache MyFaces](#) or [Mojarra](#)) and JSF component libraries such as [RichFaces](#), [ICEfaces](#), [PrimeFaces](#) and [OpenFaces](#).

A JSF facet is an extension to a Web facet, therefore a Web facet should be created first.

On this page:

- [Making sure that the Java Server Faces plugin is enabled](#)
- [Creating a module with a JSF facet](#)
- [Adding a JSF facet to an existing module](#)

Making sure that the Java Server Faces plugin is enabled

1. [Open the Settings dialog](#) (e.g. `Ctrl+Alt+S`).
2. In the left-hand part of the dialog, under **IDE Settings**, select **Plugins**.
3. In the right-hand part of the dialog, on the [Plugins page](#), type `faces` in the search box. As a result, only the plugins whose names and descriptions contain `faces` are shown in the list of plugins.
4. If the check box to the left of **Java EE: Java Server Faces** is not selected, select it.
5. Click **OK** in the **Settings** dialog.
6. If suggested, restart IntelliJ IDEA.

Creating a module with a JSF facet

Depending on the situation, you can choose to create a new project with a Java module or to add a Java module to an existing project.

1. Do one of the following:
 - If you are going to create a new project: click **Create New Project** on the [Welcome screen](#) or select **File | New Project**.
As a result, the [New Project wizard](#) opens.
 - If you are going to add a module to an existing project: [open the project](#) you want to add a module to, and select **File | New Module**.
As a result, the [New Module wizard](#) opens.
2. On the first page of the wizard, in the left-hand pane, select **Java Module**.
3. In the right-hand part of the page, specify the associated settings and click **Next**. (See the descriptions of the settings [for a new project](#) or [for an existing project](#).)
4. On the next page of the wizard (the [Technologies page](#)), select the **Web Application** check box. Select the version of the Servlet specification to be supported from the **Versions** list. If you want the deployment descriptor `web.xml` file to be created, select the **Create web.xml** check box.
5. Select the **JSF** check box. If you want the configuration file `faces-config.xml` to be created, select the **Create faces-config.xml** check box.

6. Select the required library option and, if necessary, specify the associated settings. You can choose to:

- Download and use a JSF implementation library.

To do that, under **Libraries**, select **Download**.

Now, to view or modify the associated options, click **Configure**, and in the **Downloading Options** dialog that opens:

- Select the JSF version that the library should implement.
- Specify the library name.
- Select the **library level** (global, project, or module).
- Under **Files to download**, select which of the files you want to download.
- Under **Copy downloaded files to**, specify the path to the destination folder. If you want to change the default path, click  and specify the folder location in the [dialog that opens](#).
- Use a JSF library IntelliJ IDEA is already aware of.

To do that, click **Use library** and select the required library from the list.

If necessary, configure the library settings (for example, change its name). This is done in the **Edit Library** dialog which you can open by clicking **Configure**.

The contents of the **Use library** list may be different depending on the situation:

- If you are creating a new project, this list contains the JSF libraries already defined in IntelliJ IDEA as [global libraries](#).
- If you are adding a new module to an existing project, this list, in addition, contains the project-level libraries of the current project.

- Create a new library using the appropriate JAR files available on your computer.

To do that, click **Use library** and then click **Create**. Select the required JAR files in the [dialog that opens](#). (Keep the Ctrl key pressed to perform multiple selection.)

If necessary, configure the new library (for example, change its name or **level**). To do that, click **Configure** and specify the required settings in the **Create Library** dialog.

- Postpone setting up the library until a later time. In this case, select **Set up library later**.

7. If you are going to use a JSF component library or libraries (e.g. PrimeFaces, RichFaces, etc.), select the corresponding check box or check boxes under **JSF** and specify the associated options. The procedure is similar to that for the JSF implementation library.

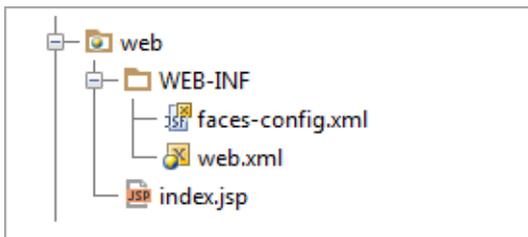
8. If, at this step, you are ready to specify the Web server you are going to deploy your application to, do so. This will result in the corresponding server-specific [run/debug configuration](#) for your module generated automatically. Otherwise, to be able to run your application, you will have to create the run/debug configuration yourself.

9. Click **Finish**.

As a result, IntelliJ IDEA does the following:

- Creates the module structure with the `web` and `WEB-INF` nodes (directories).
- In the `web` directory, creates the `index.xhtml` file that you can use as the starting page of your application.

- In the WEB-INF directory, creates:
 - web.xml, the Web application deployment descriptor.
 - faces-config.xml, the JSF configuration file.



- If specified, creates a JSF library or libraries at the corresponding level and adds the library or libraries to the [module dependencies](#).
- Creates an [artifact](#) specification for you module.
- If you have specified the server, IntelliJ IDEA also creates a [run/debug configuration](#).

Adding a JSF facet to an existing module

1. [Open the module settings](#).
2. With the desired module selected, click **+** on the toolbar, and, under **Framework**, select **Web**.
3. If necessary, [configure the Web facet resources and settings](#).
4. Select the Web facet you have just created, click **+** and select **JSF**.
5. Now, to download a JSF implementation library and add it to [module dependencies](#), in the lower part of the Facet 'JSF' page that opens in the right-hand part of the **Project Structure** dialog, click **Fix** next to the message reading *JSF library not found in the module dependencies list*.
6. In the dialog that opens, specify the library options as necessary and click **OK**.
7. If necessary, use the [JSF Facet page](#) to change the **Facelet** support option.
8. To add a JSF component library (e.g. PrimeFaces), click **+** again, and, under **Framework**, select the library of interest (e.g. **Primefaces**).
9. In the dialog that opens, specify the necessary library options and click **OK**.

See Also

Concepts:

- [Facet](#)
- [Run/Debug Configuration](#)
- [Artifact](#)
- [JSF](#)
- [Available Facets and Their Dependencies](#)

Procedures:

- [Adding Modules to a Project](#)
- [Creating a Project from Scratch](#)
- [Configuring Project and Global Libraries](#)
- [Enabling Web Application Support](#)

Reference:

- [JSF Facet Page](#)

External Links:

- [Developing JSF applications in IntelliJ IDEA](#)□

Web Resources:

- [Developer Community](#)□