

# Enabling Hibernate Support

---

IntelliJ IDEA provides a dedicated *Hibernate facet* to support the [Hibernate](#) technology. If you need to enable Hibernate in a module, you have to extend this module with the Hibernate facet.

Only one Hibernate facet is allowed per module.

This section considers enabling Hibernate support:

- In a [new](#) module.
- In an [existing](#) module.

## To enable Hibernate support in a new module

### 1. [Create a new Java module from scratch.](#)

On the first page of the wizard, select **Java Module**, not **Web Module**. (The Web module type is for developing web applications using programming languages other than Java, for example, [PHP](#), or [JavaScript](#), or [markup languages](#).)

### 2. On the **Technologies** page of the wizard, select the **Hibernate** check box. Enable the other technologies as required.

### 3. To have IntelliJ IDEA create the Hibernate configuration file `hibernate.cfg.xml`, and the class with the `main()` method automatically, select the **Create default Hibernate configuration and main class** check box.

### 4. To have a database schema imported automatically, select the **Import database schema** check box.

### 5. Specify where to download the Hibernate libraries to. By default, IntelliJ IDEA downloads the libraries to the `lib` directory of the project root. To have the libraries downloaded to another location or to use existing libraries, click the **Pick from disk** button.

### 6. Click **Finish**.

### 7. If you have selected the **Import database schema** check box, specify the data source parameters in the [Import Database Schema](#) dialog box that opens. IntelliJ IDEA creates entity beans for each table detected in the data source and fields for each column in a table.

## To enable Hibernate support in an existing module

1. Open the [Module Settings dialog box](#).
2. With the desired module selected, click **+** on the toolbar and select **Hibernate** from the list of facets available for this module.

As a result, a new Hibernate facet is created. Its settings are shown on the [Hibernate Facet page](#) in the right-hand part of the dialog.

3. Configure the libraries required for Hibernate. If you have not [configured libraries](#) in advance, the facet page displays the list of missing libraries. To resolve the problem, click the **Fix** button. In the **Specify Libraries** dialog box, specify whether you would like to use one of the existing libraries, or download the archive. You can control the target location where the archive will be placed, the library name, and the level on which the library will be created.
4. Create descriptors of the type `hibernate.cfg.xml`. To do that, click **+** (**Alt+Insert**) in the **Descriptors** section. In the dialog box that opens, specify the descriptor name and location. Apply changes.
5. In the **Data Sources Mapping** section, specify the data source for each Hibernate session factory. To do that, select the desired entry in the **Session Factory** column, double click the corresponding **Data Source**, and select one from the list of data sources [already configured](#) in your workspace.
6. Apply the changes and click **Finish**.

## See Also

### Concepts:

- [Object-Relational Mapping \(EJB, Hibernate and JPA\)](#)
- [Data Sources](#)

### Procedures:

- [Creating and Importing Data Sources](#)
- [Generating Persistence Mappings](#)

### Reference:

- [Database Tool Window](#)
- [Hibernate and JPA Facet Pages](#)

### Web Resources:

- [Developer Community](#) 