Liferay IDE , Workspace setup and Development

[1. Download Developer Studio SDK 2](#_Toc23918718)

[2. Installing Dev Studio (installing blade CLI) 2](#_Toc23918719)

[3. Creating workspace using liferay IDE: 2](#_Toc23918720)

[4. Creating Modules with Work studio : 2](#_Toc23918721)

[5. Workspace structures: 2](#_Toc23918722)

[6. portal-ext-properties: 4](#_Toc23918723)

[7. Start the server: 5](#_Toc23918724)

[8. Application development: 5](#_Toc23918725)

[8.1 Creating MVC portlet 5](#_Toc23918726)

[8.2 Creating Module 5](#_Toc23918727)

# Download Developer Studio SDK

<https://sourceforge.net/projects/lportal/files/Liferay%20IDE/3.7.1/>

# Installing Dev Studio (installing blade CLI)

<https://portal.liferay.dev/docs/7-1/tutorials/-/knowledge_base/t/installing-blade-cli>

# 3. Creating workspace using liferay IDE:

<https://portal.liferay.dev/docs/7-1/tutorials/-/knowledge_base/t/creating-a-liferay-workspace-with-liferay-ide>

# 4. Creating Modules with Work studio :

<https://portal.liferay.dev/docs/7-1/tutorials/-/knowledge_base/t/creating-modules-with-liferay-ide>

# Workspace structures:

**configs**: holds the configuration files for different environments. These files serve as your global configuration files for all Liferay Portal servers and projects residing in your workspace. To learn more about using the configs folder, see the Testing Modules section.

**gradle**: holds the Gradle Wrapper used by your workspace.

**modules**: holds your custom modules. This can also hold frontend portlets created with the Liferay JS Toolkit

**plugins**-**sdk** (generated): holds plugins to migrate from previous releases. These project types should eventually be migrated to the wars folder, if possible. This is targeted for Liferay Portal 7.0 to provide a way to migrate from the Plugins SDK to Liferay Workspace. See the Using a Plugins SDK from Your Workspace section for more information. The Plugins SDK is no longer available for Liferay Portal CE 7.1.

**themes**: holds Node.js-style themes that use the Liferay JS Theme Toolkit, which are built using the Liferay Theme Generator.

**wars**: holds traditional WAR-style web application projects and theme projects (i.e., generated by the theme project template).

**build**.**gradle**: the common Gradle build file.

**gradle**.**properties**: specifies the workspace’s project locations and Liferay Portal server configuration globally.

**gradle**-**local**.**properties**: sets user-specific properties for your workspace. This lets multiple users use a single workspace, letting them configure specific properties for the workspace on their own machine.

**gradlew**: executes the Gradle command wrapper.

**settings**.**gradle**: applies plugins to the workspace and configures its dependencies.

# portal-ext-properties:

**DB Configuration**:

Below how to configure Liferay with Oracle database.

1. Downloading the jdbc Driver

2. Go the official Oracle site, accept license agreement and download the **ojdbc8**.jar jdbc Driver

3. Copy it and past it under $**TOMCAT**\_**HOME**/**lib**/**ext**

4. Creating the **blank** database

5. Create a blank database called **lportal**.

6. Updating portal-ext.properties

# **Oracle**

jdbc.default.driverClassName=oracle.jdbc.driver.OracleDriver

jdbc.default.url=jdbc:oracle:thin:@localhost:1521:xe

jdbc.default.username=lportal

jdbc.default.password=lportal

# Start the server:

Day-3

1. **Liferay user – page – site Creation – and Admin Activities**
2. Web content
3. Document
4. Page
5. Site
6. Control panel admin

# Application development:

* 1. Creating MVC portlet  
       
     <https://portal.liferay.dev/docs/6-2/tutorials/-/knowledge_base/t/creating-a-liferay-mvc-portlet-project>

## 8.2 Creating Module