

Build Plugin Source Code using Docker

- [Generate a Plugin](#)
- [Build a Plugin](#)

If you have [Docker](#) (or a compatible container engine) installed, use the [docker-maven-joget](#) image to quickly generate and build plugins with minimal pre-requisites.

To use this simplified approach, you just need to run some commands in a terminal (Mac/Linux) or PowerShell (Windows).



NOTE: \${PWD} is for Mac, Linux and Windows PowerShell. For Windows Command Prompt, replace \${PWD} with "%cd%".



This approach is suitable to quickly get started. For long-term development, you should set up a proper Java development environment as described in [Build Source Code on Windows](#) or [Build Source Code on Linux](#).

Generate a Plugin

1. Run the following command to generate a plugin project (adjust the **groupId** and **artifactId** values as required)

```
docker run -it --rm -v ${PWD}:/usr/src/mymaven -w /usr/src/mymaven jogetworkflow/docker-maven-joget:8.0-SNAPSHOT mvn archetype:generate -DarchetypeGroupId="org.joget" -DarchetypeArtifactId="wflow-plugin-archetype" -DarchetypeVersion="8.0-SNAPSHOT" -DgroupId="org.joget.plugin" -DartifactId="custom-plugin"
```

Build a Plugin

1. Navigate to the plugin source code directory/folder containing the pom.xml file

2. Run the following command to clean and build the project

```
docker run -it --rm -v ${PWD}:/root/.m2 -v ${PWD}:/usr/src/mymaven -w /usr/src/mymaven jogetworkflow/docker-maven-joget:8.0-SNAPSHOT mvn clean install
```