

Introducing Eclipse Plug-ins 2

Contents

- OSGi
- MANIFEST.MF
- plugin.xml

OSGi

- OSGi is a Java-based framework targeted for use by systems that require long running times, dynamic updates, and minimal disruptions to the running environment.
- An Eclipse Plug-in is an OSGi Bundle with some distinct Eclipse specifics.

MANIFEST.MF

```
Manifest-Version: 1.0
Bundle-ManifestVersion: 2
Bundle-Name: HelloPlugin Plug-in
Bundle-SymbolicName: org.eclipse.course.helloplugin; singleton:=true
Bundle-Version: 1.0.0
Bundle-ClassPath: helloplugin.jar
Bundle-Activator: org.eclipse.course.helloplugin.Activator
Bundle-Vendor: Eclipse Tutorial
Bundle-Localization: plugin
Require-Bundle: org.eclipse.ui, org.eclipse.core.runtime
Eclipse-LazyStart: true
```

MANIFEST.MF

- Manifest-Version: 1.0
- Bundle-ManifestVersion: 2

- Header defined by OSGi R4 (used by Eclipse).

MANIFEST.MF

- Bundle-Name: HelloPlugin Plug-in
- Bundle-SymbolicName:
org.eclipse.course.helloplugin; *singleton:=true*

- The name of the bundle, a human-readable text.
- The plug-in id, uniquely identify the plug-in and is typically constructed using Java package naming conventions.

MANIFEST.MF

- Bundle-Version: 1.0.0

- The version of the bundle, major version number, minor version number, and service level.

MANIFEST.MF

- Bundle-ClassPath: helloplugin.jar

- This property specifies the CLASSPATH to use for the bundle. The property may contain references to directories or jar files inside the bundle jar file.

MANIFEST.MF

- Bundle-Activator:
org.eclipse.course.helloplugin.Activator
- This class is used to start and stop the bundle.

MANIFEST.MF

- Bundle-Vendor: Eclipse Tutorial
- Bundle-Localization: plugin
- The bundle vendor in a human-readable text.
- The location for the localization data, it defaults to plugin (i.e. plugin.properties).

MANIFEST.MF

- Require-Bundle: org.eclipse.ui,
org.eclipse.core.runtime
- Eclipse-LazyStart: true
- This property specifies which bundles and their exported packages to import for use in the given bundle.
- Makes the plug-in to be lazily loaded by Eclipse.

MANIFEST.MF

- If a plug-in has been successfully compiled and built but, during execution, throws a `NoClassDefFoundError`, it may indicate that the plug-in project's Java classpath is out of sync with the `Require-Bundle` declaration in the `MANIFEST.MF` file.
- It is important to keep the classpath and the `Require-Bundle` declaration in sync.

plugin.xml

```
<?xml version="1.0" encoding="UTF-8"?>
<?eclipse version="3.2"?>
<plugin>

  <extension point="org.eclipse.ui.actionSets">
    ...
  </extension>

</plugin>
```

plugin.xml

- A plug-in can provide an extension to be used by other plug-ins.
- It needs to create an extension point schema and declares it in plugin.xml.
- But usually we create plug-ins to extend Eclipse functionalities.

plugin.xml

- We use `org.eclipse.ui.actionSets` to provide a new action (button or menu).
- `<extension point="org.eclipse.ui.actionSets"> ... </extension>`

plugin.xml

- `actionSet`: This element is used to define a group of actions and/or menus.
- `label`: a translatable name used by the Workbench to represent this action set to the user.
- `visible`: an optional attribute indicating whether the action set is initially visible when a perspective is open.
- `id`: a unique identifier for this action set.

plugin.xml

- **menu:** This element is used to define a new menu.
- **label:** a translatable name used by the Workbench for this new menu. The name should include mnemonic information.
- **id:** a unique identifier that can be used to reference this menu.

plugin.xml

- **action:** This element defines an action that the user can invoke in the UI.
- **label:** a translatable name used either as the menu item text or toolbar button label. The name can include mnemonic information.
- **icon:** a relative path of an icon used to visually represent the action in its context.

plugin.xml

- **class:** a fully qualified name of a class which implements `org.eclipse.ui.IWorkbenchWindowActionDelegate` or `org.eclipse.ui.IWorkbenchWindowPulldownDelegate`.
- **tooltip:** a translatable text representing the action's tool tip. Only used if the action appears in the toolbar.
- **id:** unique identifier used as a reference for this action.

plugin.xml

- **menubarPath:** a slash-delimited path ('/') used to specify the location of this action in the menu bar.
- **toolbarPath:** a slash-delimited path ('/') that is used to specify the location of this action in the toolbar.