



EBook Gratuito

APPENDIMENTO eclipse-plugin

Free unaffiliated eBook created from
Stack Overflow contributors.

#eclipse-
plugin

Sommario

Di.....	1
Capitolo 1: Introduzione a eclipse-plugin.....	2
Osservazioni.....	2
Examples.....	2
Installazione o configurazione.....	2
Ciao mondo.....	4
Titoli di coda.....	14

You can share this PDF with anyone you feel could benefit from it, downloaded the latest version from: [eclipse-plugin](#)

It is an unofficial and free eclipse-plugin ebook created for educational purposes. All the content is extracted from [Stack Overflow Documentation](#), which is written by many hardworking individuals at Stack Overflow. It is neither affiliated with Stack Overflow nor official eclipse-plugin.

The content is released under Creative Commons BY-SA, and the list of contributors to each chapter are provided in the credits section at the end of this book. Images may be copyright of their respective owners unless otherwise specified. All trademarks and registered trademarks are the property of their respective company owners.

Use the content presented in this book at your own risk; it is not guaranteed to be correct nor accurate, please send your feedback and corrections to info@zzzprojects.com

Capitolo 1: Introduzione a eclipse-plugin

Osservazioni

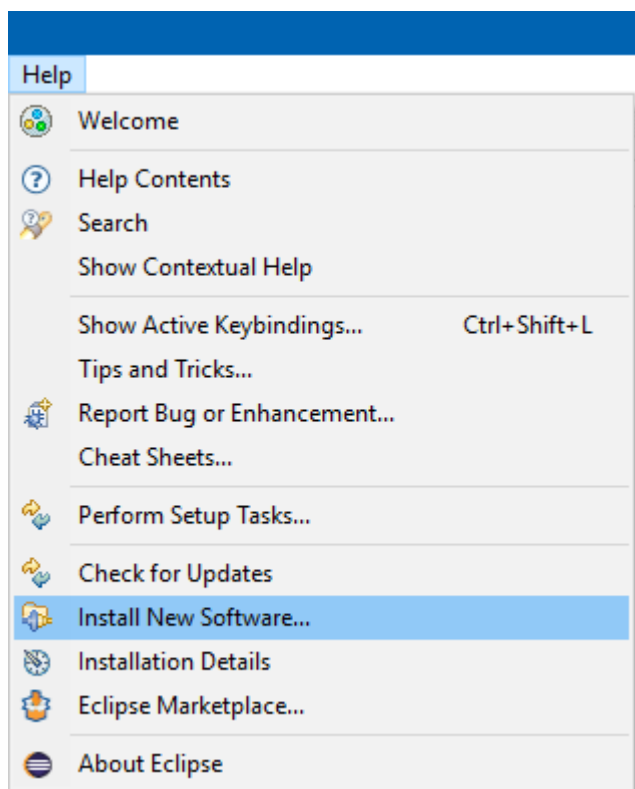
Questa sezione fornisce una panoramica di cosa sia eclipse-plugin e perché uno sviluppatore potrebbe volerlo utilizzare.

Dovrebbe anche menzionare qualsiasi argomento di grandi dimensioni all'interno di eclipse-plugin e collegarsi agli argomenti correlati. Poiché la documentazione di eclipse-plugin è nuova, potrebbe essere necessario creare versioni iniziali di tali argomenti correlati.

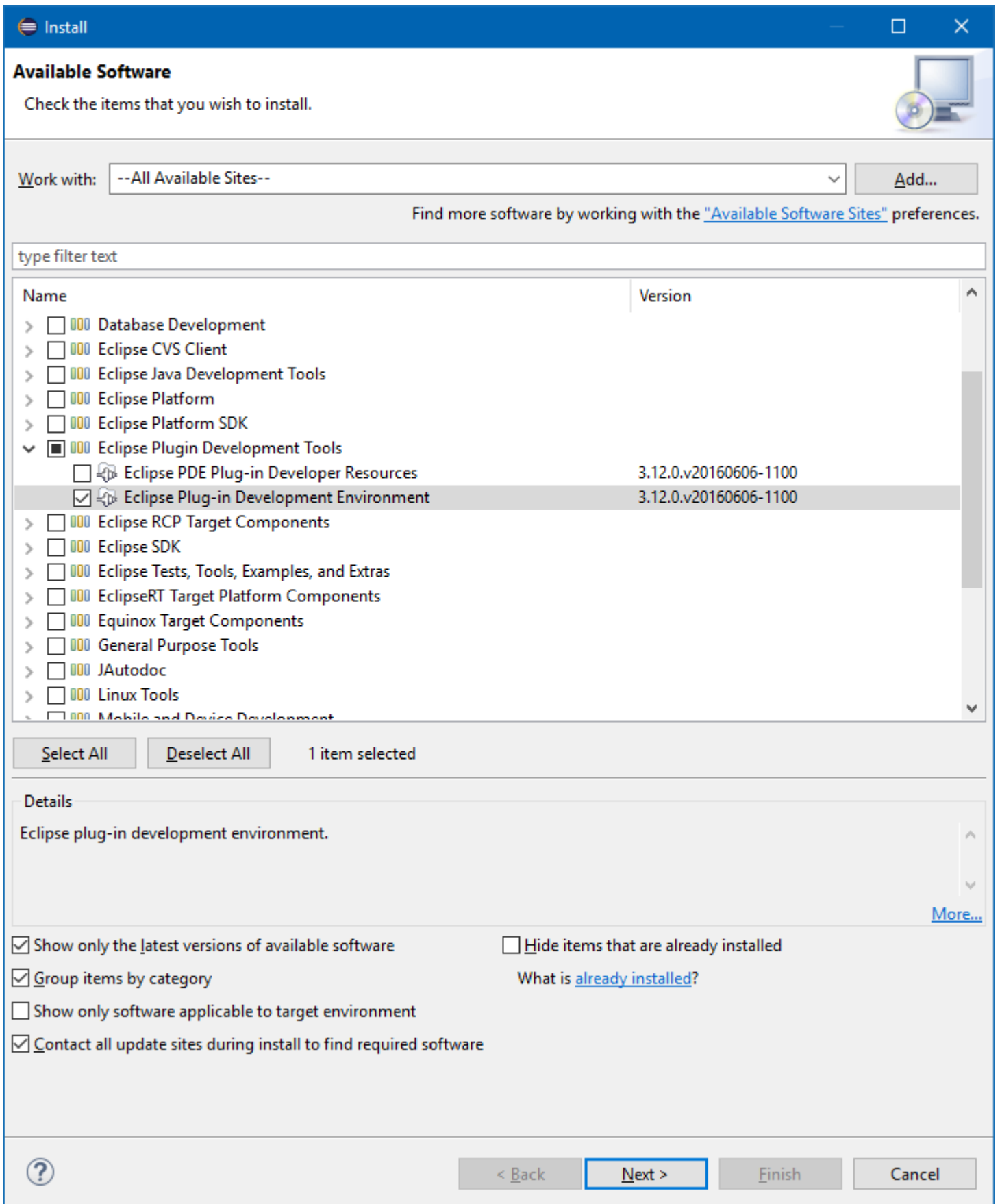
Examples

Installazione o configurazione

Supponendo di avere [Eclipse IDE per gli sviluppatori Java](#) installati, avviare Eclipse, fare clic su " Guida " -> " *Installa nuovo software ...* "



Seleziona " *--Tutti i siti disponibili--* " in " *Lavora con:* " e vai a " *Eclipse Plugin Development Tools* ". Seleziona " *Eclipse Plug-in Development Environment* " spuntando la casella di controllo di fronte ad esso.

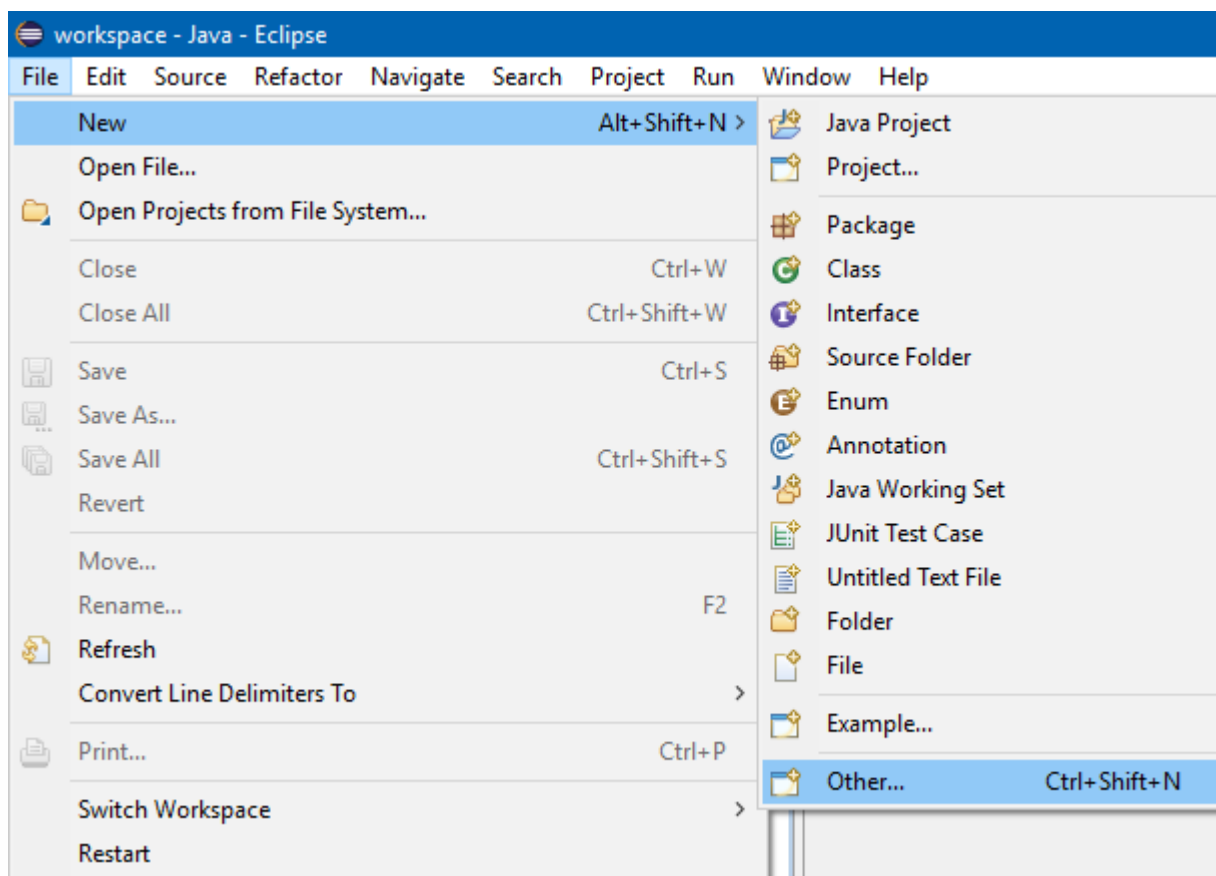


Fare clic su "Avanti" per consentire a Eclipse di verificare eventuali dipendenze necessarie. Fare nuovamente clic su "Avanti" per avviare l'installazione.

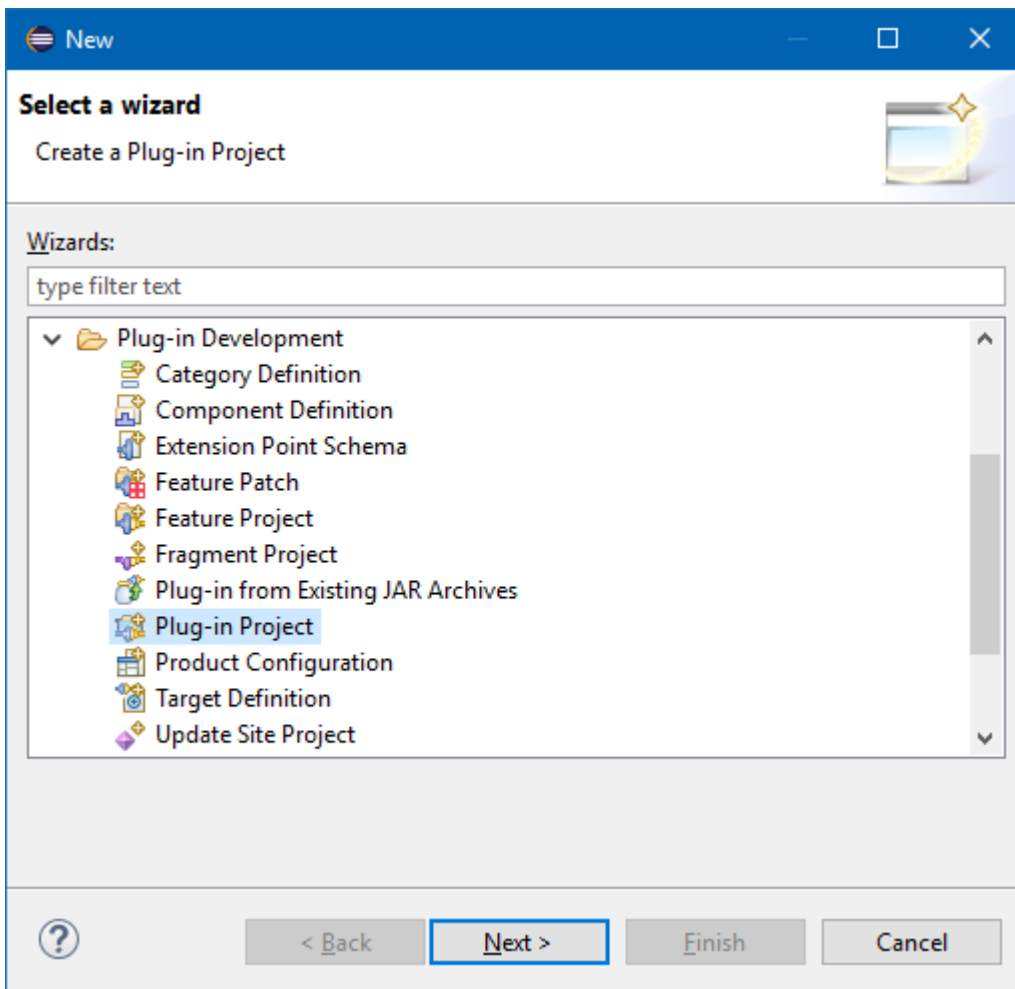
Al termine, riavvia Eclipse.

Ciao mondo

Per creare un plug-in Hello World per Eclipse, fai clic su: *File* → *Nuovo* → *Altro ...*

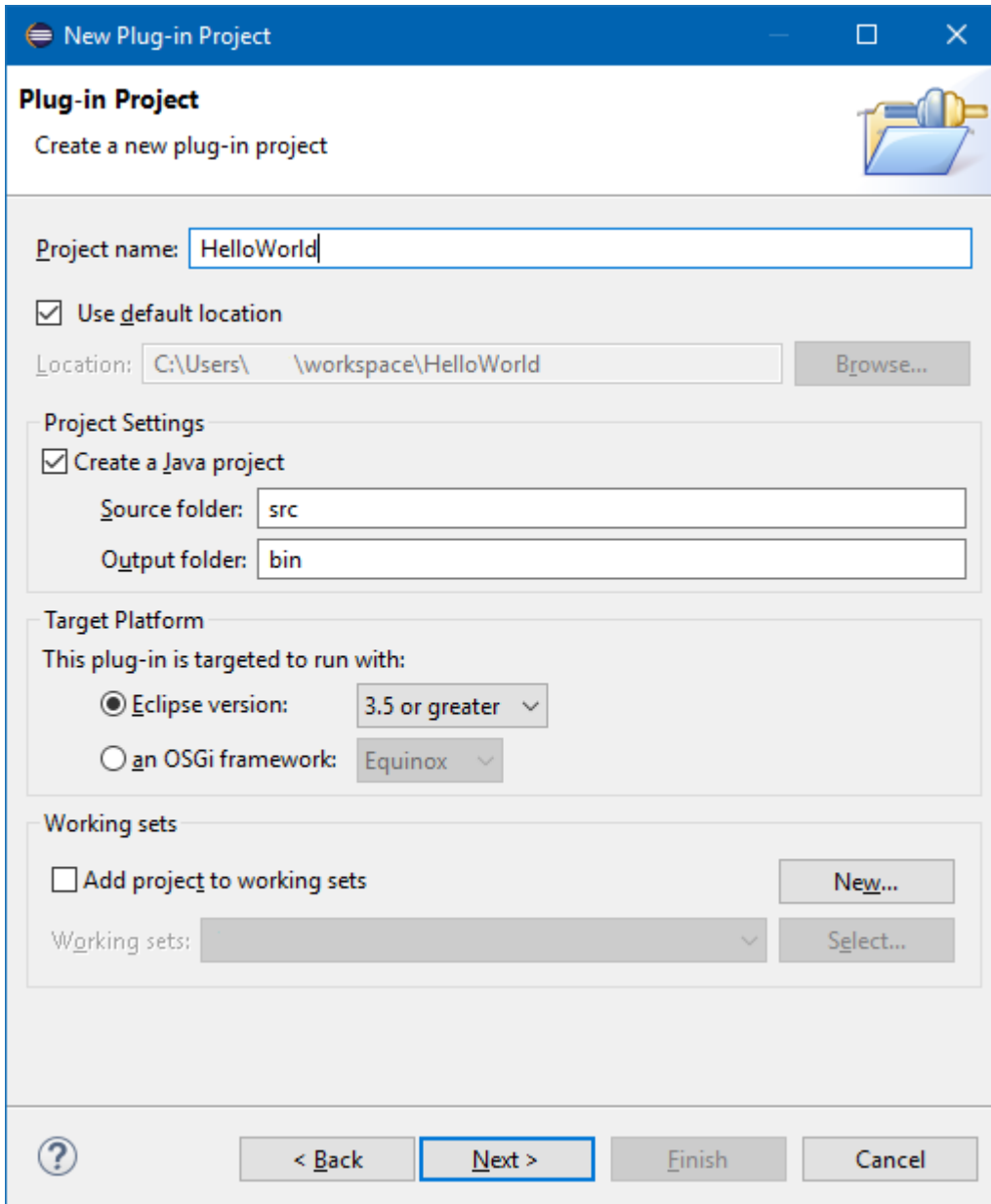


Seleziona *Progetto plug-in* e fai clic su *Avanti*>



La procedura guidata *Nuovo progetto plug-in* ti guiderà attraverso le opzioni per la creazione di un nuovo plug-in.

Inserisci un nome per il progetto (come HelloWorld) e fai clic su *Avanti*>



Nella pagina *Contenuto* , è possibile impostare l' *ID* , la *versione* , il *nome* e il *fornitore* del plug-in.

La versione sarà *1.0.0.qualifier* per impostazione predefinita. Puoi lasciare questo così com'è, ma è meglio cambiarlo in qualcosa di significativo. Il [wiki di eclipse](#) consiglia una sintassi come *vYYYYMMDD* (anno, mese giorno).

New Plug-in Project

Content
Enter the data required to generate the plug-in.

Properties

ID: HelloWorld

Version: 1.0.0.201701010000

Name: HelloWorld

Vendor: HelloWorldVendor

Execution Environment: JavaSE-1.8

Options

Generate an activator, a Java class that controls the plug-in's life cycle

Activator: helloworld.Activator

This plug-in will make contributions to the UI

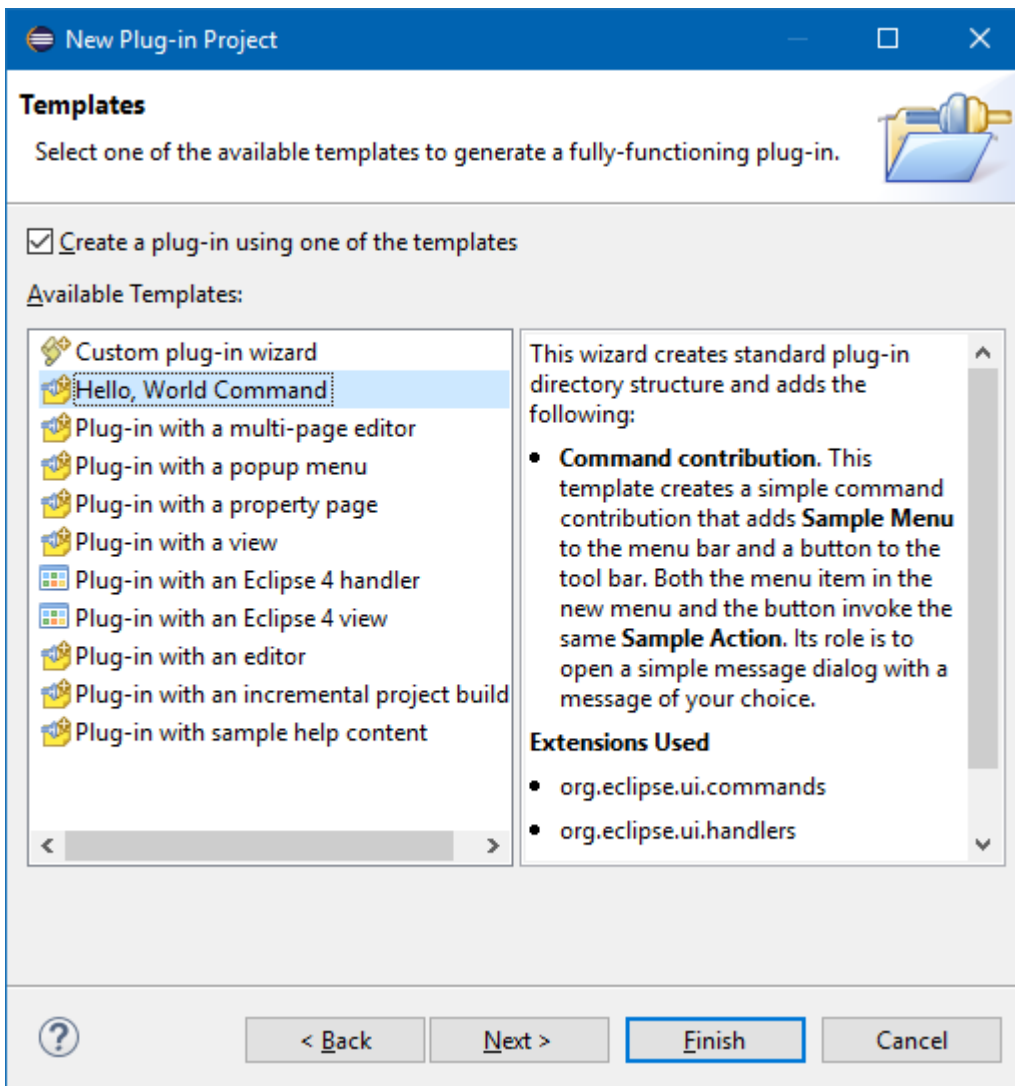
Enable API analysis

Rich Client Application

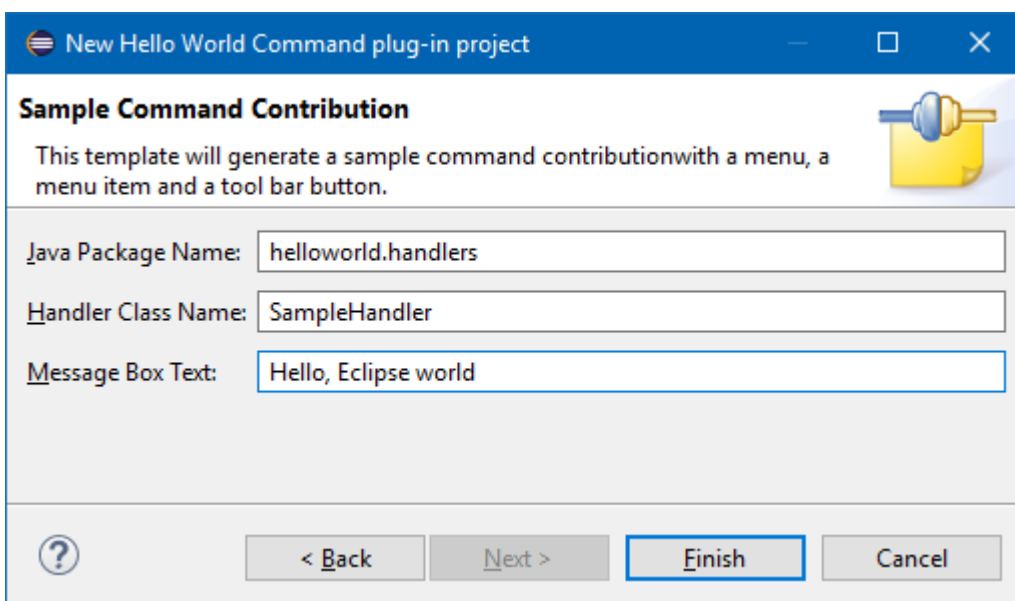
Would you like to create a rich client application? Yes No

? < Back Next > Finish Cancel

Nella pagina *Modelli* , puoi scegliere di creare il plug-in da qualsiasi modello selezionandolo e facendo clic su *Avanti*> . In alternativa è possibile combinare questi modelli scegliendo la *procedura guidata Custom plug-in* o per creare un nuovo plug-in senza modello deselezionando la casella di controllo davanti a *Crea un plug-in utilizzando uno dei modelli* .

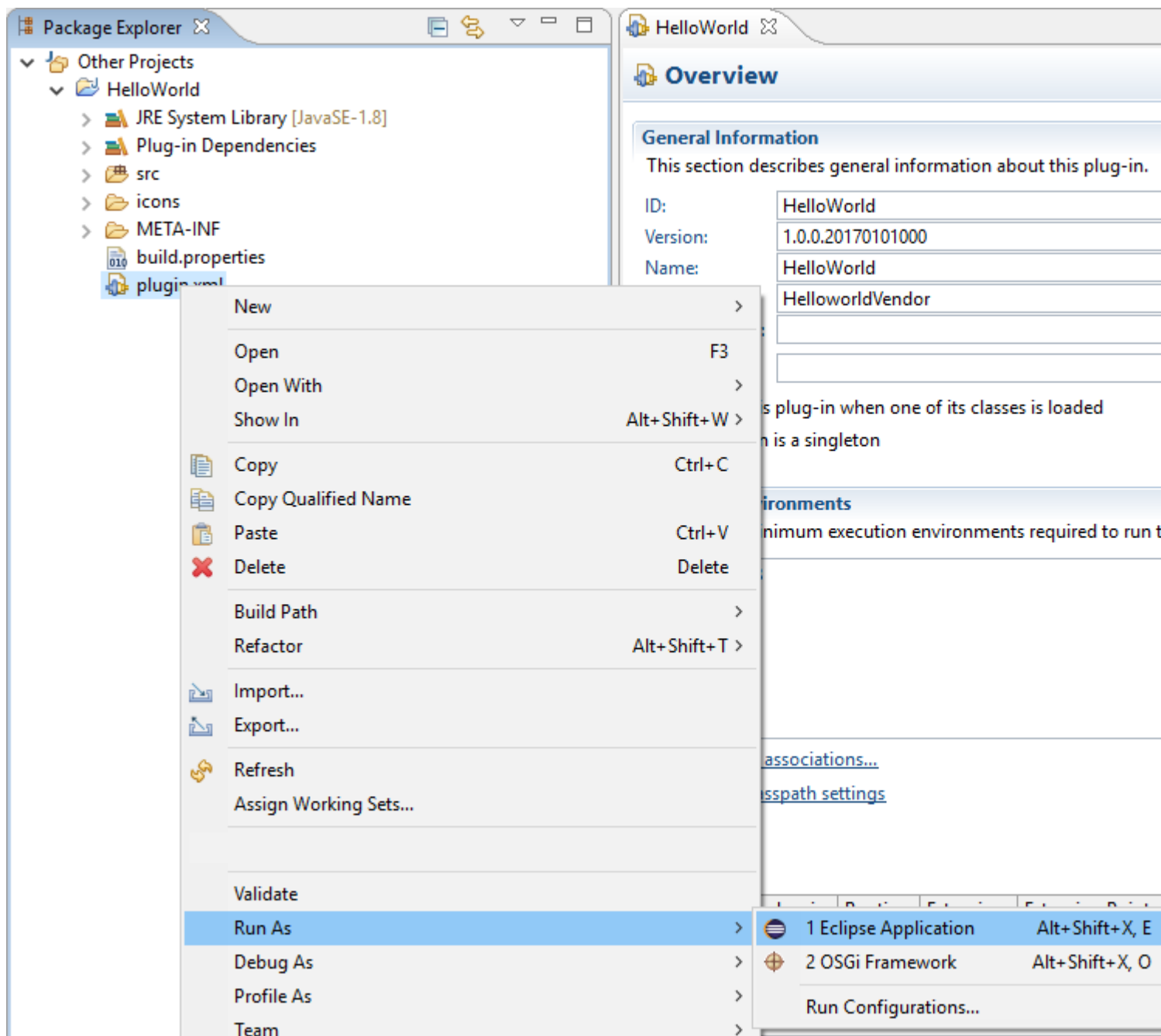


Per il modello *Hello, World Command*, ci sono altre impostazioni: il nome del pacchetto, il nome della classe Handler e il testo per la finestra del messaggio.



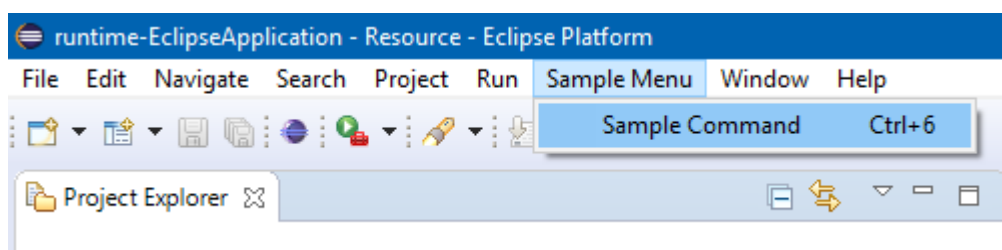
Una volta creato il plug-in, puoi eseguirlo facendo clic con il pulsante destro del mouse su *plugin.xml* → *Esegui come* → *Applicazione Eclipse*

Questo lancerà una nuova istanza di Eclipse (con un proprio spazio di lavoro) che avrà il tuo plug-in caricato.



Questo plug-in *Hello World* ha fornito 3 contributi alla GUI di Eclipse:

1. Un menu di esempio (con comando di esempio):



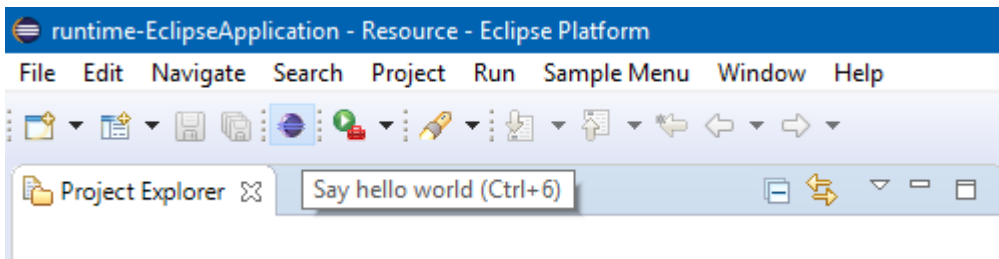
plugin.xml:

```

<extension
  point="org.eclipse.ui.menu">
  <menuContribution
    locationURI="menu:org.eclipse.ui.main.menu?after=additions">
    <menu
      label="Sample Menu"
      mnemonic="M"
      id="HelloWorld.menus.sampleMenu">
      <command
        commandId="HelloWorld.commands.sampleCommand"
        mnemonic="S"
        id="HelloWorld.menus.sampleCommand">
      </command>
    </menu>
  </menuContribution>
</extension>

```

2. Un'icona della barra degli strumenti:



plugin.xml:

```

<extension
  point="org.eclipse.ui.menu">
  <menuContribution
    locationURI="toolbar:org.eclipse.ui.main.toolbar?after=additions">
    <toolbar
      id="HelloWorld.toolbars.sampleToolbar">
      <command
        commandId="HelloWorld.commands.sampleCommand"
        icon="icons/sample.gif"
        tooltip="Say hello world"
        id="HelloWorld.toolbars.sampleCommand">
      </command>
    </toolbar>
  </menuContribution>
</extension>

```

3. Una scorciatoia da tastiera (Ctrl + 6)

plugin.xml:

```

<extension
  point="org.eclipse.ui.bindings">
  <key
    commandId="HelloWorld.commands.sampleCommand"
    contextId="org.eclipse.ui.contexts.window"
    sequence="M1+6"
    schemeId="org.eclipse.ui.defaultAcceleratorConfiguration">
  </key>

```

```
</extension>
```

Quando si attiva uno di questi 3, verrà eseguita la classe Handler:

plugin.xml:

```
<extension
  point="org.eclipse.ui.commands">
  <category
    name="Sample Category"
    id="HelloWorld.commands.category">
  </category>
  <command
    name="Sample Command"
    categoryId="HelloWorld.commands.category"
    id="HelloWorld.commands.sampleCommand">
  </command>
</extension>
<extension
  point="org.eclipse.ui.handlers">
  <handler
    commandId="HelloWorld.commands.sampleCommand"
    class="helloworld.handlers.SampleHandler">
  </handler>
</extension>
```

SampleHandler.java:

```
package helloworld.handlers;

import org.eclipse.core.commands.AbstractHandler;
import org.eclipse.core.commands.ExecutionEvent;
import org.eclipse.core.commands.ExecutionException;
import org.eclipse.ui.IWorkbenchWindow;
import org.eclipse.ui.handlers.HandlerUtil;
import org.eclipse.jface.dialogs.MessageDialog;

/**
 * Our sample handler extends AbstractHandler, an IHandler base class.
 * @see org.eclipse.core.commands.IHandler
 * @see org.eclipse.core.commands.AbstractHandler
 */
public class SampleHandler extends AbstractHandler {

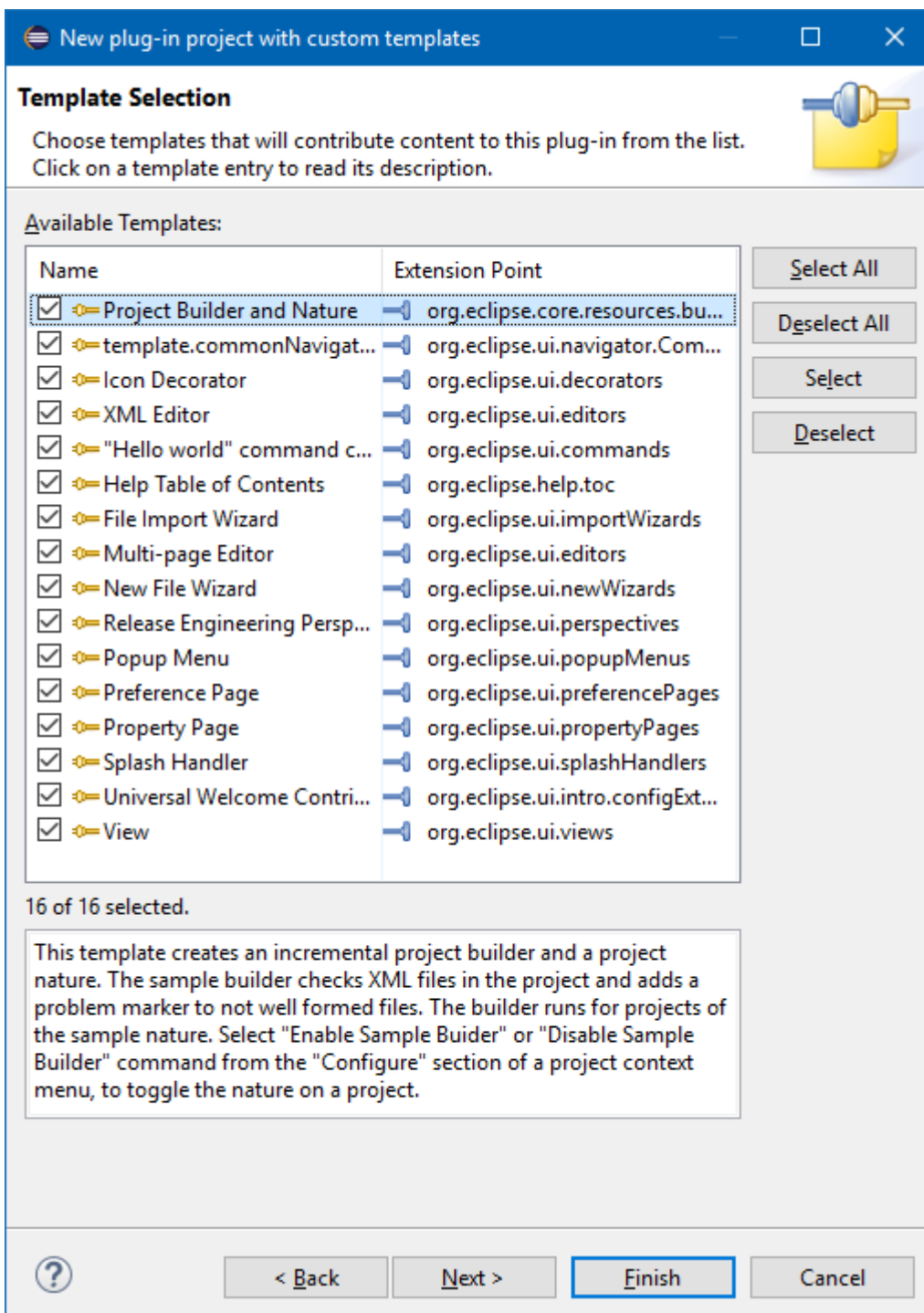
  @Override
  public Object execute(ExecutionEvent event) throws ExecutionException {
    IWorkbenchWindow window = HandlerUtil.getActiveWorkbenchWindowChecked(event);
    MessageDialog.openInformation(
      window.getShell(),
      "HelloWorld",
      "Hello, Eclipse world");
    return null;
  }
}
```

Quando viene eseguita la classe Handler, MessageBox mostrerà:



Questo è tutto ciò che fa il plug-in Hello World.

Se vuoi creare un plug-in con più funzionalità, potresti aver scelto un modello che meglio si adatta alle tue necessità o creare un plug-in tramite la *procedura guidata Custom plug-in* per combinare questi modelli:



Leggi Introduzione a eclipse-plugin online: <https://riptutorial.com/it/eclipse-plugin/topic/4982/introduzione-a-eclipse-plugin>

Titoli di coda

S. No	Capitoli	Contributors
1	Introduzione a eclipse-plugin	Community , KompjoeFrick