**开发一个简单的插件**

本节我们将动手开发一个简单的 Hello World 插件，该插件将向 Eclipse 的工

作台（Workbench）中添加一个新的菜单“Hello”，这个菜单将包含“Say Hello”

子菜单，点击这个子菜单，将显示一个“Hello World!”对话框。

**搭建插件开发环境**

在 Eclipse 中进行插件开发，需要插件开发环境（PDE），您可以从 Eclipse 网

站（http://www.eclipse.org/downloads/）上下载带有插件开发环境的 Eclipse 版本，

如：Eclipse Classic 版。EOS Studio 也包含了 PDE 开发环境，我们将以 EOS Studio

作为我们的插件开发环境。

(1). 获取并安装 EOS 6.X 版本的 Studio；

(2). 启动 Studio 后，选择**窗口打开透视图其他…**菜单项；

(3). 在弹出的“打开透视图”对话框中，选择“**插件开发**”，如下所示：

(4). 点击确定，系统将切换到插件开发透视图。

**新建一个插件项目**

一个插件对应一个插件项目，插件开发环境（PDE）为我们提供了新建插件

项目向导。新建一个插件项目的步骤如下：

(1). 选择**文件新建项目…**菜单项；

(2). 系统弹出“新建项目”向导，选择“**插件项目**”，点击“下一步”；

(3). 在“插件项目”向导页中，指定项目的名称：com.primeton.studio.hello，

点击“下一步”；

(4). 在“插件内容”向导页中，可以指定插件的一些基本属性，如：插件标

识、插件版本等，如下所示：



(5). 点击“完成”，向导将在您的工作空间中创建一个插件项目。

**插件清单编辑器**

在工作台左边的“包资源管理器”视图中，可以看到我们刚刚新建的插件项

目，选择 META-INF 目录下的 MANIFEST.MF 文件，右键选择“打开方式插件清

单编辑器”，系统将打开插件清单编辑器，如下所示：



插件清单编辑器主要用来编辑 MANIFEST.MF 和 plugin.xml 文件，主要的编辑

页说明如下。

“概述”页中显示了插件的基本信息，如：插件标识、版本、激活器等。

“依赖性”页中显示了插件所依赖的其它插件以及导入的包，通过该页可以

为插件添加新的插件依赖。

“运行时”页中显示了插件可以供外部访问的包，以及插件的类路径。通常，

我们可以在“类路径”中添加插件所引用的第三方类库。

“扩展”页中显示了该插件所提供的扩展，在该页中，我们可以为插件定义

新的扩展。

“扩展点”页中显示了该插件定义的扩展点信息。

**添加一个扩展**

Eclipse 的工作台（Workbench）插件，为我们提供了 org.eclipse.ui.actionSets

扩展点，通过该扩展点可以很容易的在工作台中加入自己的菜单。添加一个菜单

的步骤如下：

**1. 添加 org.eclipse.ui.actionSets 扩展**

(1). 打开插件清单编辑器，选择“扩展”页；



(2). 在“所有扩展”列表中，选择“**添加…**”按钮，系统弹出新建扩展向导；

(3). 在扩展点过滤器中，输入：org.eclipse.ui.actionSets，如下所示：

(4). 点击“完成”，向导将在“所有扩展”列表中添加一个新的扩展；

(5). 选择新建的 org.eclipse.ui.actionSets 扩展，在右边将显示扩展的详细信息，

如下所示：

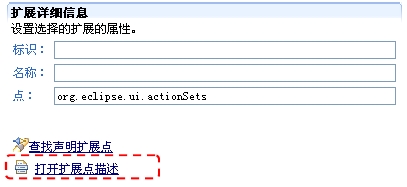
点击“打开扩展点描述”链接，将打开 org.eclipse.ui.actionSets 扩展点的

详细描述信息，这个信息将帮助我们很好的理解该扩展点的作用，以及

如何使用这个扩展点。

**2. 新建 actionSet**





(1). 在新建的 org.eclipse.ui.actionSets 扩展上点击右键，选择“**新建actionSet**”

菜单项；

(2). 编辑 actionSet 的属性，如下所示：

**3. 新建 menu**

(1). 在新建的 actionSet 上，右键选择“**新建menu**”菜单项；

(2). 编辑 menu 的属性，如下所示：

**4. 新建分组（separator）**

(1). 在新建的 menu 上，右键选择“新建separator”菜单项；

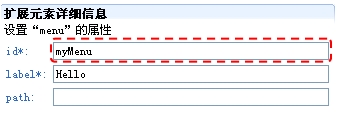
(2). 编辑 separator 的属性，如下所示：

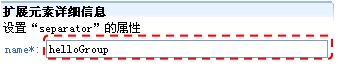
**5. 新建 action**

(1). 在新建的“MyActions”actionSet 上，右键选择“新建action”菜单项；

(2). 设置 action 的属性，如下所示：







其中 label 属性指定了 action 的名称；menubarPath 属性指定了该 action

将出现在菜单条的哪个位置上，这里的值为 myMenu/helloGroup，说明

了这个 action 将显示在我们刚才定义的 myMenu 菜单的 helloGroup 分组

中。

(3). 我们需要为 action 指定一个实现类，根据 org.eclipse.ui.actionSets 有关

action

的 描 述 信 息 ， 这 个 实 现 类 需 要 实 现

org.eclipse.ui.IWorkbenchWindowActionDelegate 接口。点击 hello action

中的 class 属性，如下所示：

系统将弹出 Java 类新建向导，指定包名和类名信息，如下所示：





点 击 “ 完 成 ”， 系 统 将 创 建 一 个 HelloAction 类 ， 该 类 实 现 了

org.eclipse.ui.IWorkbenchWindowActionDelegate 接口；

(4). HelloAction 类创建完毕后，系统将自动回填 hello action 的 class 属性值，

如下所示：

(5). 激活插件的插件清单编辑器，点击“保存”来保存上述配置的扩展信息。

**6. 实现 action 功能**

HelloAction 类执行时，将运行其中的 run()方法，我们编辑 HelloAction#run

方法，让它弹出一个对话框来显示“Hello World!”信息，HelloAction 类

如下所示：

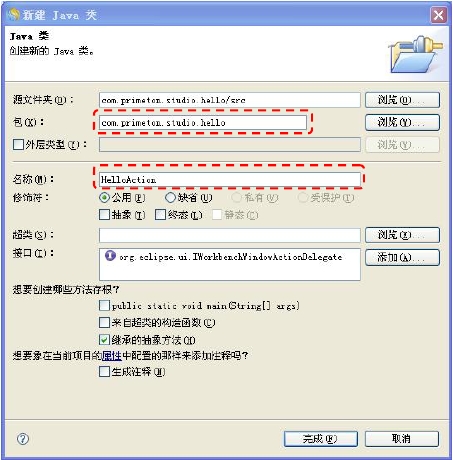
**package** com.primeton.studio.hello;

**import** org.eclipse.jface.action.IAction;

**import** org.eclipse.jface.dialogs.MessageDialog;

**import** org.eclipse.jface.viewers.ISelection;

**import** org.eclipse.ui.IWorkbenchWindow;





**import** org.eclipse.ui.IWorkbenchWindowActionDelegate;

**public class** HelloAction **implements** IWorkbenchWindowActionDelegate {

**private** IWorkbenchWindow window;

**public void** dispose() {

}

**public void** init(IWorkbenchWindow window) {

**this**.window = window;

}

**public void** run(IAction action) {

MessageDialog.*openInformation*(

window.getShell(),

"Hello",

"Hello World!\n欢迎进入插件开发的世界!");

}

**public void** selectionChanged(IAction action, ISelection selection) {

}

}

**运行插件项目**

经过以上步骤后，我们的第一个插件就开发完成了，现在我们来运行这个插

件项目，看看它的效果。运行插件的步骤如下：

(1). 选择“运行运行…”菜单项，弹出运行配置向导；

(2). 在左边的“运行类型”树上，选择“Eclipse 应用程序”，右键选择“新

建”菜单，将新建一个 Eclipse 应用程序运行项；

(3). 编辑相关信息，如：名称、工作空间位置、要运行的产品等信息，如下

所示：



(4). 编辑“自变量”信息，添加“程序自变量”和“VM 自变量”，如下所示：

之所以要编辑这些信息，是因为我们要运行的 Eclipse 应用程序包含了许

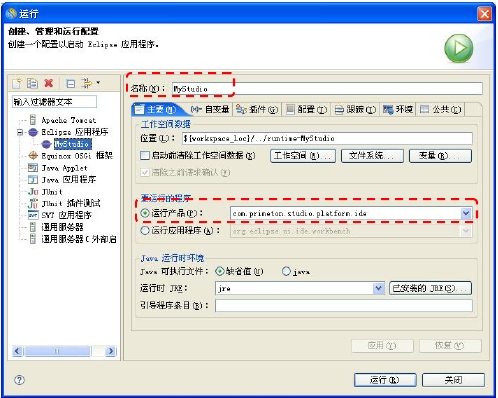
多插件，为了使我们的 Eclipse 应用程序平稳运行，我们调整了 Java 虚拟

机的内存。

(5). “插件”配置页中，显示了即将要运行的 Eclipse 应用程序所包含的插件，

可以看出，Eclipse 自动将我们工作空间中的插件加入到了插件列表中，

如下所示：





(6). 点击“应用”按钮，保存运行配置信息；

(7). 点击“运行”按钮，将启动一个新的 Eclipse 应用程序，它包含了我们所

创建的 hello 插件。启动完毕后，可以看到新启动的 EOS Studio 的菜单条

中包含了我们所添加的“hello”菜单项，点击“hello”菜单项，将显示

如下对话框：

至此，我们的第一个插件已经开发完成了，可以看出开发一个插件还是比较

简单的。同时也可以看出由于 Eclipse 本身的扩展性，使我们可以很容易的扩展

它，为它添加新的功能

