

## Capítulo 2

# Toma de contacto de Eclipse

### 1. La importancia de organizarse

Este capítulo permite familiarizarse con el entorno de desarrollo Eclipse.

Crear una aplicación implica modificar archivos de texto. Conforme se avanza en esta creación, los archivos se acumulan y resulta cada vez más importante organizarse con el objetivo de encontrar fácilmente el código.

Para ello, Java propone un sistema de organización por espacio de nombres o paquete, que permite agrupar distintas clases. Conceptualmente, es más fácil pensar en ello como distintas carpetas del equipo.

En el nivel organizativo más alto, Eclipse organiza el código de una aplicación en un proyecto. El proyecto agrupa todo el código necesario para realizar las funcionalidades solicitadas. Cada elemento de código se agrupa, a continuación, en diversas carpetas según su utilidad para la funcionalidad propiamente dicha o sus pruebas.

## 2. Primer proyecto

Para familiarizarse con el **workbench**, se crea una clase muy sencilla.

Primero debe crear un proyecto y después un package, toda clase de una aplicación que se considere profesional debe pertenecer a un package.

Como recordatorio, un package es físicamente una carpeta, y simbólicamente un espacio de nombre que permite reagrupar clases con la misma funcionalidad.

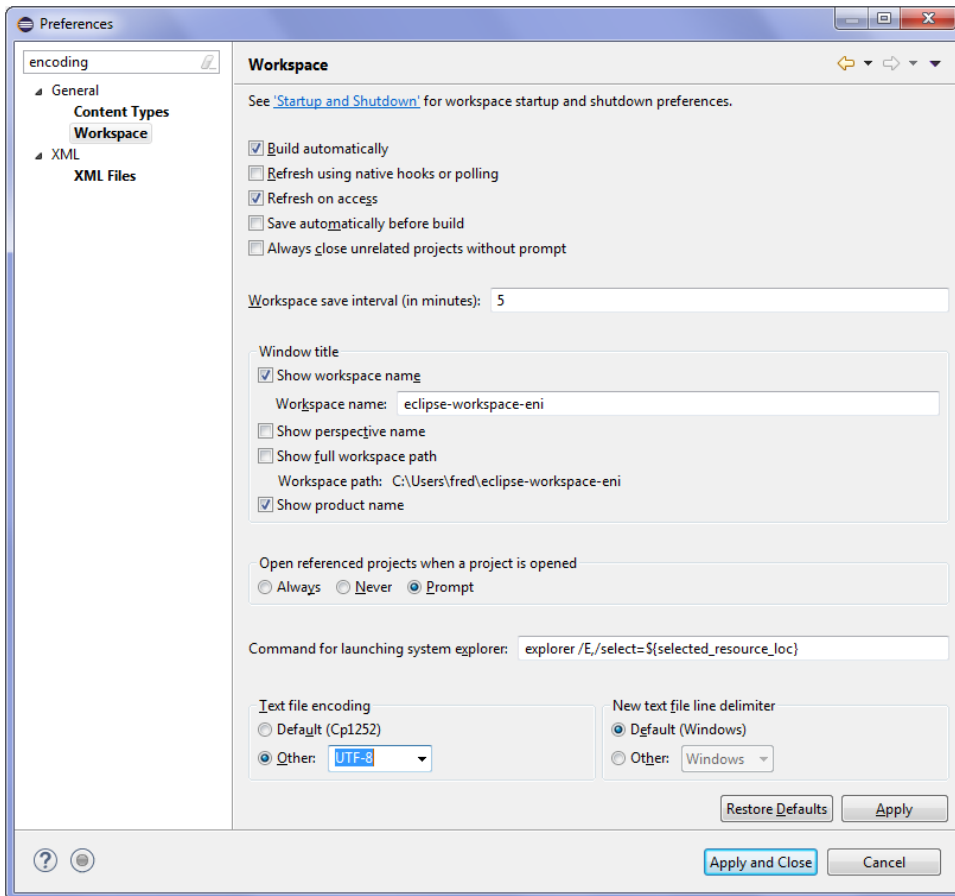
### ■ Observación

*Si no reagrupa sus clases en un package, Eclipse la guardará en el package por defecto. Está desaconsejado usar el package por defecto.*

También es la ocasión de proponer una estructura para los archivos del proyecto, con el fin de mejorar la legibilidad y testeabilidad de las aplicaciones.

Como preámbulo a este trabajo, y para facilitar la apertura de los archivos que provengan de sistemas operativos diferentes, es conveniente indicar la codificación de los archivos.

■ En el menú, elija **Window - Preferences**. Diríjase al apartado **General - Workspace** y elija el valor **Other - UTF-8** en la sección **Text file encoding**.

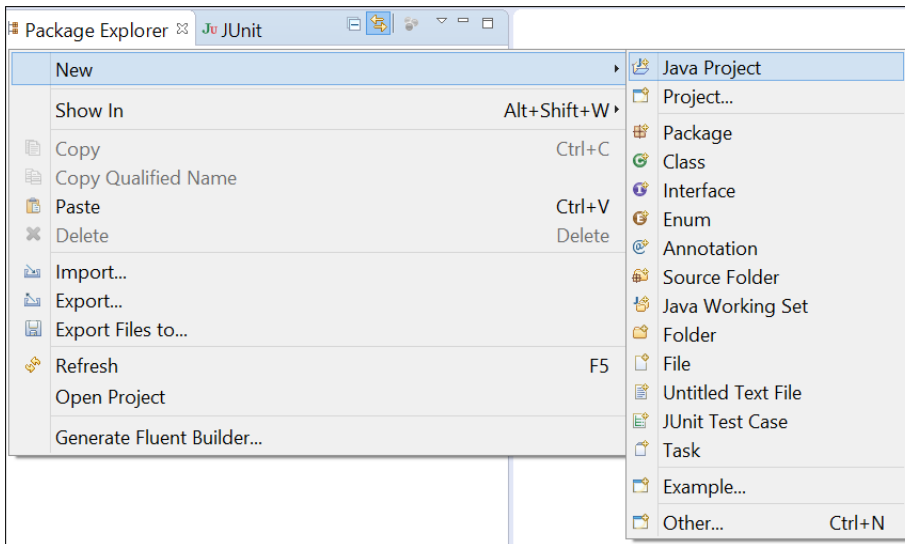


Este parámetro permite que los archivos con caracteres acentuados en Eclipse puedan leerse correctamente, ya sea en Windows, Linux o Mac.

### ■ Observación

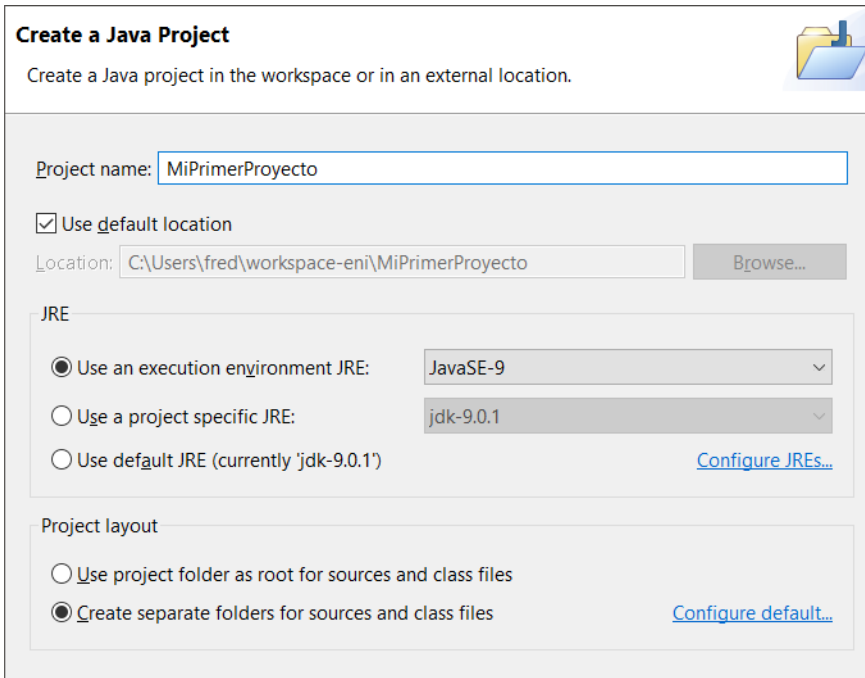
Los acentos así como los caracteres especiales están permitidos en los nombres de variables, pero no se recomienda su uso. Lo entenderá el día en el que abra su proyecto en un ordenador Linux, por ejemplo, o cuando tenga que trabajar en un proyecto con los nombres de variables en chino, japonés, hebreo o farsi.

- En el menú elija **File - New - Java Project**, o haga un clic derecho en la vista y después **New - Java Project**.



Aparece la ventana de diálogo de creación de proyecto. Permite dar un nombre al nuevo proyecto y elegir dónde guardarlo en el disco duro donde almacenaremos los archivos del proyecto (por defecto, los proyectos se almacenan en la carpeta del espacio de trabajo elegido durante la instalación de Eclipse).

- Introduzca el nombre del proyecto y haga clic en el enlace **Configure default** junto a la opción **Create separate folders for sources and class files**.



**Create a Java Project**

Create a Java project in the workspace or in an external location.

Project name:

☒ Use default location

Location:  [Browse...](#)

**JRE**

☒ Use an execution environment JRE:

☐ Use a project specific JRE:

☐ Use default JRE (currently 'jdk-9.0.1') [Configure JREs...](#)

**Project layout**

☐ Use project folder as root for sources and class files

☒ Create separate folders for sources and class files [Configure default...](#)

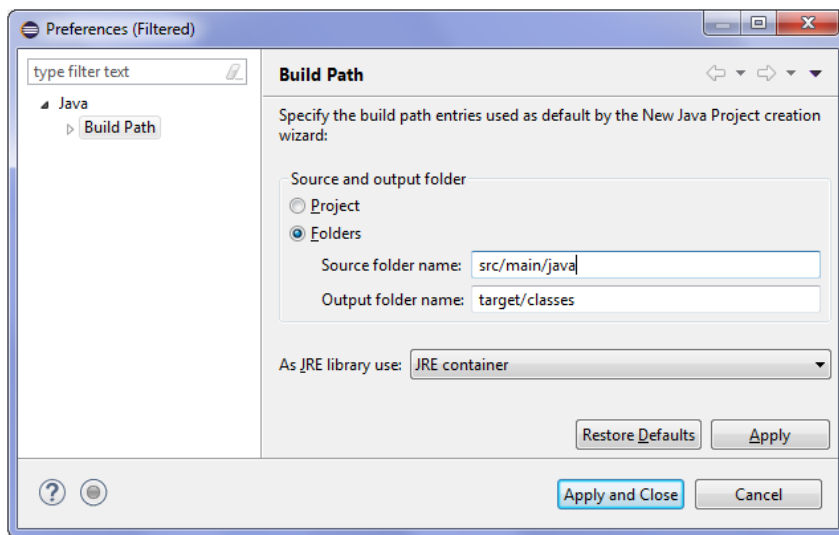
Esta acción permite definir las carpetas en las cuales guardaremos los archivos fuente y los archivos binarios compilados. De manera general, estos dos tipos de archivos deben estar separados. Esto permite por ejemplo copiar o eliminar fácilmente los binarios.

Los archivos fuente en un proyecto Java tienen muchas veces varios tipos: existen archivos fuente para las clases de la aplicación, archivos fuente para las clases de test, archivos de recursos tales como imágenes o configuraciones...

La siguiente acción permite definir las distintas carpetas para cada uno de estos tipos. Estas carpetas están inspiradas de los proyectos Maven (<http://maven.apache.org/>) y Gradle (<https://gradle.org/>), que son potentes herramientas para la gestión de proyectos informáticos, y permiten lanzar desde la línea de comandos o mediante una interfaz gráfica la compilación de archivos fuente en binarios, la realización de test unitarios, la transformación de clases en archivos jar, el empaquetado en un instalador y el despliegue en un servidor, por citar solamente algunas de las funcionalidades disponibles.

Maven aplica el refrán «cada cosa en su sitio, y un sitio para cada cosa» y define la carpeta **src/main/java** como la carpeta para los archivos fuente, **src/main/resources** para los archivos de configuración y las imágenes, **src/test/java** para los archivos fuente de test y **src/test/resources** para los archivos de configuración de los test. En cuanto a los archivos binarios, se guardan en la carpeta **target/classes**.

■ Modifique los nombres de las carpetas como en la siguiente imagen y valide.



Eclipse vuelve a la ventana de diálogo de creación del proyecto.

■ Haga clic en **Finish** para terminar la creación del proyecto.

El proyecto y su contenido son ahora visibles en la vista **Package Explorer**.

