

# Guía del Entorno de Desarrollo de Java. Eclipse

Dpto. Lenguajes y Ciencias de la Computación. E.T.S.I. Informática.  
Universidad de Málaga

Programación Orientada a Objetos

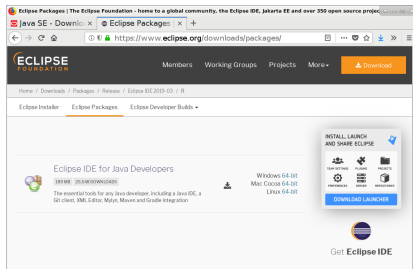
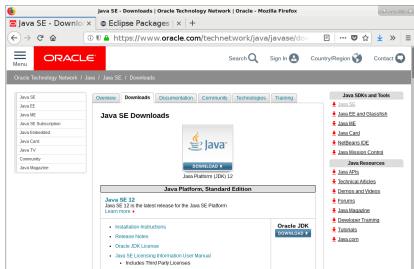
# Instalación del Entorno de Desarrollo Java Eclipse

- Descargue e instale el *kit de desarrollo Java SE JDK* de Oracle, desde la página web:

<https://www.oracle.com/java/technologies/javase-downloads.html>

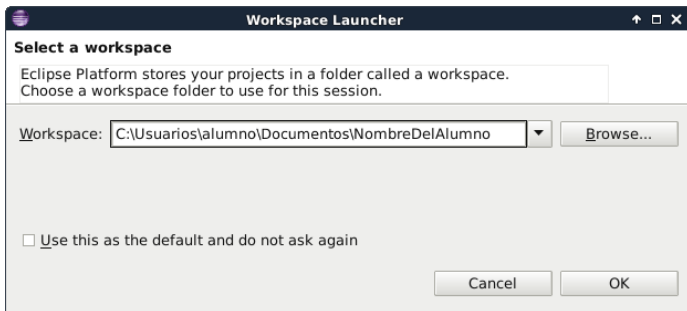
- Descargue e instale el *entorno de desarrollo Eclipse IDE for Java Developers*, desde la página web:

<https://www.eclipse.org/downloads/packages/>



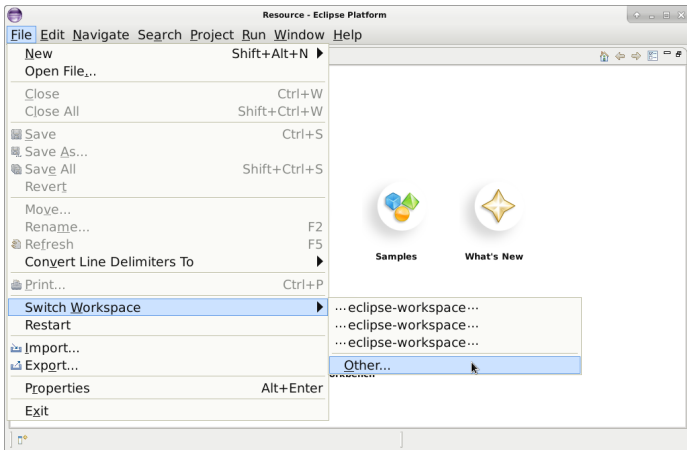
# El Espacio de Trabajo de Eclipse.

- Cuando comienza *Eclipse*, solicitará que se seleccione un *espacio de trabajo*. El alumno deberá utilizar **su propio nombre** dentro de la carpeta **Documentos** para el espacio de trabajo de eclipse.
- Es importante que el alumno **utilice su propio espacio de trabajo**, ya que a veces, el *espacio de trabajo* existente puede tener una **configuración inadecuada**.



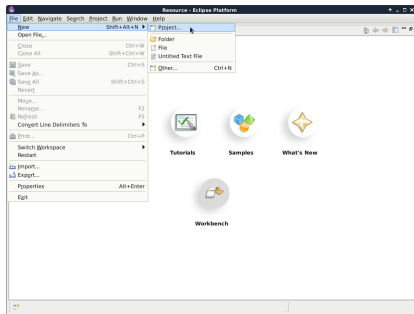
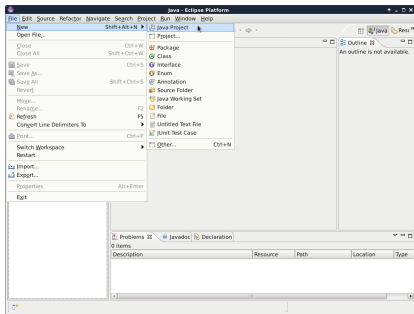
# Cambiar a otro Espacio de Trabajo.

- Es importante que el alumno **utilice su propio espacio de trabajo**, ya que a veces, el *espacio de trabajo* existente puede tener una **configuración inadecuada**.
  - Para seleccionar un *espacio de trabajo*. El alumno deberá utilizar **su propio nombre** dentro de la carpeta **Documentos** para el espacio de trabajo de eclipse.
- Una vez que hemos entrado en *Eclipse* con un determinado espacio de trabajo, a veces deseamos **cambiar a otro espacio de trabajo** distinto, para ello podemos utilizar la siguiente opción:



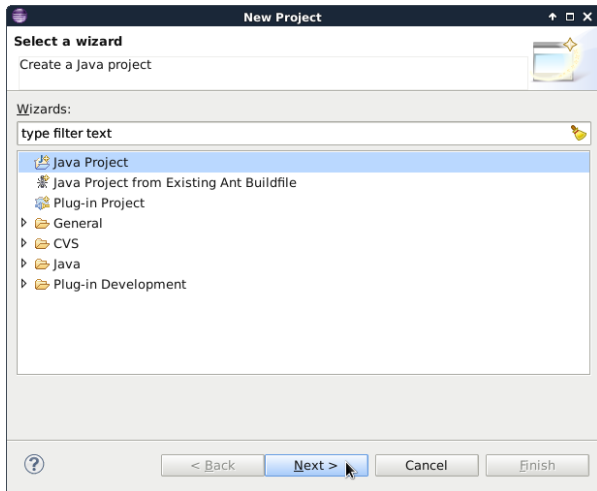
# Creación de un Nuevo Proyecto en Eclipse (I)

- Crearemos un proyecto Java nuevo seleccionando la opción File -> New -> Java Project si existe, o en otro caso, la opción File -> New -> Project



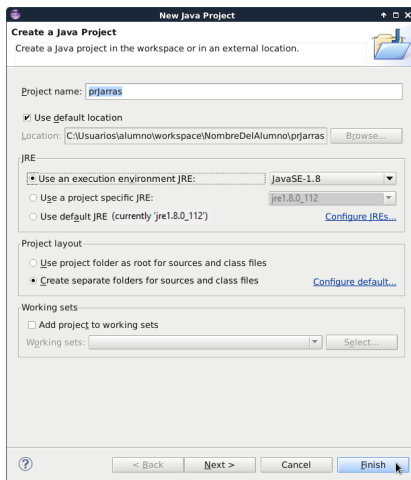
# Creación de un Nuevo Proyecto en Eclipse (II)

- En caso de ser necesario, seleccionamos Java Project como tipo de proyecto y pulsamos Next.



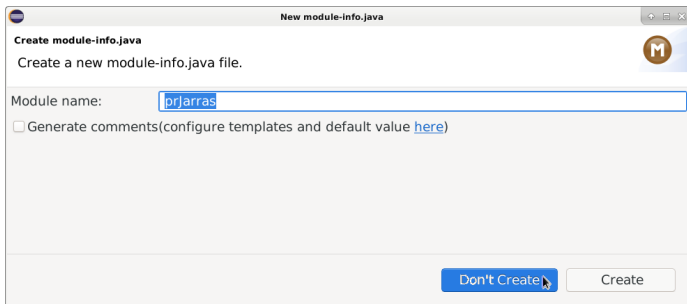
# Creación de un Nuevo Proyecto en Eclipse (III)

- Especificamos el nombre del proyecto, según las especificaciones del problema, marcamos (si es necesario) **Use an execution environment JRE: JavaSE-<versión>**, y pulsamos **Finish** para crear el proyecto especificado.



# Creación de un Nuevo Proyecto en Eclipse (IV)

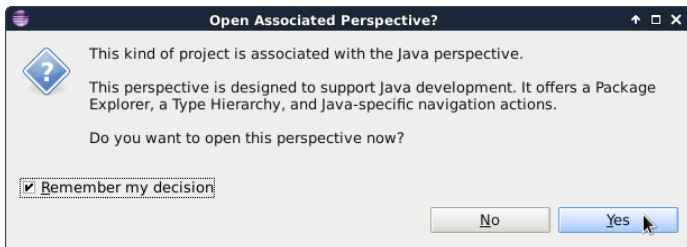
- En caso de que nos pregunte sobre la creación de un módulo (module-info.java), indicaremos que no se cree el módulo, pulsando sobre el botón **Don't Create**.





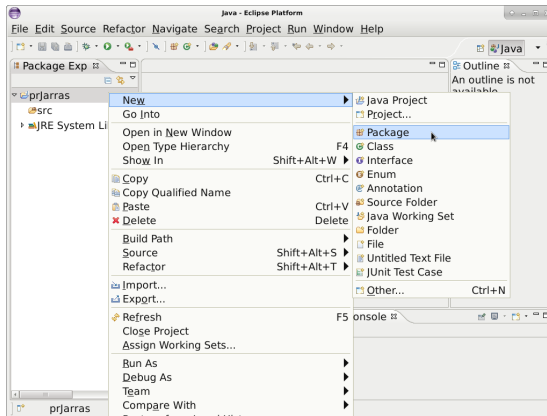
# Creación de un Nuevo Proyecto en Eclipse (V)

- En caso de que nos pregunte sobre la perspectiva, seleccionaremos la perspectiva de Java y seleccionaremos que *recuerde la decisión*.



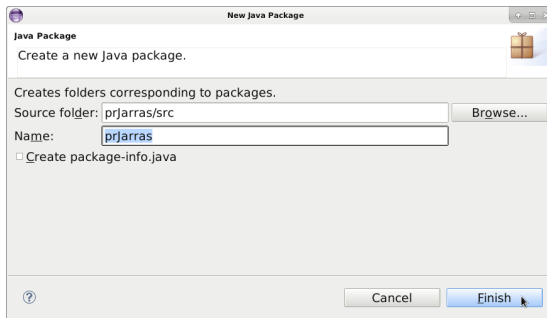
# Creación de un Nuevo Paquete en Eclipse (I)

- Añadiremos nuevos paquetes al proyecto seleccionando el proyecto, pulsando con el *botón derecho*, y seleccionando New -> Package.



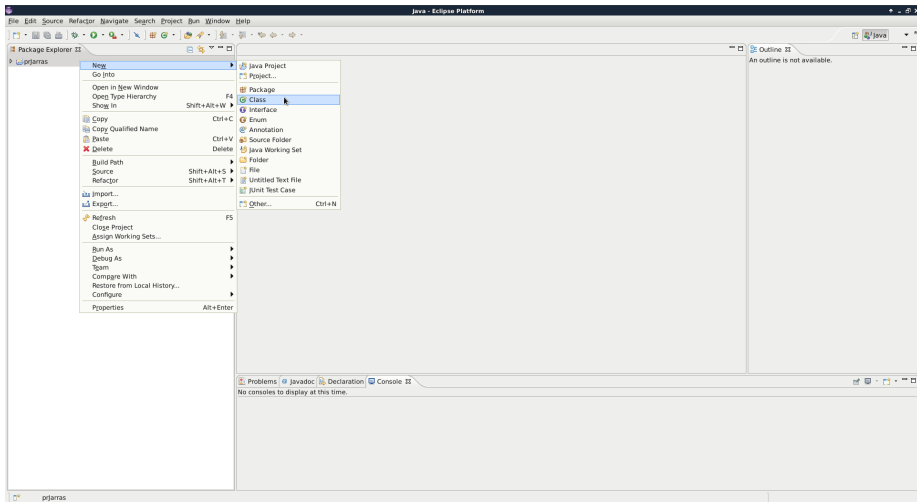
# Creación de un Nuevo Paquete en Eclipse (II)

- Al crear un nuevo paquete dentro de un proyecto, hay que especificar el **nombre del paquete**, y pulsar **Finish**.



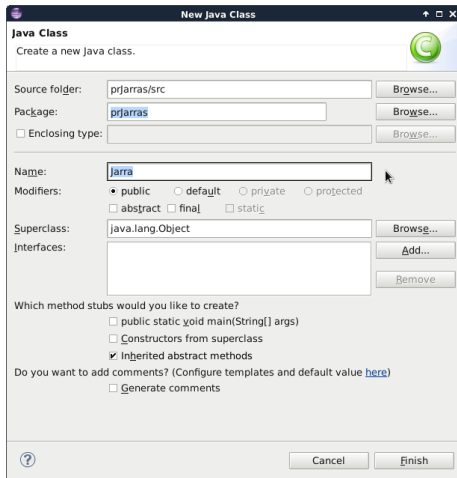
# Creación de una Nueva Clase en Eclipse (I)

- Añadiremos nuevas clases al proyecto seleccionando el proyecto, pulsando con el *botón derecho*, y seleccionando **New -> Class**.



# Creación de una Nueva Clase en Eclipse (II)

- Al crear una nueva clase, hay que especificar el **nombre del paquete** al que pertenece la nueva clase (se debe dejar en blanco para el paquete anónimo) y el **nombre de la clase**, y pulsar **Finish**.



The screenshot shows the 'New Java Class' dialog box in Eclipse. The title bar says 'New Java Class'. Inside, the 'Java Class' section has the text 'Create a new Java class.' and a green 'C' icon. Below this, there are three text fields for 'Source folder:', 'Package:', and 'Enclosing type:', each with a 'Browse...' button. The 'Package:' field contains 'prjarras'. The 'Name:' field contains 'jarra'. The 'Modifiers:' section has radio buttons for 'public' (selected), 'default', 'private', and 'protected', and checkboxes for 'abstract', 'final', and 'static'. The 'Superclass:' field contains 'java.lang.Object' with a 'Browse...' button. The 'Interfaces:' field is empty with 'Add...' and 'Remove' buttons. The 'Which method stubs would you like to create?' section has checkboxes for 'public static void main(String[] args)', 'Constructors from superclass', and 'Inherited abstract methods' (checked). The 'Do you want to add comments?' section has a checkbox for 'Generate comments'. At the bottom are 'Cancel' and 'Finish' buttons.

Java Class  
Create a new Java class.

Source folder: prjarras/src Browse...

Package: prjarras Browse...

☐ Enclosing type: Browse...

Name: jarra

Modifiers: ☒ public ☐ default ☐ private ☐ protected  
☐ abstract ☐ final ☐ static

Superclass: java.lang.Object Browse...

Interfaces: Add... Remove

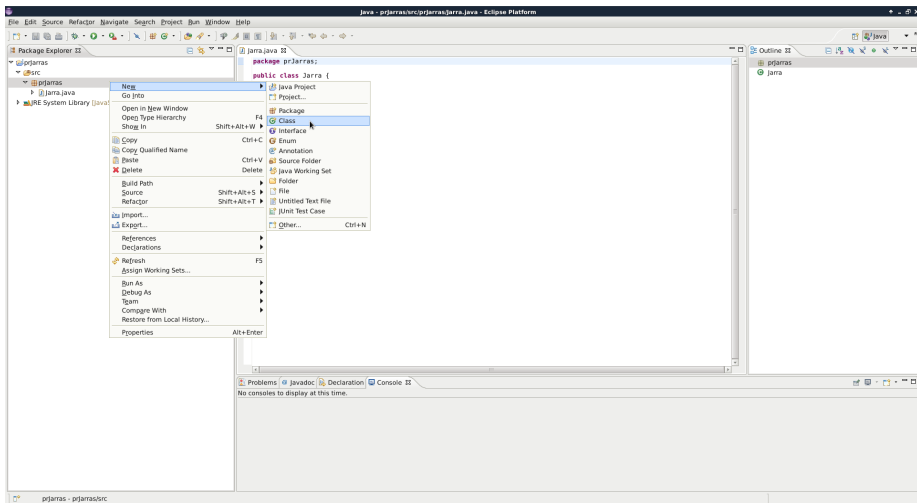
Which method stubs would you like to create?  
☐ public static void main(String[] args)  
☐ Constructors from superclass  
☒ Inherited abstract methods

Do you want to add comments? (Configure templates and default value [here](#))  
☐ Generate comments

Cancel Finish

# Creación de una Nueva Clase en Eclipse (III)

- Añadiremos clases directamente a un paquete del proyecto seleccionando la carpeta del paquete dentro del proyecto, pulsando con el *botón derecho*, y seleccionando New -> Class.



# Creación de una Nueva Clase en Eclipse (IV)

- Al crear una nueva clase dentro de un paquete específico, el *nombre del paquete* al que pertenece se especifica automáticamente, por lo que solo debemos especificar el **nombre de la clase**, y pulsar Finish.



The screenshot shows the 'New Java Class' dialog box in Eclipse. The title bar says 'New Java Class'. Inside, the 'Java Class' section has the text 'Create a new Java class.' and a green 'C' icon. Below this, there are three fields for 'Source folder:', 'Package:', and 'Enclosing type:', each with a 'Browse...' button. The 'Package:' field contains 'prjarras'. The 'Name:' field is empty. The 'Modifiers:' section has radio buttons for 'public' (selected), 'default', 'private', and 'protected', and checkboxes for 'abstract', 'final', and 'static'. The 'Superclass:' field contains 'java.lang.Object' with a 'Browse...' button. The 'Interfaces:' section has an 'Add...' button and a 'Remove' button. Below these, there is a section 'Which method stubs would you like to create?' with checkboxes for 'public static void main(String[] args)', 'Constructors from superclass', and 'Inherited abstract methods' (checked). At the bottom, there is a section 'Do you want to add comments? (Configure templates and default value [here](#))' with a checkbox for 'Generate comments'. The dialog has 'Cancel' and 'Finish' buttons at the bottom right.

Java Class

Create a new Java class.

Source folder: prjarras/src Browse...

Package: prjarras Browse...

☐ Enclosing type: Browse...

Name:

Modifiers: ☒ public ☐ default ☐ private ☐ protected  
☐ abstract ☐ final ☐ static

Superclass: java.lang.Object Browse...

Interfaces: Add... Remove

Which method stubs would you like to create?

☐ public static void main(String[] args)

☐ Constructors from superclass

☒ Inherited abstract methods

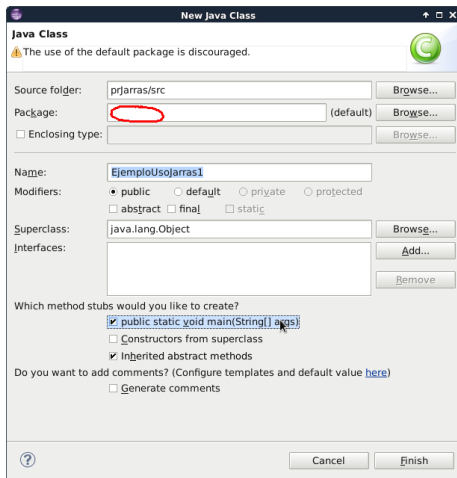
Do you want to add comments? (Configure templates and default value [here](#))

☐ Generate comments

Cancel Finish

# Creación de una Nueva Clase en Eclipse (V)

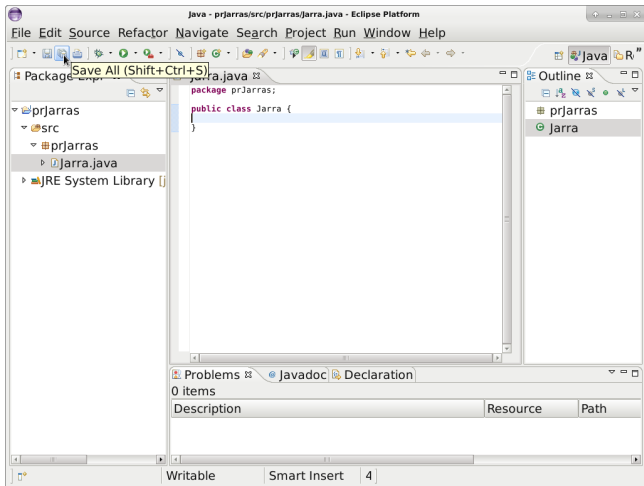
- Al crear una nueva clase distinguida para **main**, usualmente la crearemos en el **paquete anónimo**, por lo que el nombre del paquete se debe dejar en **blanco**, especificar el **nombre de la clase**, marcar que genere el método **main()** y pulsar **Finish**.





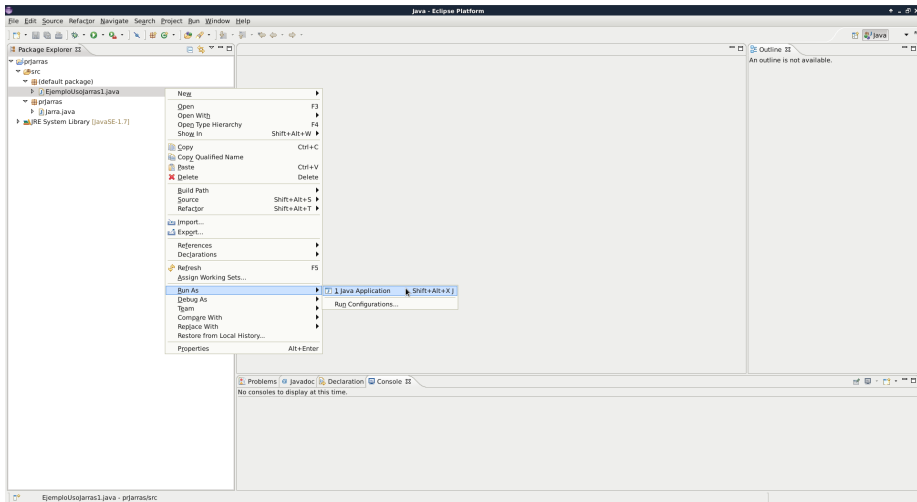
# Guardando los Cambios de los Ficheros

- Se puede pulsar sobre el icono de Save (Ctrl+S) o Save All (Ctrl+Shift+S) para guardar los cambios a los ficheros (el asterisco \* en la pestaña indica que la versión actual no está *guardada* en el fichero).
- Se debe tener en cuenta que tanto el sistema como el entorno de desarrollo podrían **fallar**, y se podría **perder** todo el trabajo realizado, por lo que es importante **guardar frecuentemente** el código fuente.



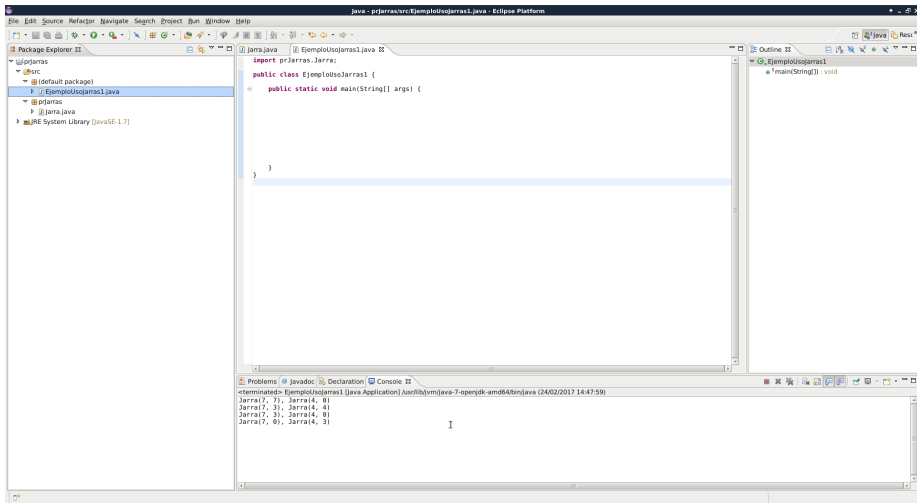
# Ejecución de Aplicaciones Java en Eclipse (I)

- Para ejecutar una determinada clase distinguida (**main**), seleccionaremos la clase, pulsando con el *botón derecho*, y seleccionaremos Run As -> Java Application.



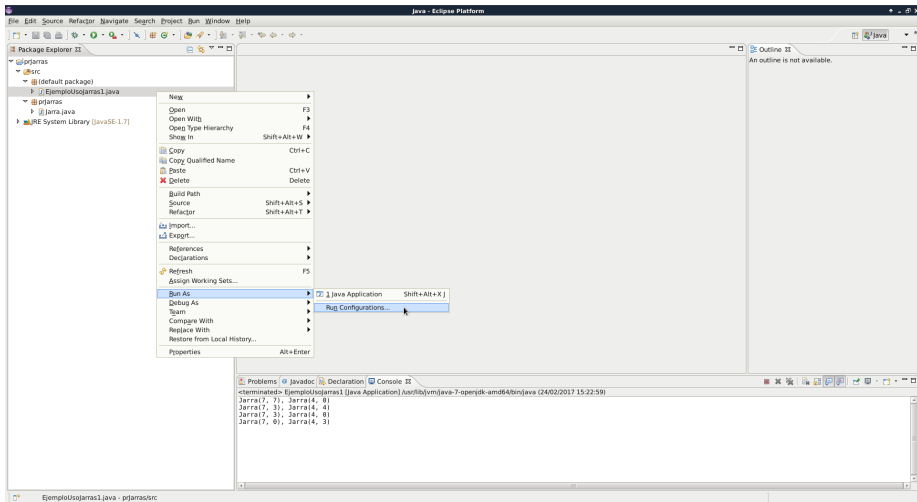
# Ejecución de Aplicaciones Java en Eclipse (II)

- La salida de la ejecución de nuestra aplicación se muestra en la **consola** (ventana inferior).



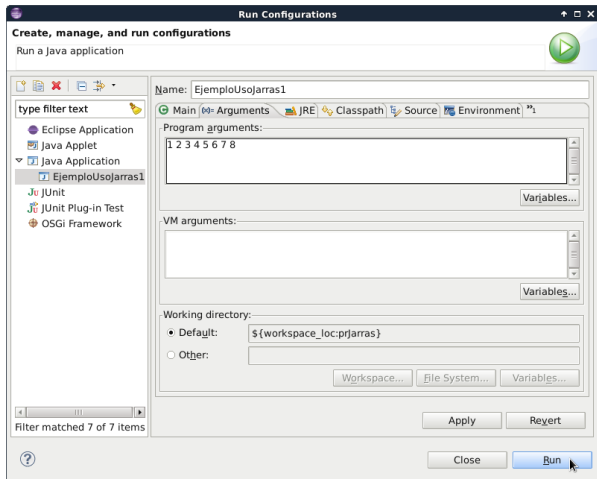
# Ejecución de Aplicaciones Java en Eclipse (III)

- En caso de querer pasar argumentos a la ejecución de nuestra aplicación, seleccionaremos la clase, pulsando con el *botón derecho*, y seleccionaremos Run As -> Run Configurations.



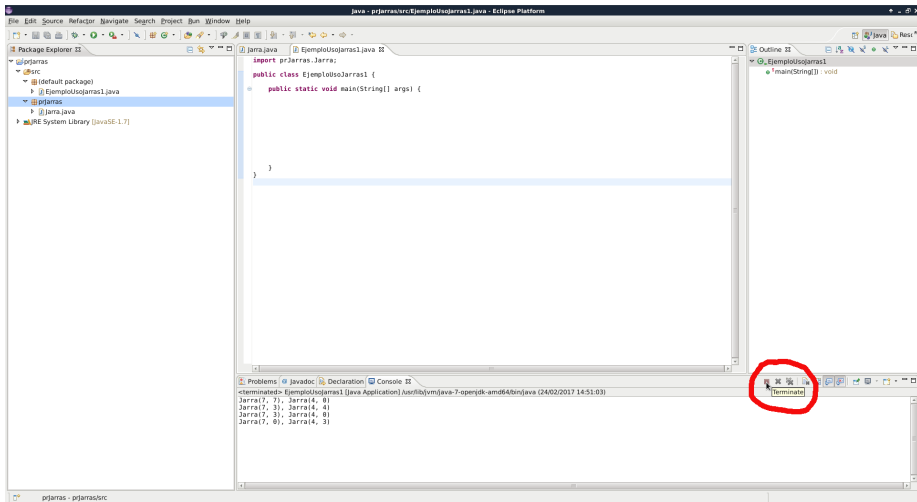
# Ejecución de Aplicaciones Java en Eclipse (IV)

- En caso de querer pasar argumentos a la ejecución de nuestra aplicación, seleccionaremos la pestaña Arguments, especificaremos los argumentos, y pulsaremos Run.



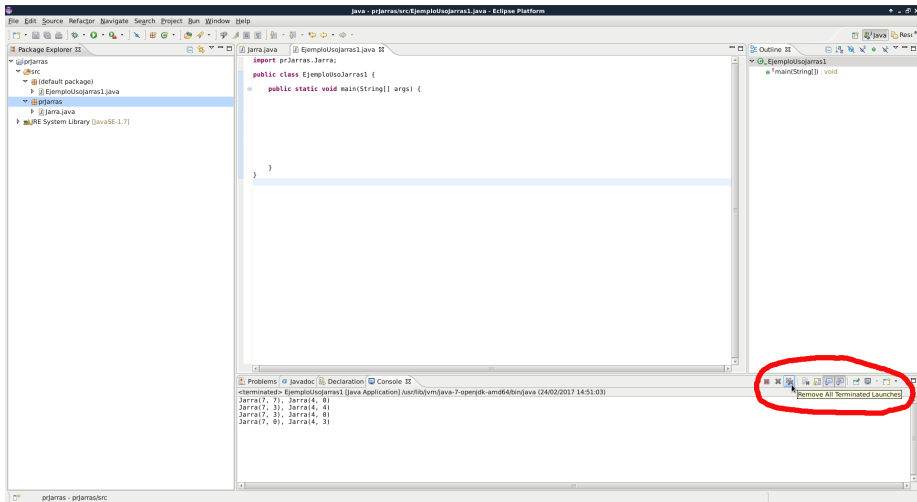
# Ejecución de Aplicaciones Java en Eclipse (V)

- En caso de querer **abortar** la ejecución de nuestra aplicación, pulsaremos sobre el icono Terminate de la ventana de **consola**.



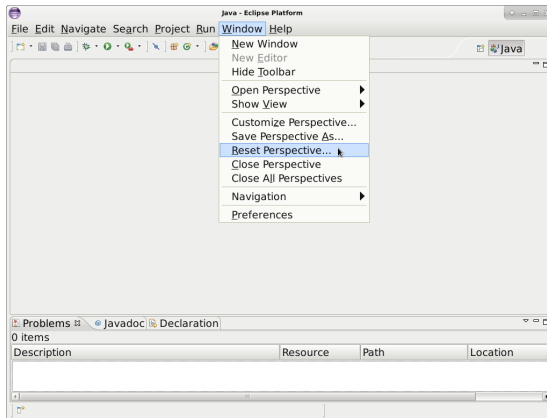
# Ejecución de Aplicaciones Java en Eclipse (VI)

- En caso de tener múltiples instancias de las ventanas de consola, podemos eliminarlas pulsando sobre el icono **Remove All Terminated Launches** de la ventana de **consola**.



# Restaurar la Perspectiva por Defecto

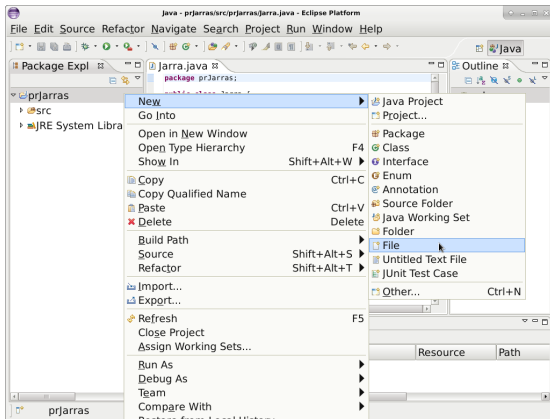
- A veces cambiamos o eliminamos algunas ventanas de la perspectiva que posteriormente deseamos restaurar. La siguiente opción permite restaurar el estado de la **perspectiva por defecto**.





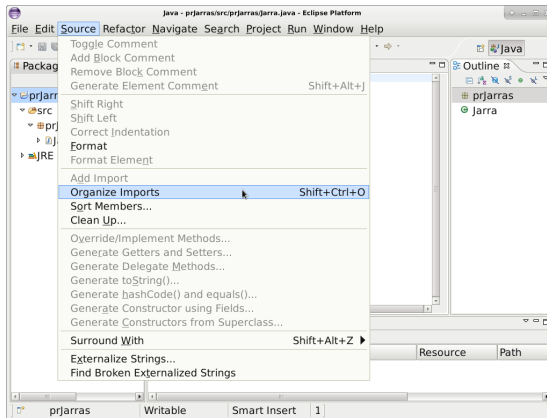
# Ficheros de Datos e Imágenes

- Usualmente, solemos situar los ficheros de datos e imágenes en la carpeta raíz del proyecto. En caso de que el fichero ya exista, simplemente lo copiaremos a dicha carpeta raíz del proyecto. En caso de que queramos crear con el editor el fichero con los datos adecuados, podemos utilizar la siguiente opción:



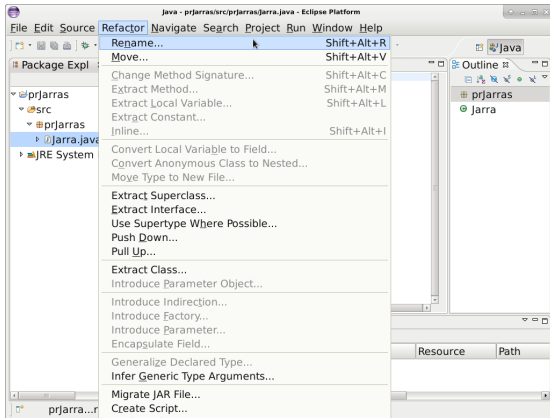
# Organizar Automáticamente los Paquetes Importados

- La siguiente opción permite organizar automáticamente los paquetes importados. La siguiente combinación de teclas (**Ctrl + Mayús + O**) permite realizar la misma función.



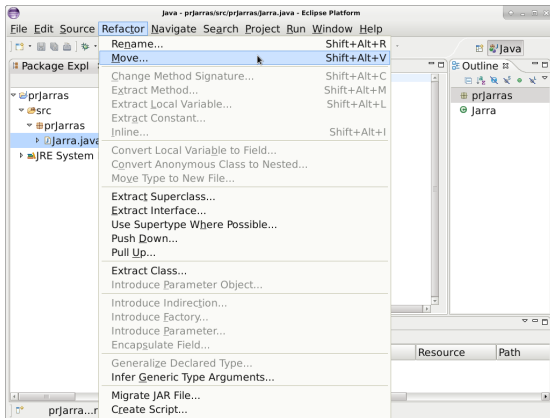
# Modificar el Nombre de Proyectos, Paquetes y Clases

- Para modificar el nombre del proyecto, o del paquete o de una determinada clase, se debe marcar la entidad (proyecto, paquete o clase) que se quiera renombrar, y seleccionar la siguiente opción.



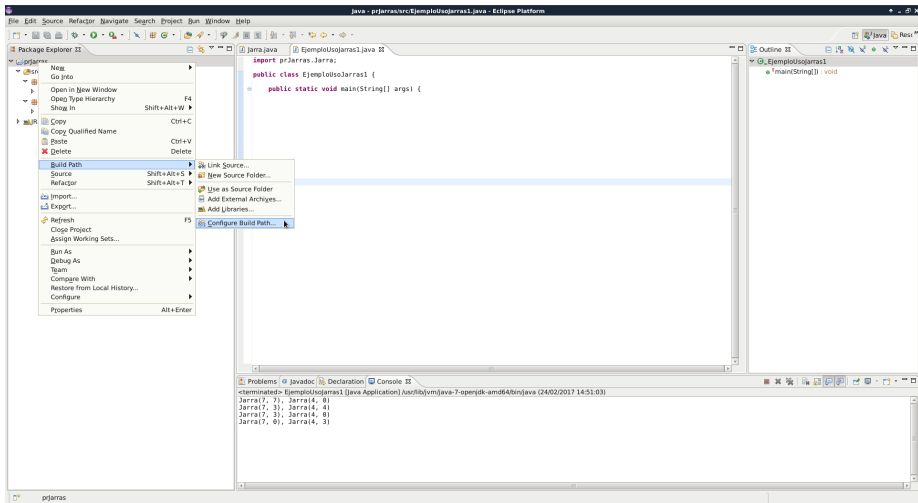
# Mover Clases y Ficheros a otros Paquetes y Carpetas

- Para mover una determinada clase a otro paquete, o para mover un determinado fichero de recursos a otra carpeta, se debe marcar la entidad (clase o fichero) que se quiera mover, y seleccionar la siguiente opción.



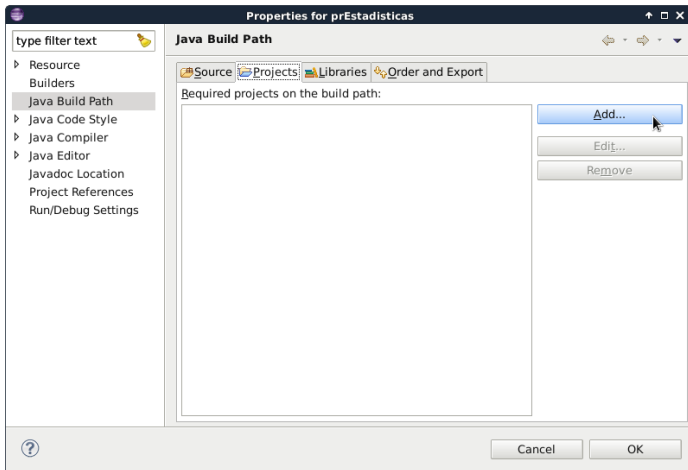
# Utilización de Clases de Otros Proyectos en Eclipse (I)

- En caso de querer utilizar las **clases de otro proyecto** en el proyecto actual, debemos seleccionar el proyecto actual, pulsando con el *botón derecho*, seleccionamos Build Path y seleccionamos Configure Build Path.



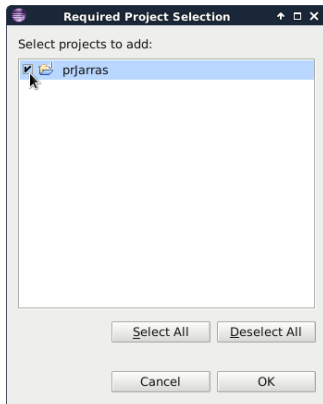
# Utilización de Clases de Otros Proyectos en Eclipse (II)

- Seleccionamos la pestaña Projects y pulsamos sobre el botón Add.



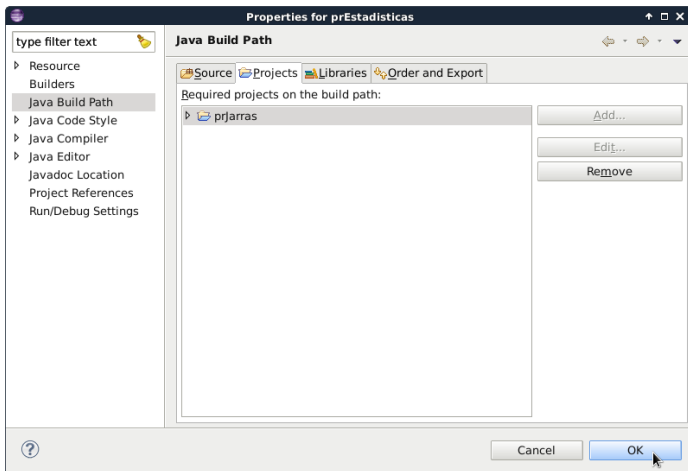
# Utilización de Clases de Otros Proyectos en Eclipse (III)

- Seleccionamos y **marcamos** aquellos proyectos donde residen las clases que queremos utilizar en nuestro nuevo proyecto.



# Utilización de Clases de Otros Proyectos en Eclipse (IV)

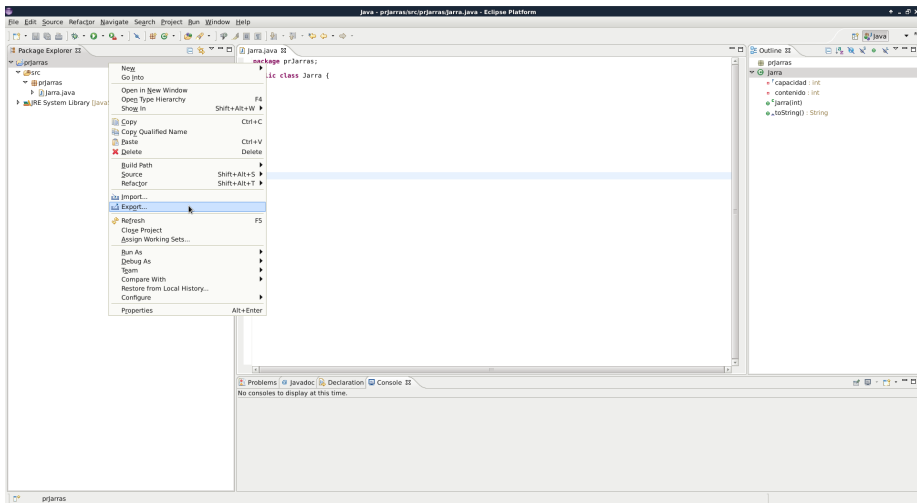
- Finalmente, pulsamos sobre OK cuando hayamos añadido todos los proyectos que queramos utilizar.





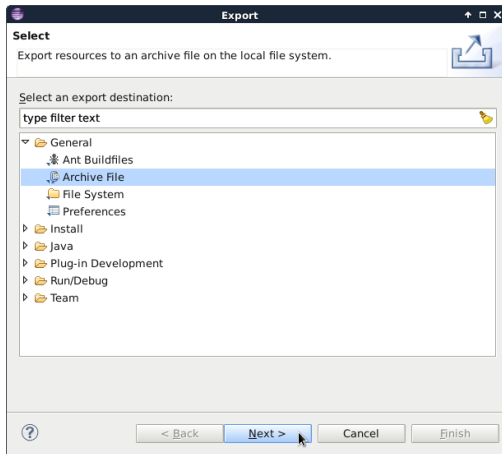
# Exportar Proyectos en Eclipse (I)

- En caso de querer **exportar** un determinado proyecto, debemos seleccionar el proyecto actual, pulsando con el *botón derecho*, seleccionamos **Export**.



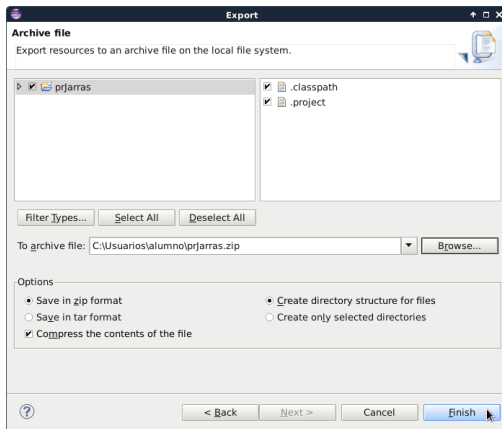
# Exportar Proyectos en Eclipse (II)

- En caso de querer **exportar** un determinado proyecto, seleccionamos Archive File para indicar que queremos exportar a un *archivo ZIP*, y pulsamos Next.



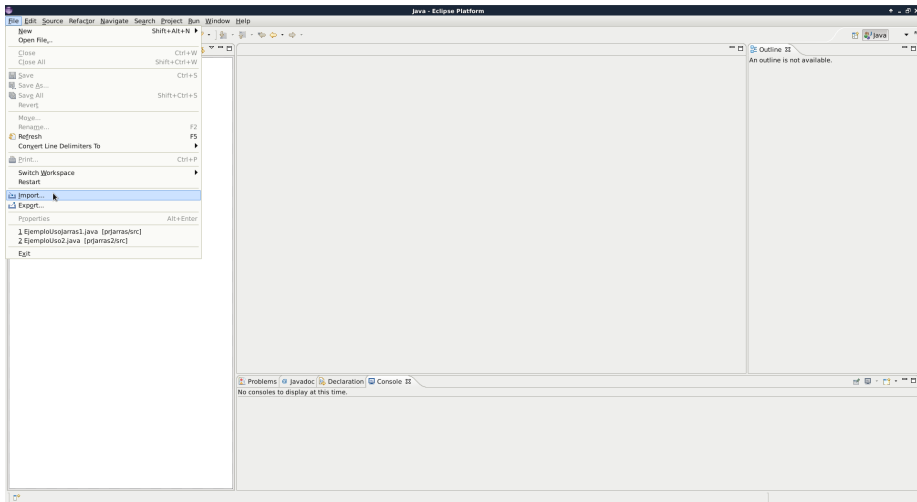
# Exportar Proyectos en Eclipse (III)

- En caso de querer **exportar** un determinado proyecto, seleccionamos (Browse) la *ruta* y el nombre del *archivo ZIP* donde será exportado el proyecto, marcamos la opción Save in zip format, y pulsamos Finish.



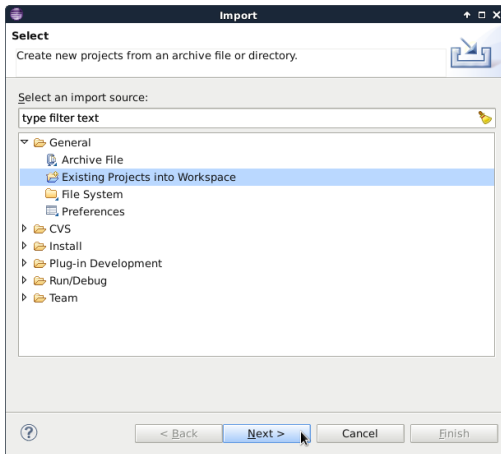
# Importar Proyectos en Eclipse (I)

- En caso de querer **importar** un proyecto *exportado* según el método especificado anteriormente, debemos seleccionar el menú **File**, y seleccionamos **Import**.



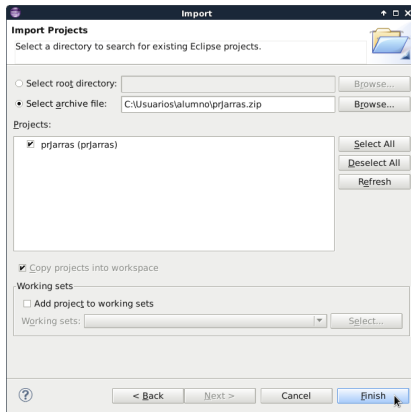
# Importar Proyectos en Eclipse (II)

- En caso de querer **importar** un proyecto, seleccionamos Existing Project into Workspace para indicar que queremos importar desde un *proyecto Eclipse*, y pulsamos Next.



# Importar Proyectos en Eclipse (III)

- En caso de querer **importar** un proyecto, marcamos **Select archive file** para indicar que queremos importar desde un proyecto Eclipse exportado a un *archivo ZIP*, seleccionamos (Browse) el *archivo ZIP* desde donde importar el proyecto, y pulsamos **Finish**



- Combinaciones de teclas útiles:
  - **Ctrl + Mayús + O**: organizar las importaciones de paquetes.
  - **Ctrl + Mayús + F**: formatea el código.
  - **Ctrl + Espacio**: asistente de contexto.
  - **Ctrl + I**: corregir la indentación.
  - **Ctrl + 1**: corrección rápida.
  - **Ctrl + /**: comentar una zona.
- Configurar el auto-completado (**esta opción puede afectar al rendimiento**):

Windows -> Preferences -> Java -> Editor -> Content Assist

- Pulsar el botón "Restore Defaults"
- Rellenar el campo "Auto activation triggers for Java" con:
  - ABCDEFGHIJKLMNOPQRSTUVWXYZ.abcdefghijklmnopqrstuvwxyz
  - Nótese el punto después de la Z para que también lo tenga en cuenta.
- En la ventana Advanced, deben estar marcados (en ambas sub-ventanas):
  - Java Proposals (Task-Focused)
  - SWT Template Proposals
  - Template Proposals
- Pulsar el botón "Apply and Close"