

# Configuración del entorno

## Proyecto Escuela

### Configuración del entorno Eclipse

Versión 1.00 del jueves, 29 de marzo de 2018

Estado : Trabajo

## Destinatario(s)

## Historial

Versión	Fecha	Origen de la actualización	Redactado por	Validado por
1.0.0	28/03/2018		Silvia Carpena	
1.0.1	29/03/2018		David Díez	
1.0.2	03/04/2018		Ana Collado	

# Índice

1.	Instalación Eclipse	4
1.1.	<b>Descargar Eclipse</b>	<b>4</b>
1.2.	<b>Instalar Eclipse en el equipo</b>	<b>4</b>
1.3.	<b>Instalar el JDK (Java Development Kit)</b>	<b>5</b>
2.	Incluir Tomcat en Eclipse	5
3.	Templates	12
4.	Plugin: SonarLint	13
4.1.	<b>Descargar e instalar el plugin SonarLint en Eclipse</b>	<b>14</b>
4.2.	<b>Uso de SonarLint</b>	<b>16</b>
5.	Plugin : Hibernate Tools	19
5.1.	<b>Descargar e instalar el plugin Hibernate Tools en Eclipse</b>	<b>19</b>
6.	Plugin : ERMaster	23
6.1.	<b>Descargar e instalar el plugin ERMaster en Eclipse</b>	<b>23</b>
7.	Plugin : Checkstyle	24
7.1.	<b>Descargar e instalar el plugin Checkstyle en Eclipse</b>	<b>25</b>
7.2.	<b>Uso de SonarLint</b>	<b>26</b>
8.	GIT	28

# 1. Instalación Eclipse

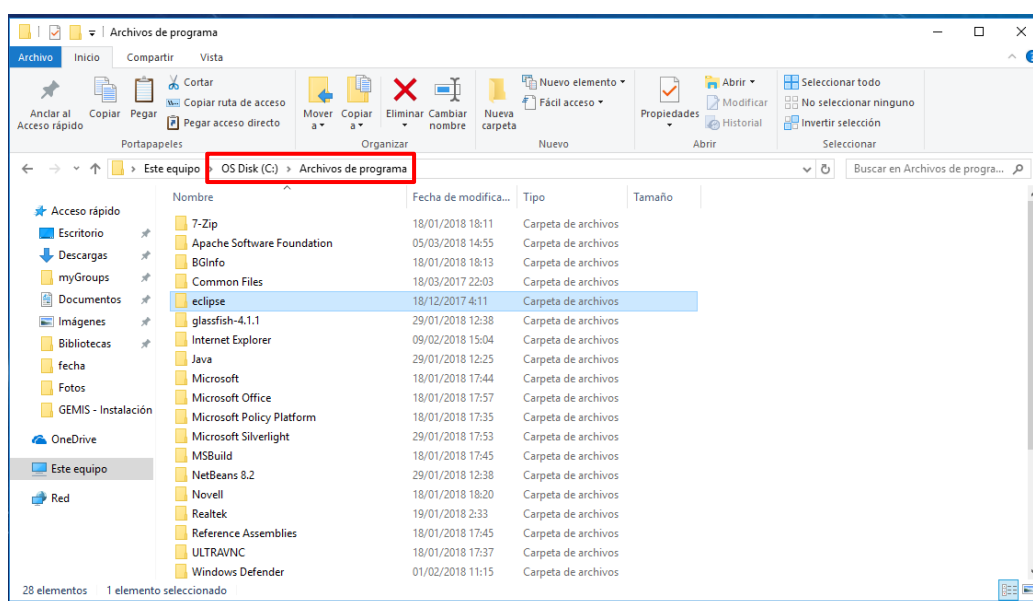
## 1.1. Descargar Eclipse

Para la elaboración de este documento se ha utilizado la versión de Eclipse Oxygen 2, la cual se puede descargar a través del siguiente enlace:

[http://www.eclipse.org/downloads/download.php?file=/technology/epp/downloads/release/oxygen/2/eclipse-je-e-oxygen-2-win32-x86\\_64.zip](http://www.eclipse.org/downloads/download.php?file=/technology/epp/downloads/release/oxygen/2/eclipse-je-e-oxygen-2-win32-x86_64.zip)

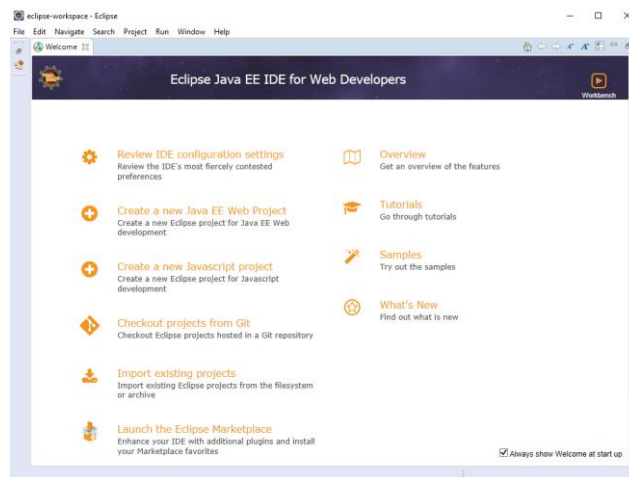
## 1.2. Instalar Eclipse en el equipo

Para instalar Eclipse se deberá descomprimir el archivo .zip descargado en el punto anterior en el directorio **C: > Archivos de programa**:



Para ejecutar Eclipse hay que hacer doble clic al archivo **eclipse.exe** que se encuentra dentro del directorio.





### 1.3. Instalar el JDK (Java Development Kit)

En caso de que no esté previamente instalado en el equipo, se deberá descargar e instalar el JDK. Se descargará directamente de la web de Oracle:

<http://www.oracle.com/technetwork/java/javase/downloads/index.html>

En caso de que el JDK no estuviese instalado se deberá añadir el archivo ejecutable (jdk-10\_windows-x64\_bin.exe, en el caso de que el SO sea Windows) al PATH del sistema. Para ello, se deberá acceder a :

Panel de control > Sistema > Configuración avanzada del sistema > Variables de entorno

Se deberá añadir a la variable del sistema Path la ubicación del jdk de Java.

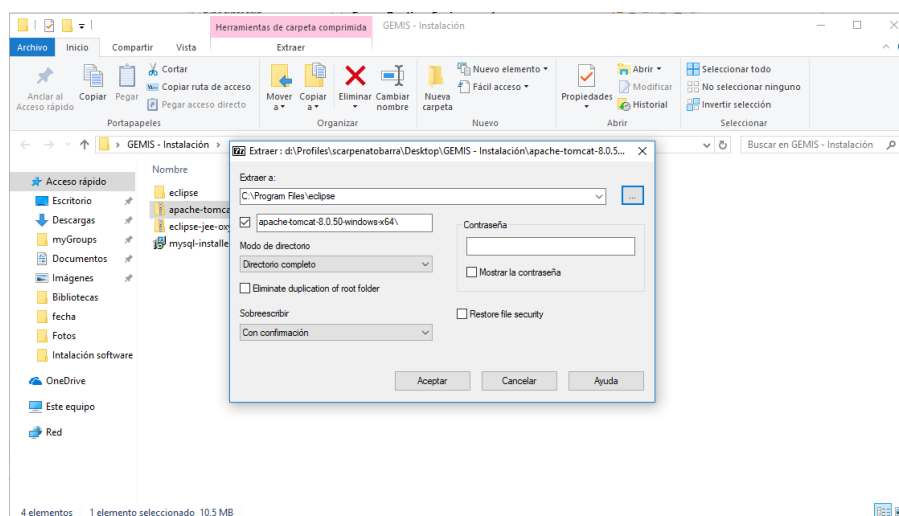
## 2. Incluir Tomcat en Eclipse

Es necesario descargar Tomcat 8.0.50, el cual se puede descargar en:

<https://tomcat.apache.org/download-80.cgi#8.0.50>

A continuación, hay que extraer Tomcat **en el directorio donde está instalado Eclipse** (en este caso C:/Archivos de programa/eclipse). Para ello se puede usar el asistente de 7-ZIP como se muestra en la captura o copiarla manualmente.

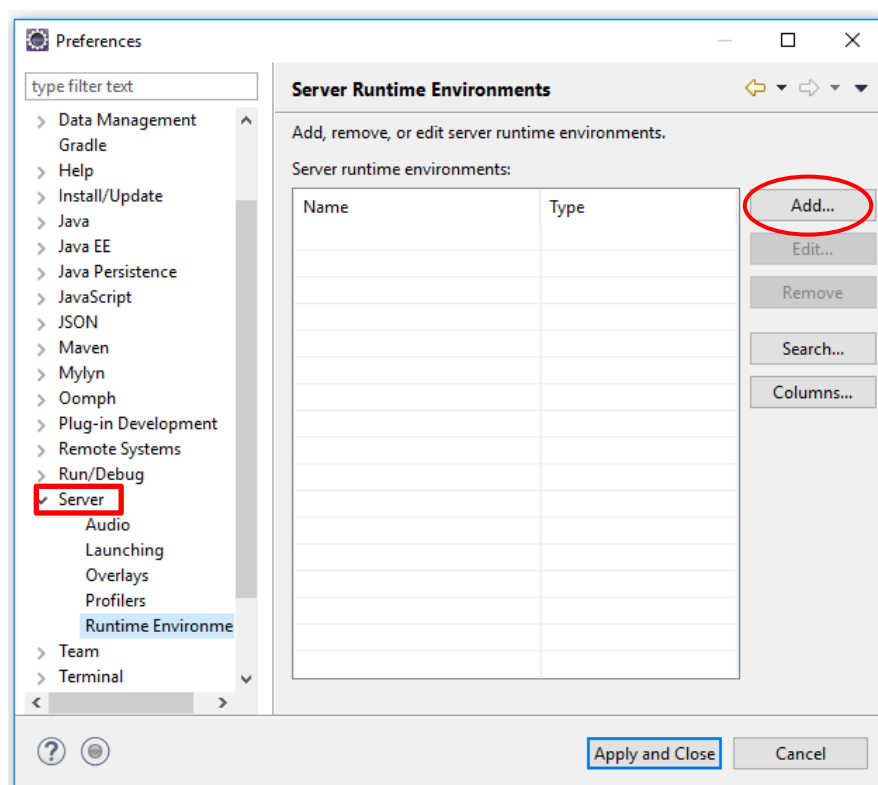


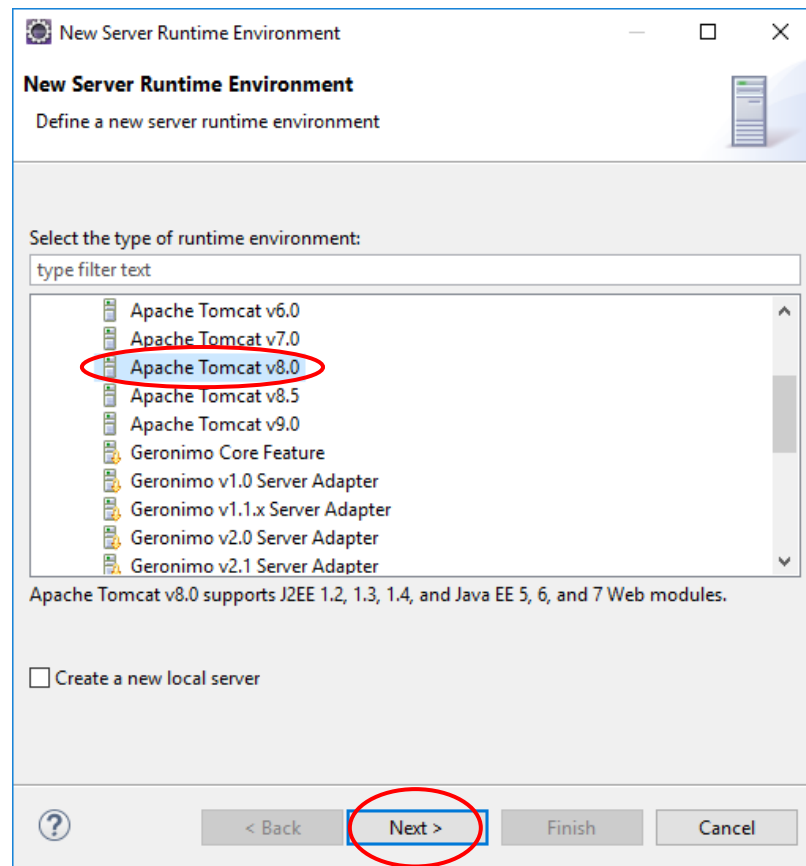


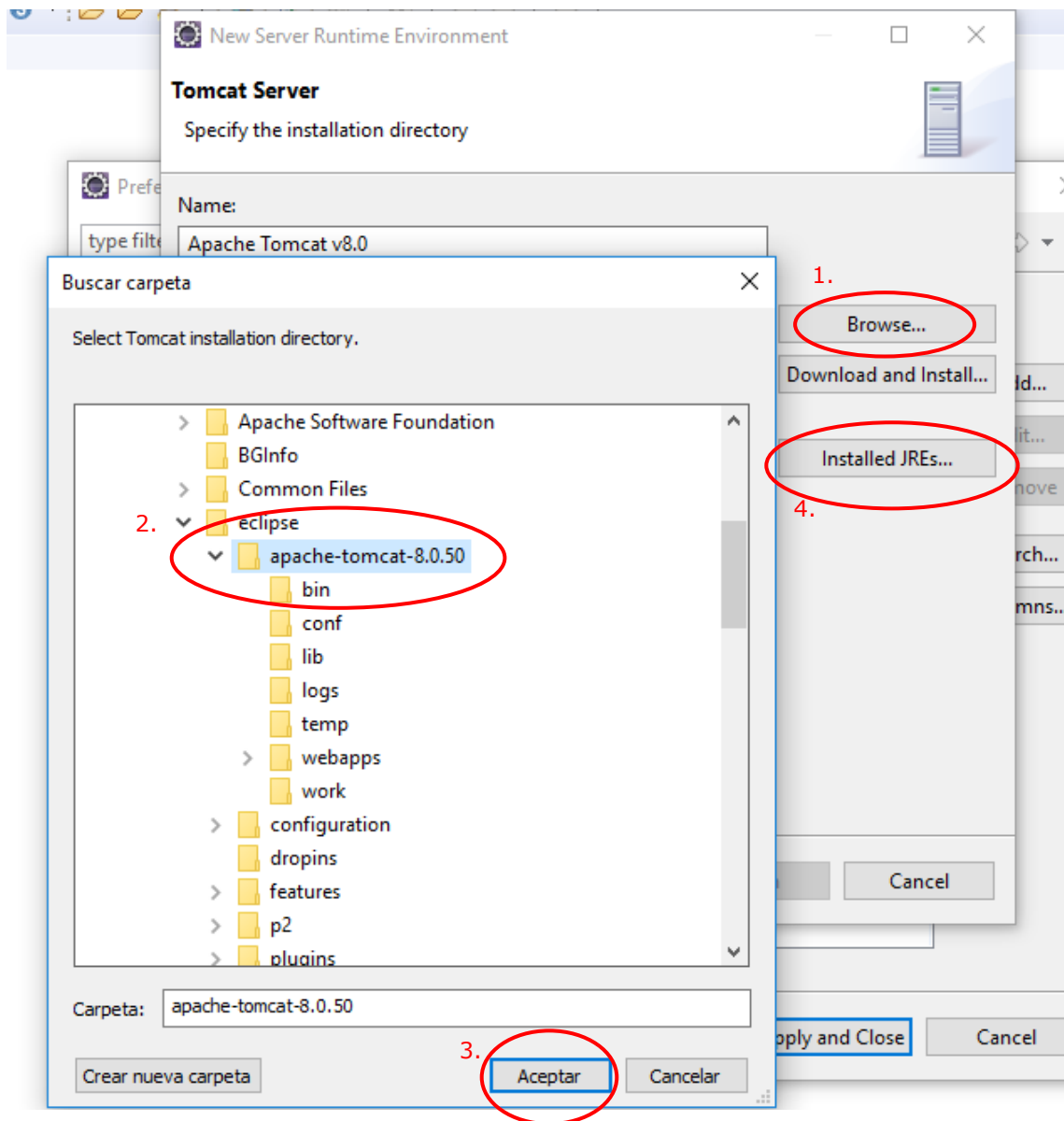
Volviendo a **Eclipse**:

A continuación, se seguirán los siguientes pasos:

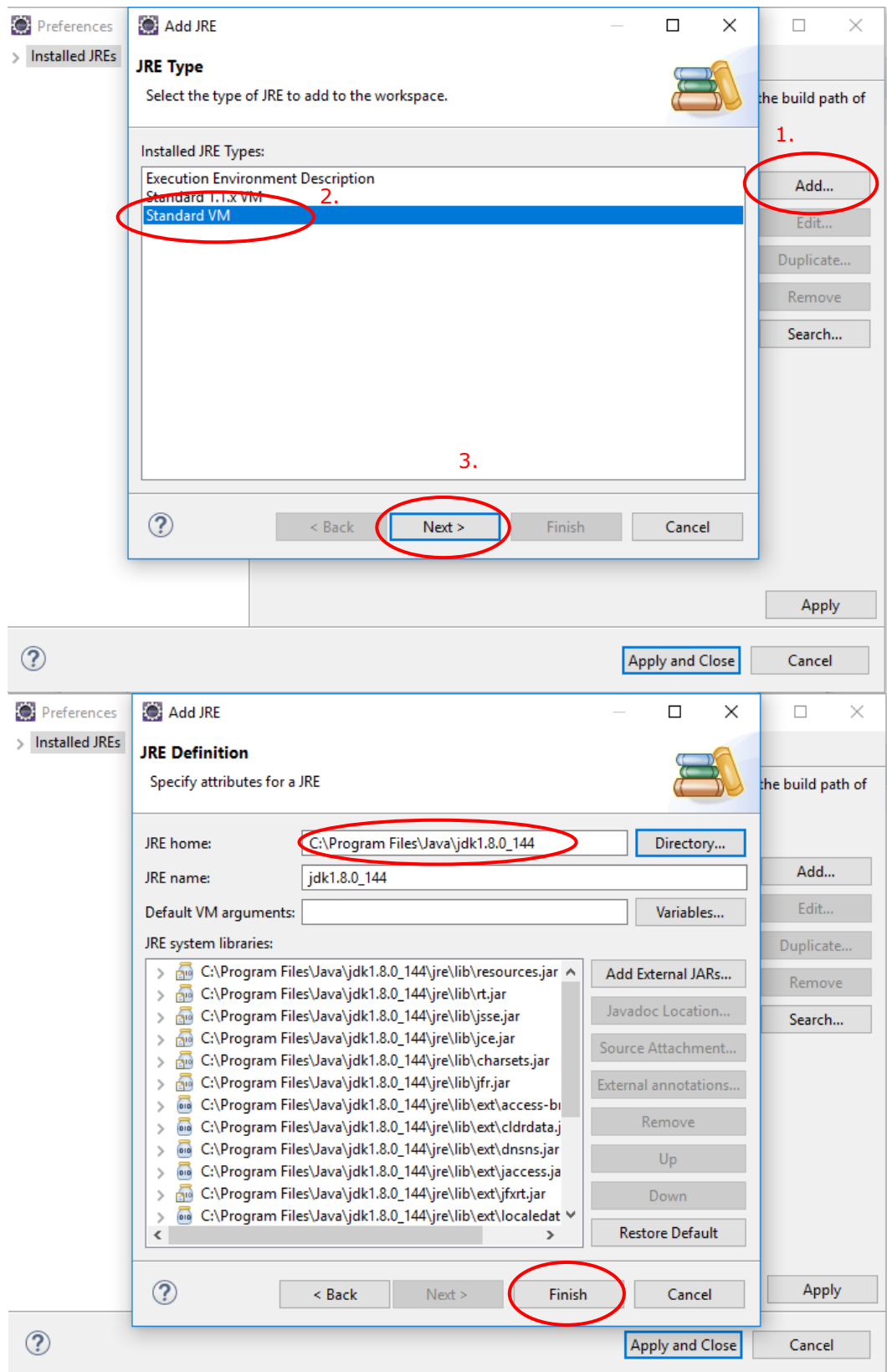
- Ir a Window > Show view > Servers (para que Eclipse muestre la pestaña Servers)
- Ir a Window > Preferences

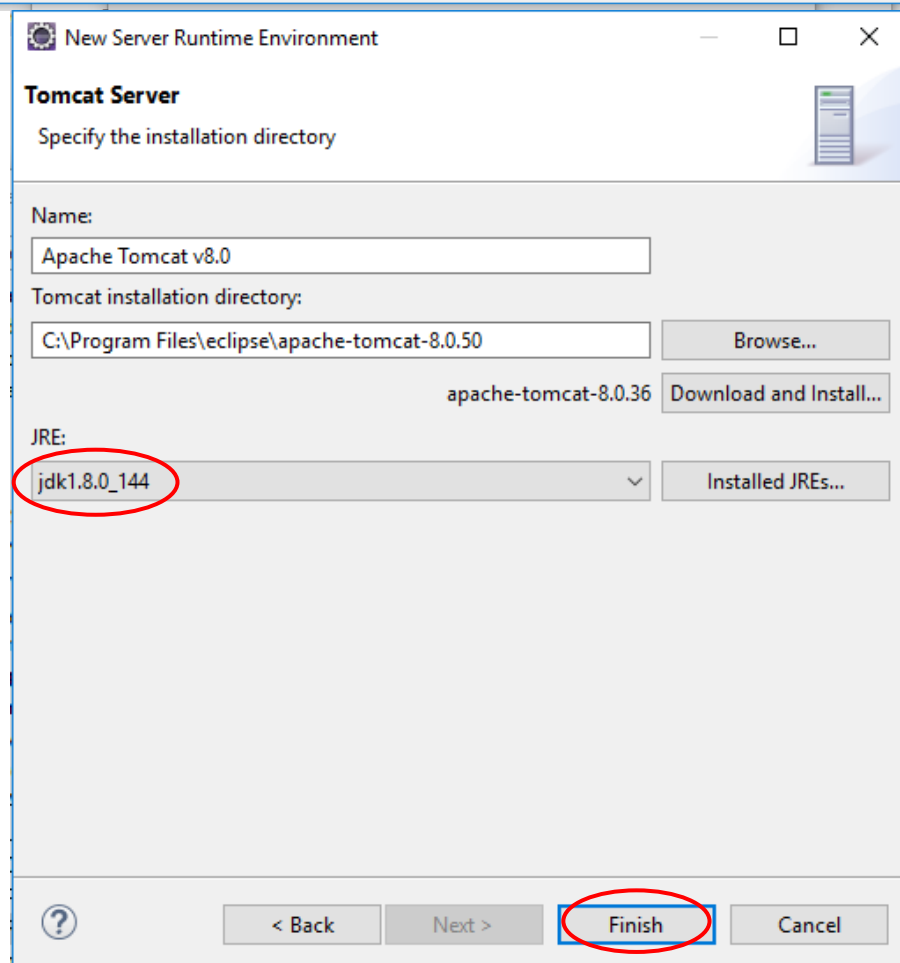
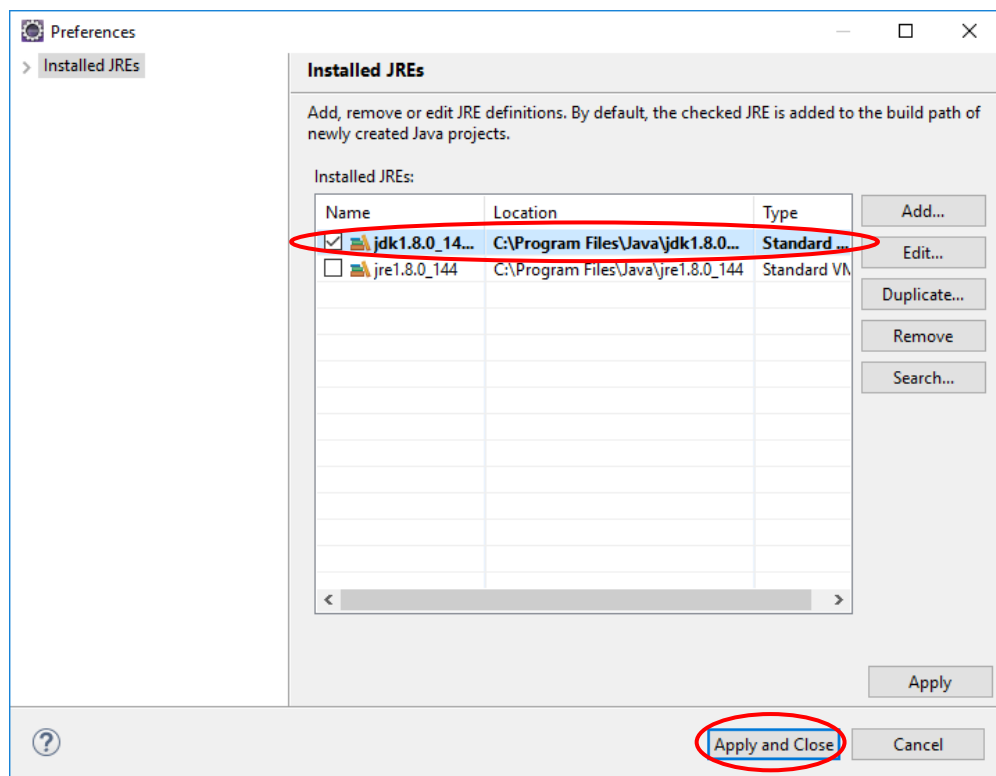


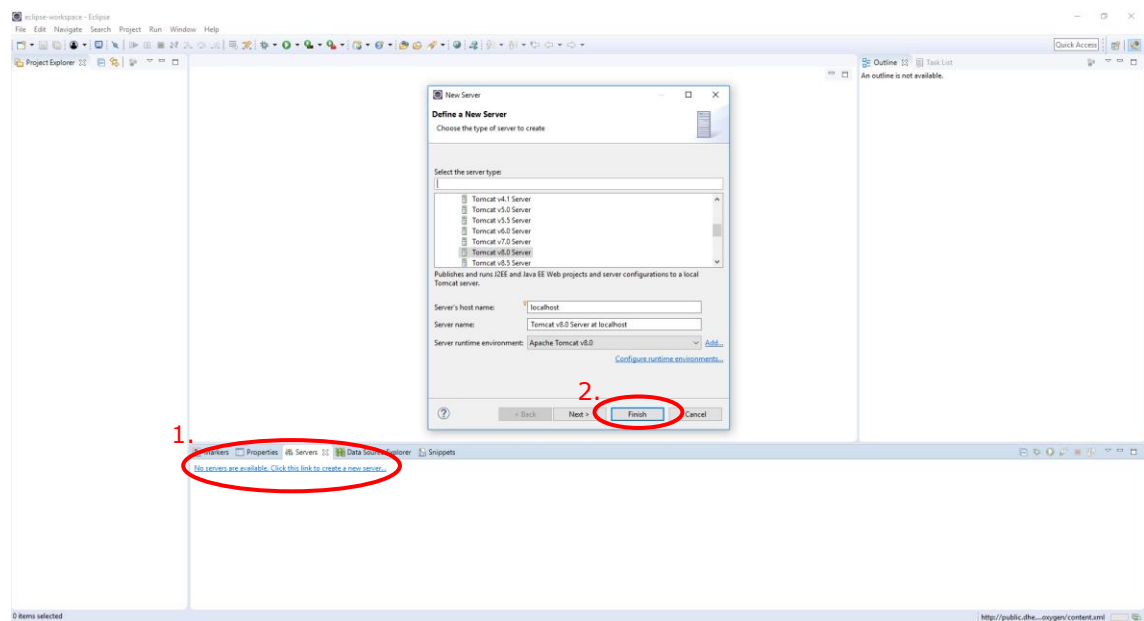






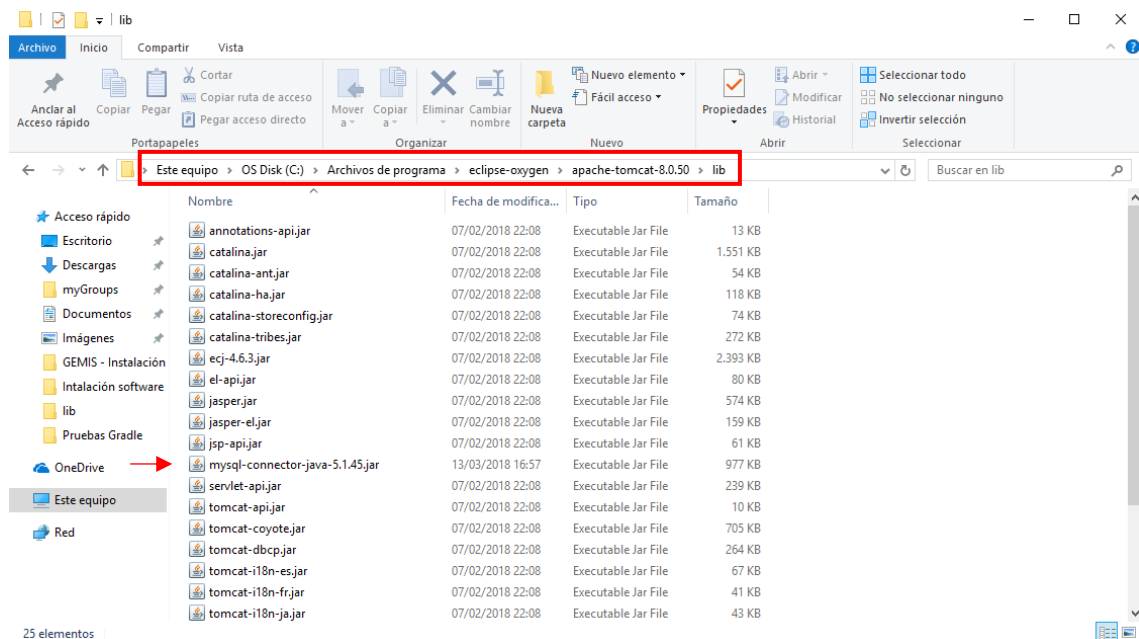






Para poder conectarnos a la base de datos necesitaremos descargar el conector de MySQL desde la siguiente URL: <https://dev.mysql.com/downloads/connector/j/>

El **.jar** que descarguemos lo debemos incluir en el siguiente directorio:

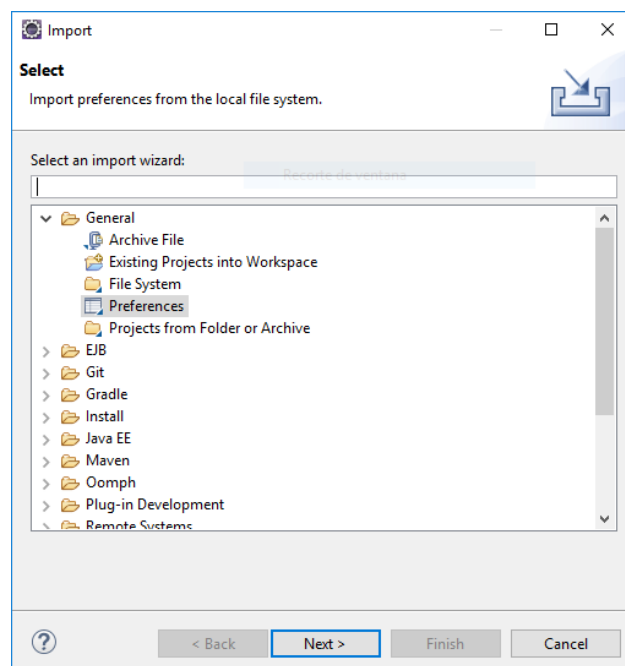


### 3. Templates

Para importar el archivo de preferencias se debe hacer desde Eclipse. Previamente se dispondrá de dos archivos de preferencias preparados, que son los siguientes:

poste-dev-1.0.1-preferences-codestyle-eclipse.epf  
poste-dev-1.0.7-preferences-eclipse.epf

Desde Eclipse, menú File -> Import y se importa el tipo de archivo, preferencias, dentro del apartado General.

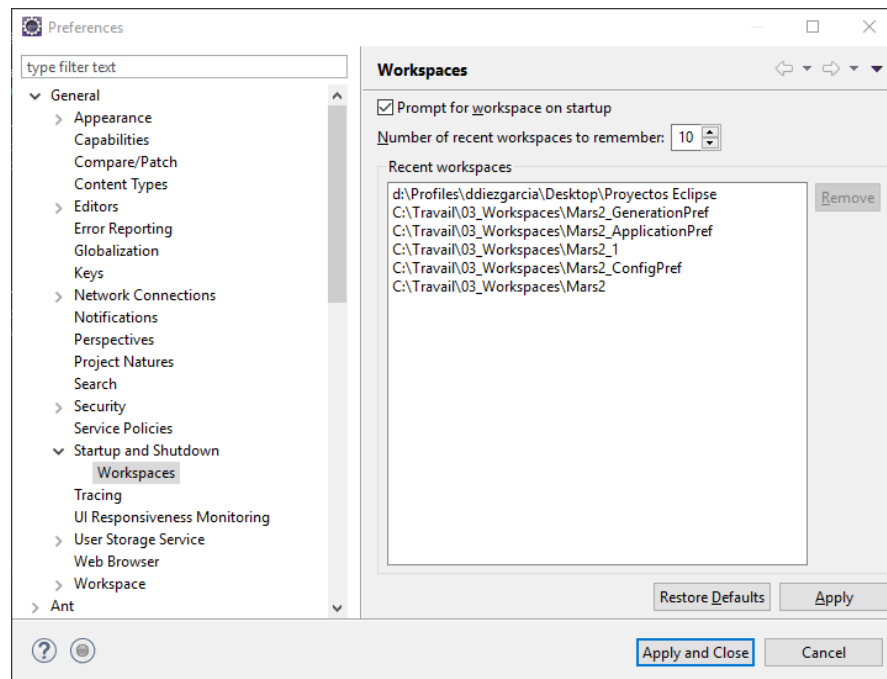


Esta configuración de Eclipse sirve como ayuda para que el código pase correctamente por SonarLint. Eclipse lanzará warnings, por ejemplo, cuando no se escriba el JAVADOC después de escribir cualquier método y/o clase.

El archivo de preferencias también importa los workspace definidos en el sistema, al reiniciar Eclipse si no lo cambiamos puede llevar a confusión no encontrar el proyecto en el que se desea trabajar. Para solventar este problema se va a Windows -> Preferencias y dentro del menú de configuración

General -> Startup and Shutdown -> Workspaces





En este paso se quitan los workspaces que no se vayan a utilizar, se han añadido desde el archivo de preferencias que se ha cargado ya que se ha obtenido de un proyecto anterior y los workspace quedan configurados en él. Se debe eliminar:

C:\Travail\03\_Workspaces\Mars2\_GenerationPref

C:\Travail\03\_Workspaces\Mars2\_ApplicationPref

C:\Travail\03\_Workspaces\Mars2\_1

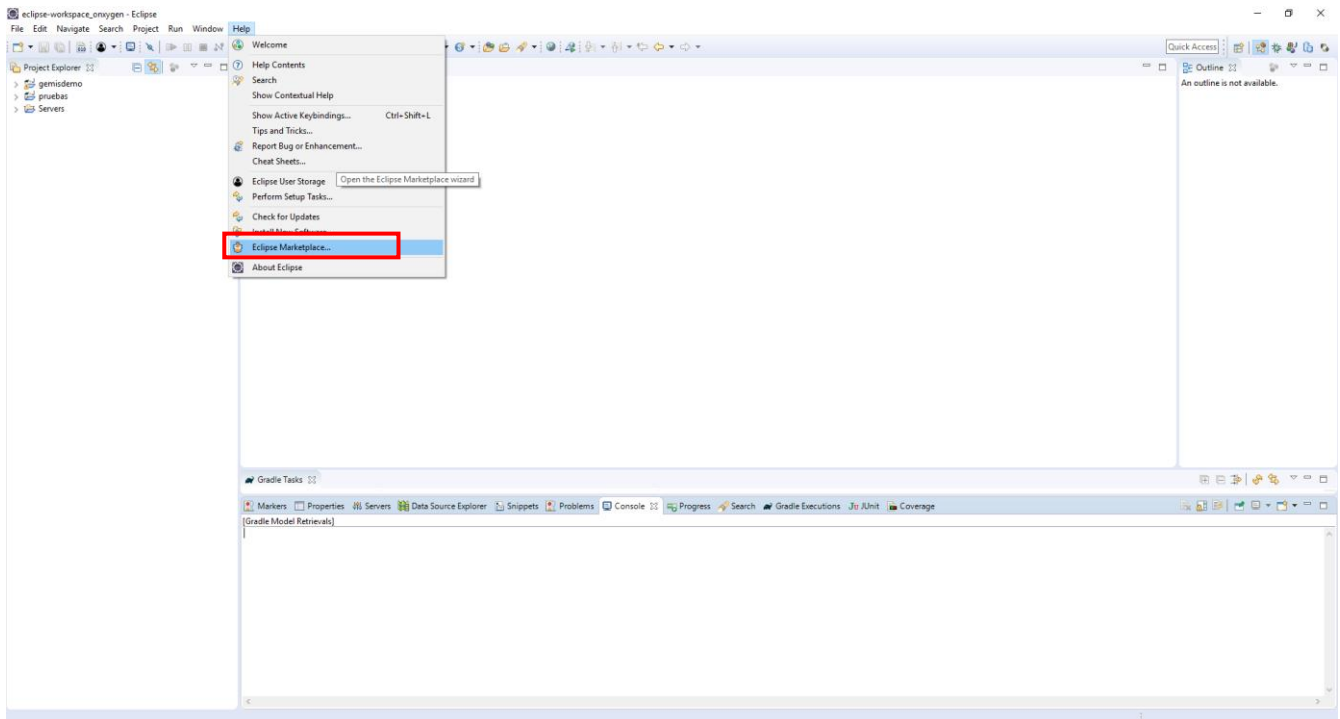
C:\Travail\03\_Workspaces\Mars2\_ConfigPref

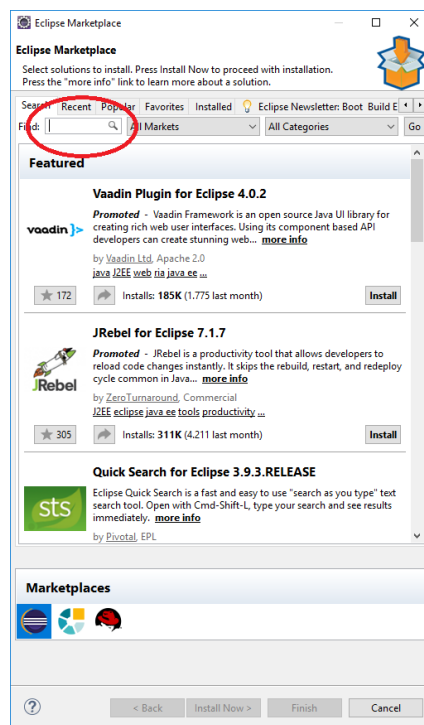
C:\Travail\03\_Workspaces\Mars2

## 4. Plugin: SonarLint

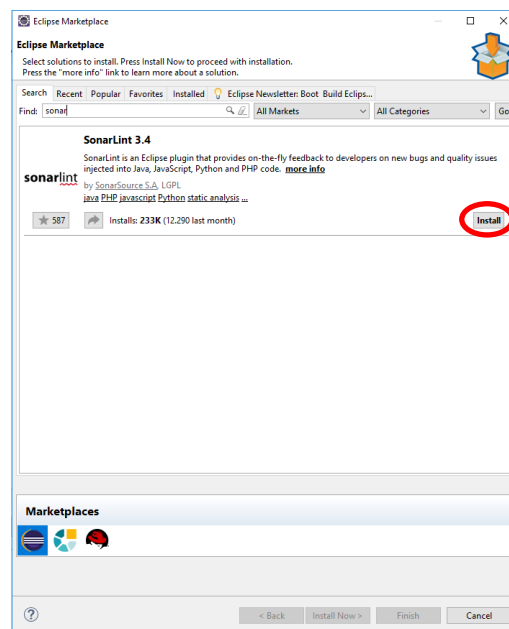
**SonarLint** es un *plugin* que añade depuración de código automática a alto nivel con el fin de construir un código robusto. Después de la comprobación sintáctica que realiza **Eclipse**, **SonarLint** inspecciona el código y avisa de los fallos de diseño que contiene.

## 4.1. Descargar e instalar el plugin SonarLint en Eclipse





En el cuadro de texto se introducirá el nombre del *plugin* que se quiera buscar, en este caso, **SonarLint**. Durante la realización de este manual se ha utilizado la versión 3.4.

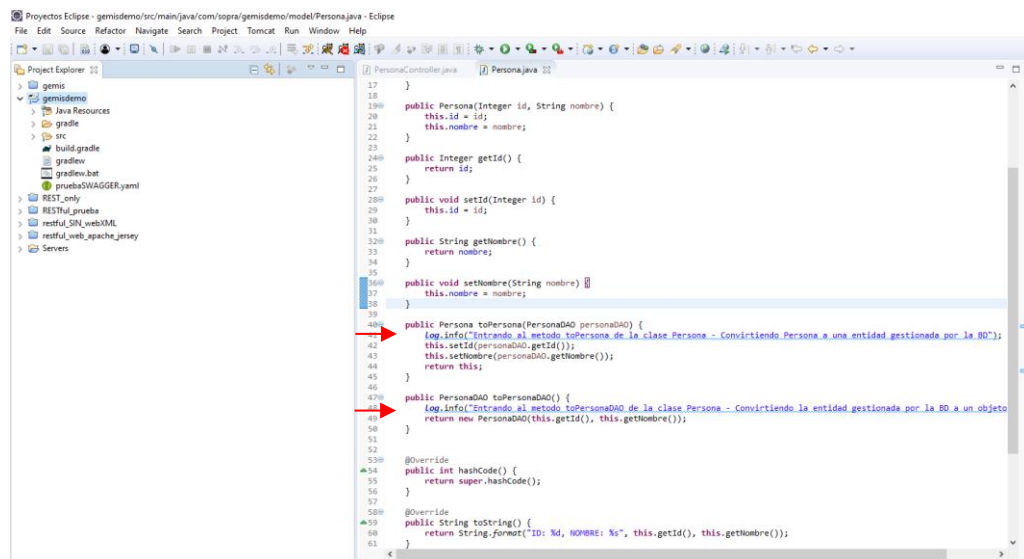


Cuando finalice la instalación del *plugin*, será necesario reiniciar Eclipse.

## 4.2. Uso de SonarLint

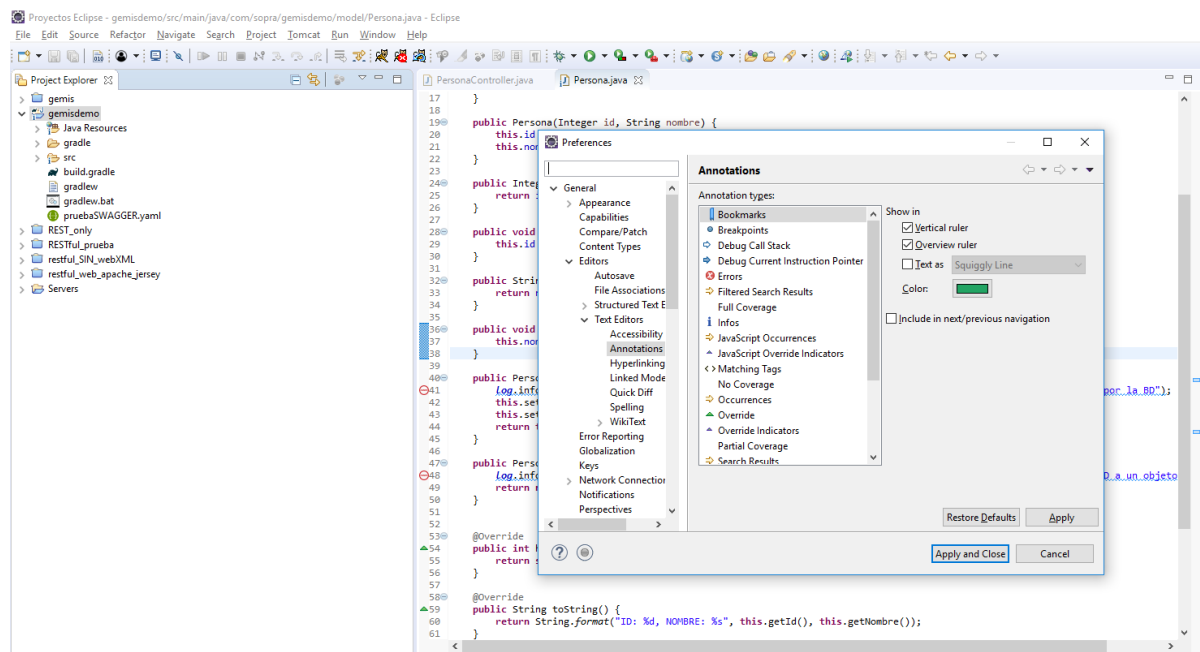
A continuación, se muestra un ejemplo en Eclipse tras instalar el *plugin*. Como se puede observar, las líneas que tienen errores detectados por **SonarLint** están subrayados en azul, pero si se quieren hacer más visibles, es posible configurarlo para que también muestre un icono de error a la izquierda.

Si se quiere añadir el proyecto de ejemplo ir a File>Import>Gradle>Existing Gradle Project y añadir la ruta del proyecto gemisdemo.



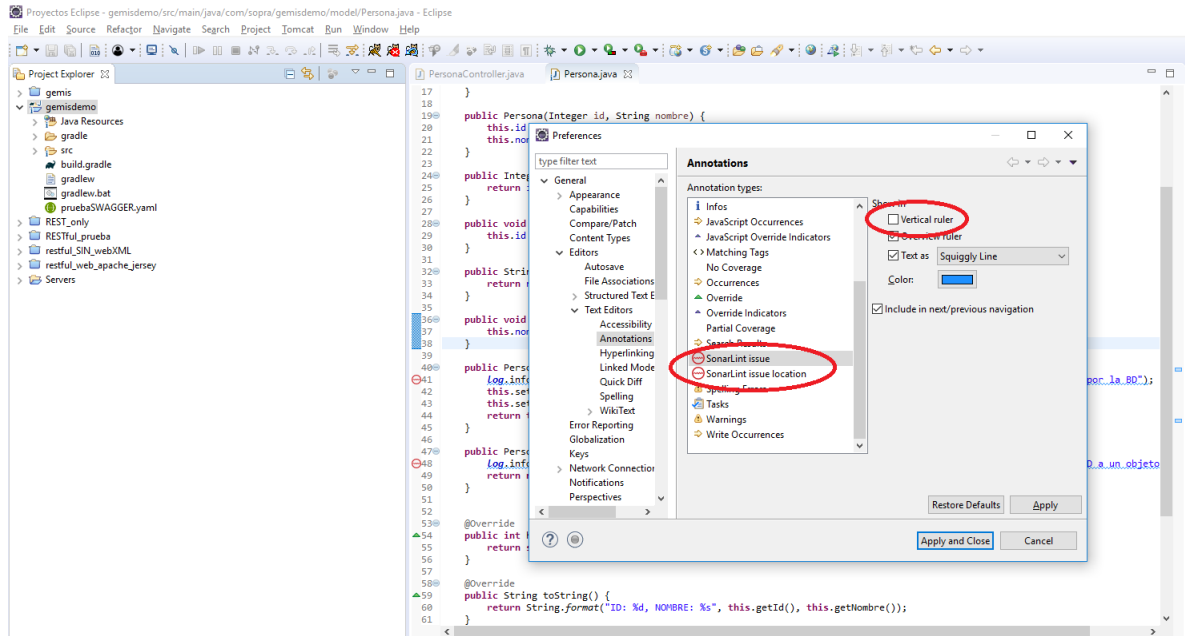
Si se desea también añadir el icono de error, se deberá ir a:

- Window > Preferences > General > Editors > Text Editors > Annotations

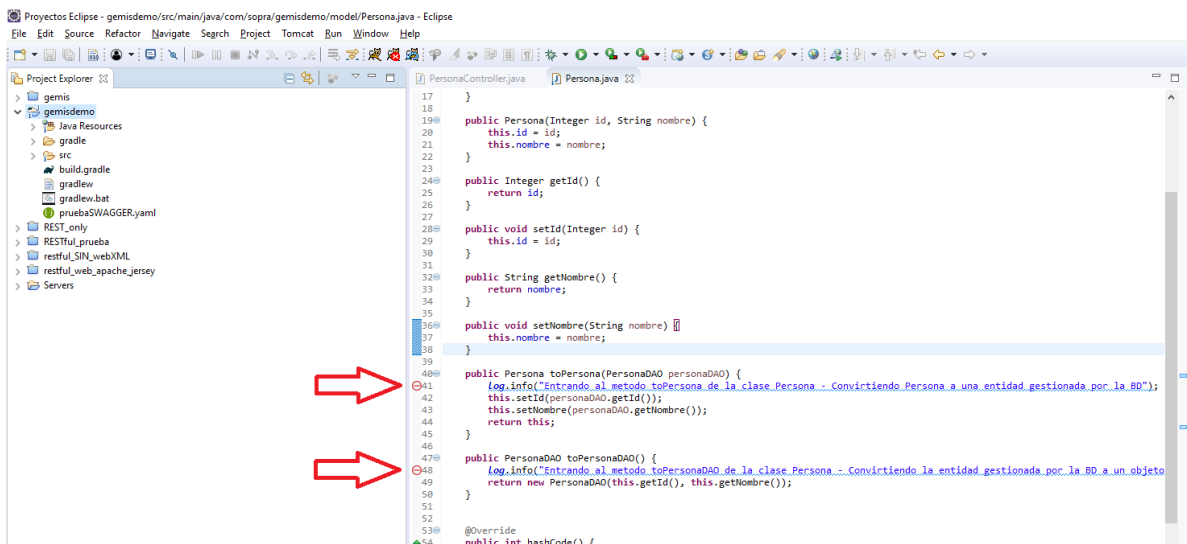




Dentro de este campo, se han de buscar las dos anotaciones de SonarLint (SonarLint issue y SonarLint issue location) y marcar la opción de Vertical Ruler, que por defecto viene desmarcada.

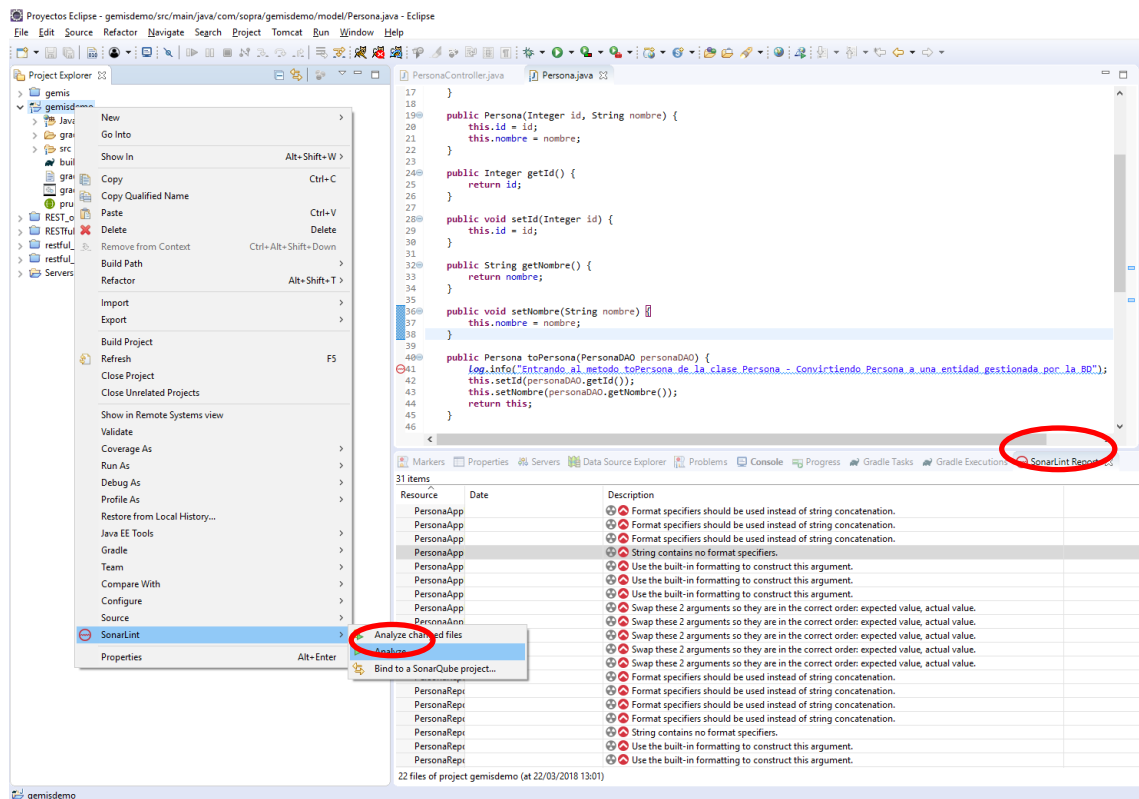


Ahora marca los errores en la barra izquierda con el símbolo característico de SonarLint además del subrayado por defecto.



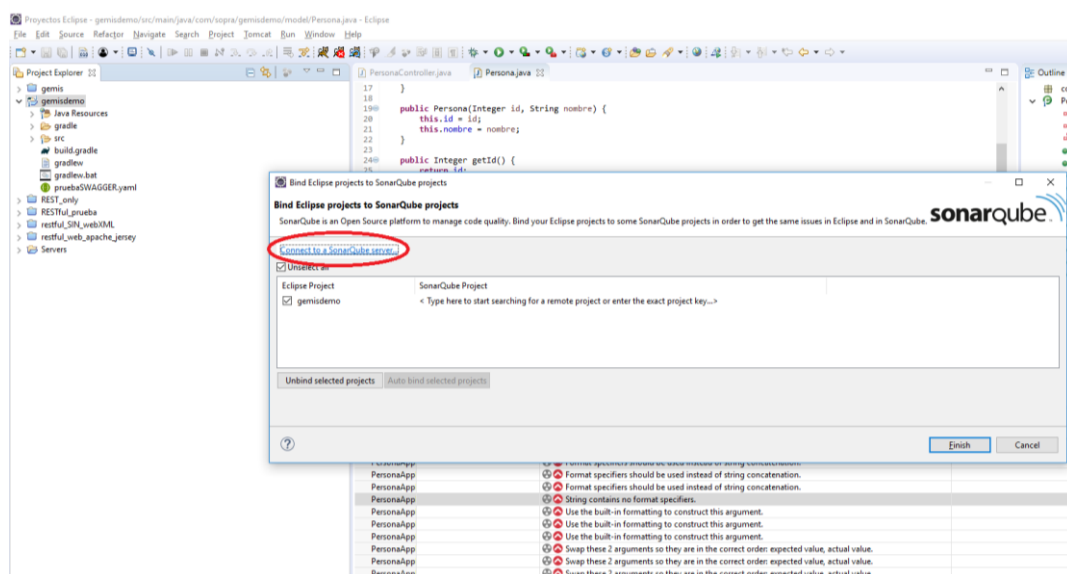
Si se quiere además que se evalúe todo un proyecto, SonarLint puede ser invocado:

- Botón derecho sobre el proyecto > SonarLint > Analyze



Por defecto un proyecto estará conectado a SonarCloud, es decir, a la nube. De esta manera se aplican las condiciones estándar. En un futuro sería posible preferir una conexión a un servidor en concreto que proporcione unas restricciones específicas a aplicar en una aplicación. Esto se configura:

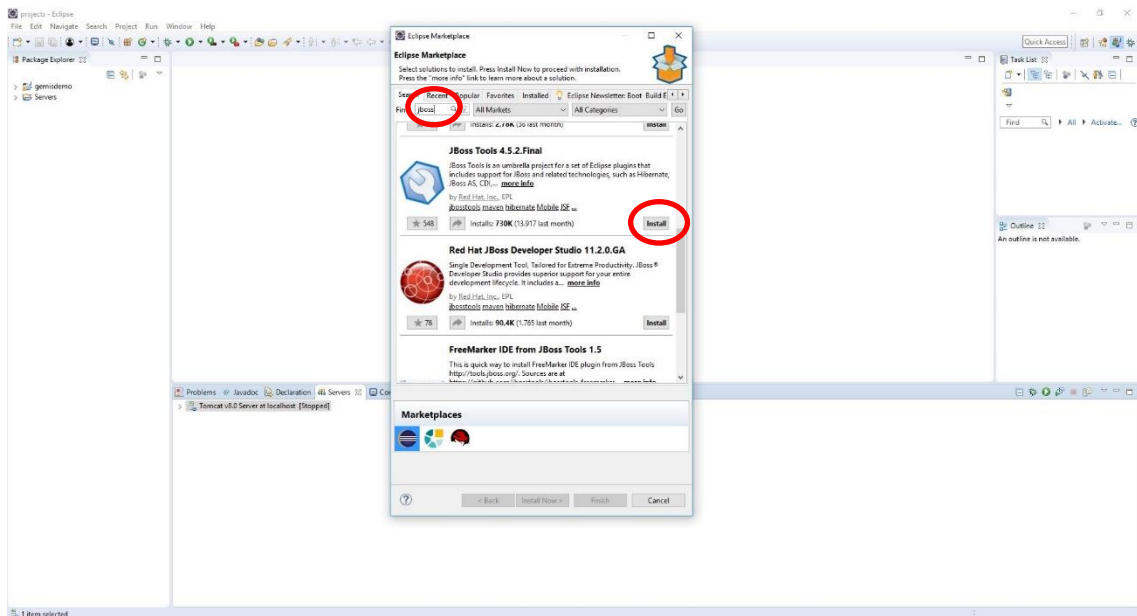
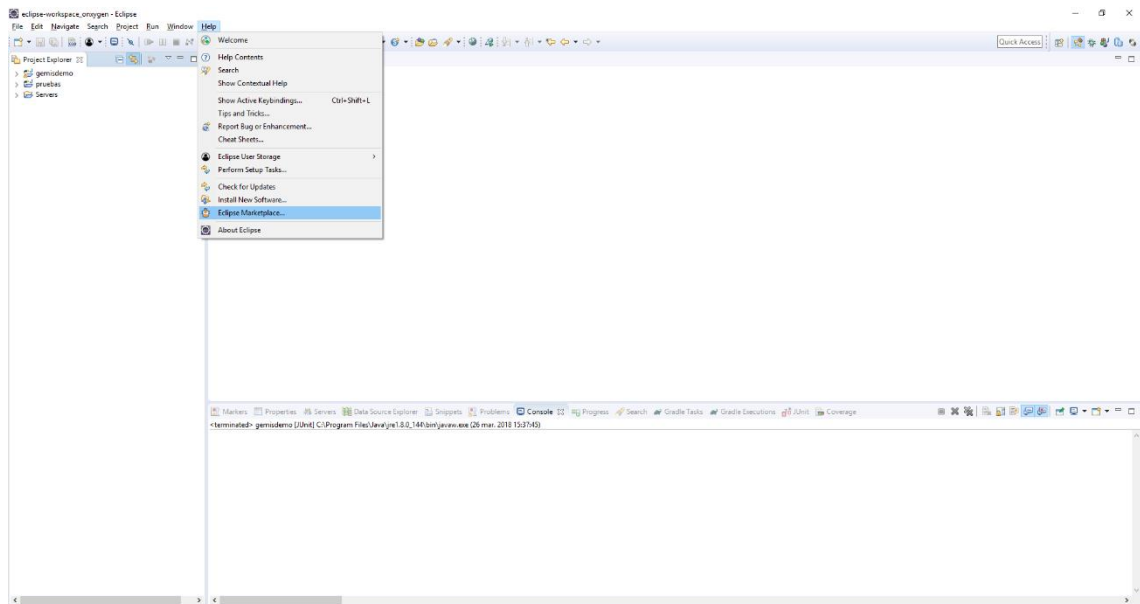
- Botón derecho sobre el proyecto > SonarLint > Bind Eclipse to SonarQube projects

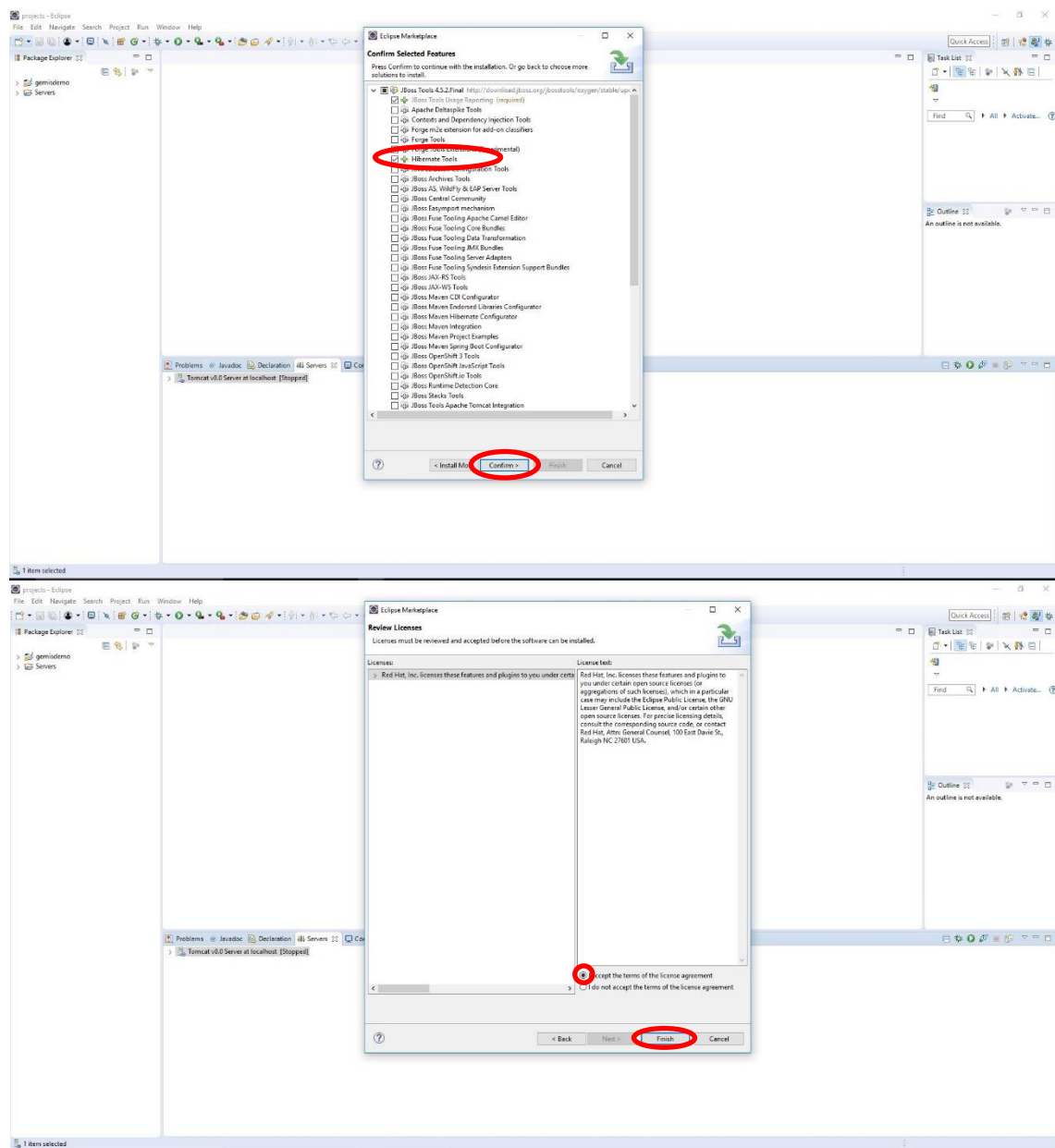


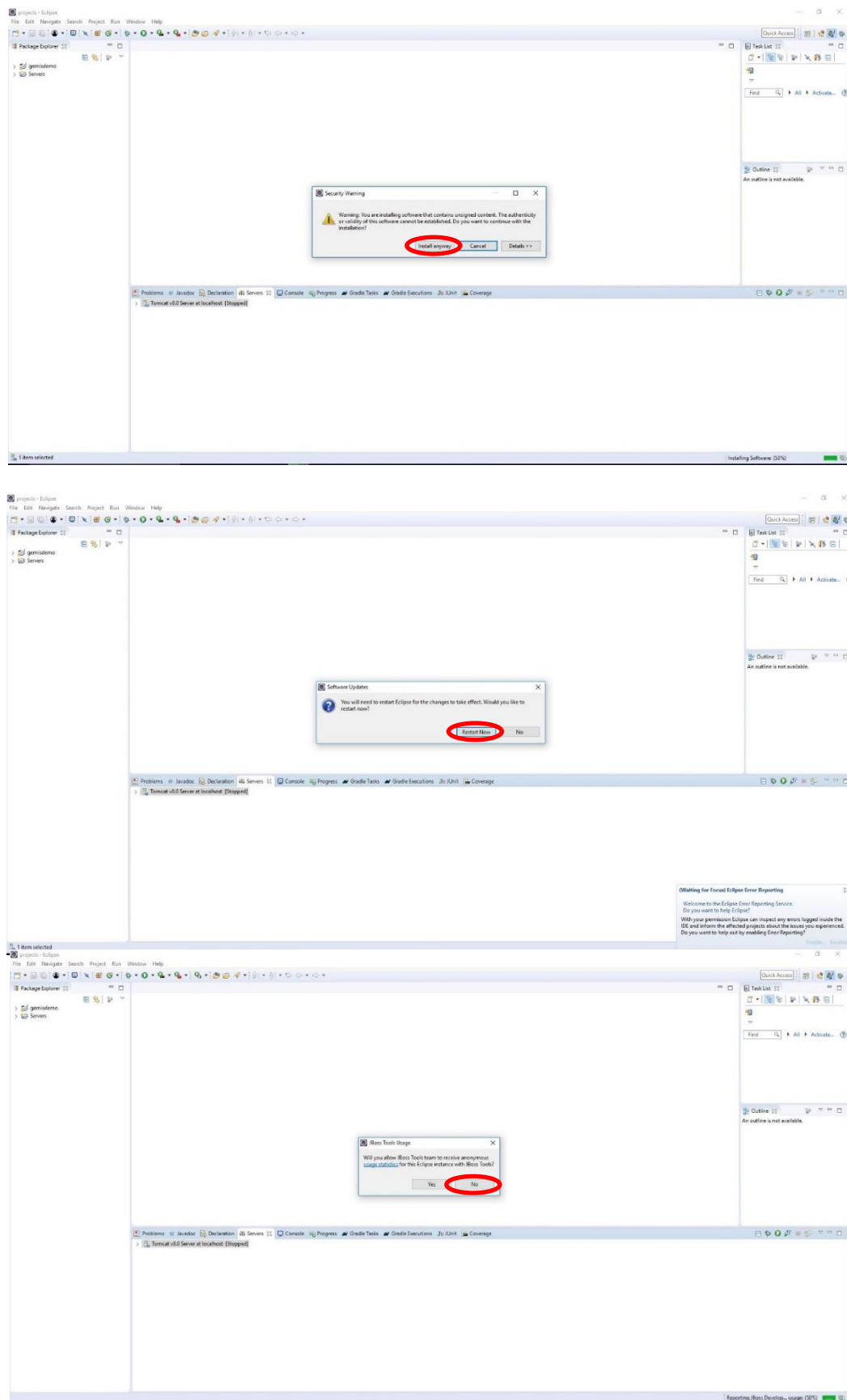
## 5. Plugin : Hibernate Tools

### 5.1. Descargar e instalar el plugin Hibernate Tools en Eclipse

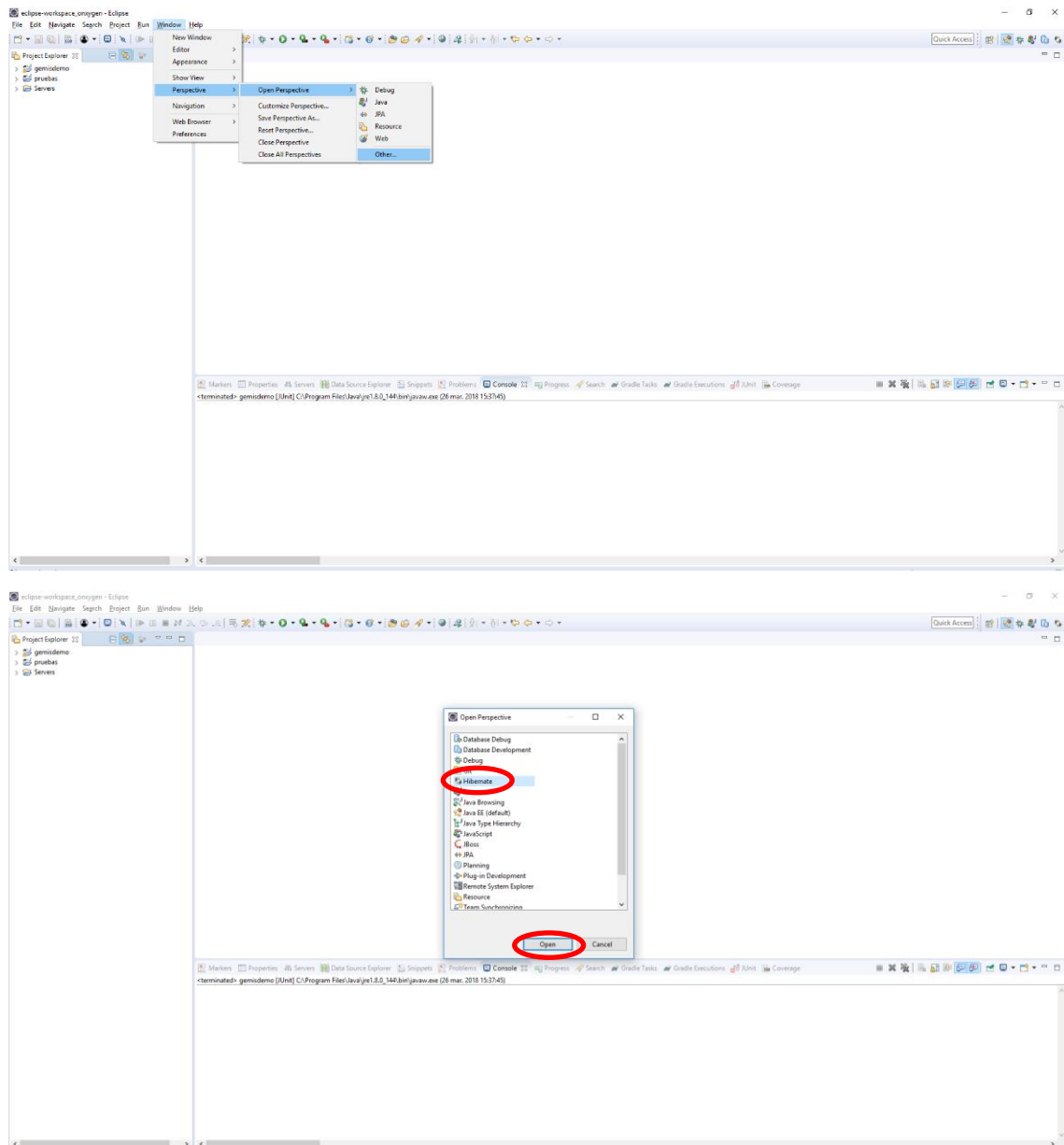
**Hibernate Tools** está dentro de **JBoss Tools**, que es un conjunto de *plugins* para **Eclipse** que tienen como finalidad ayudar a los desarrolladores de aplicaciones.





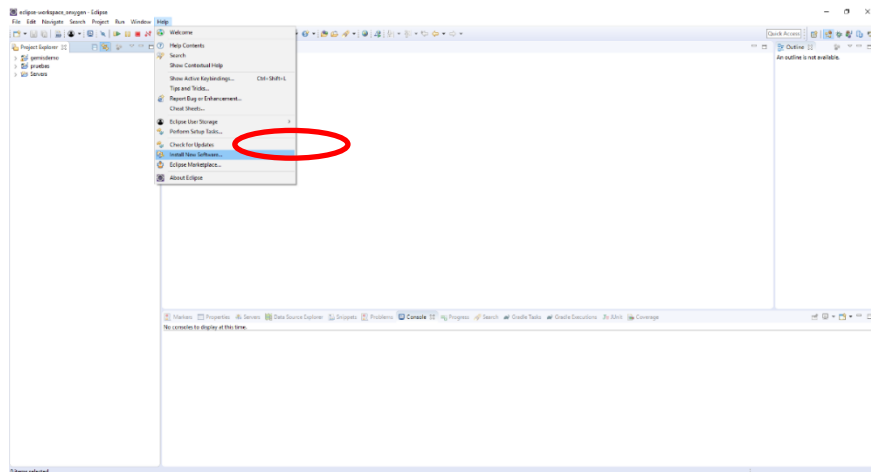


Finalmente, para acceder a las funcionalidades del *plugin*, se deberá acceder a la siguiente perspectiva en Eclipse:

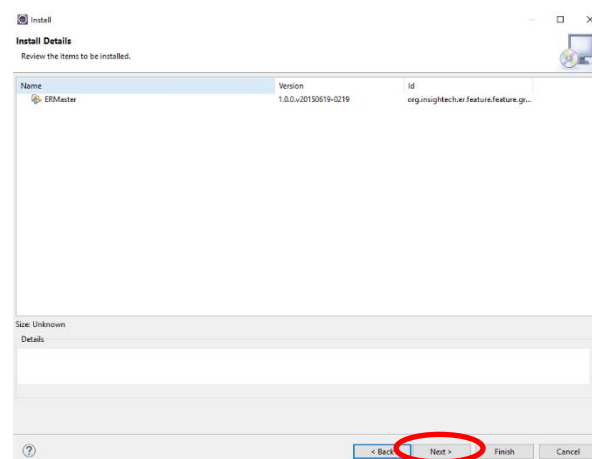
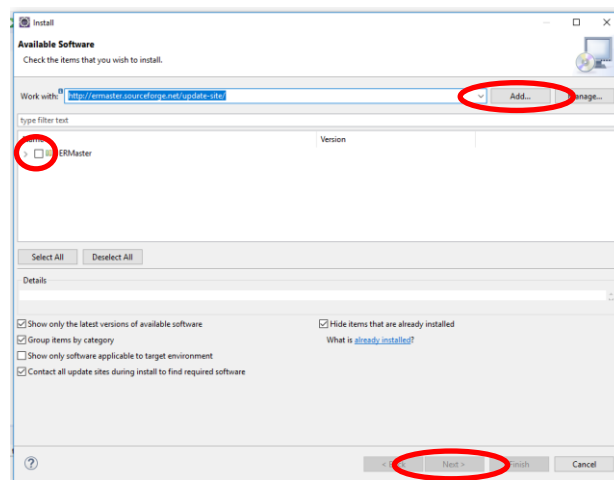


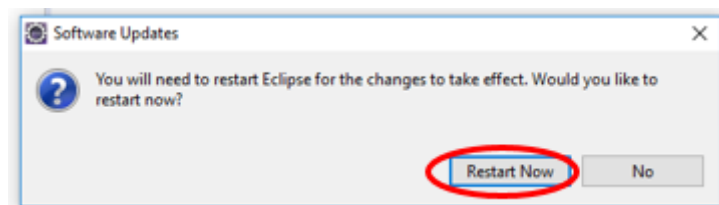
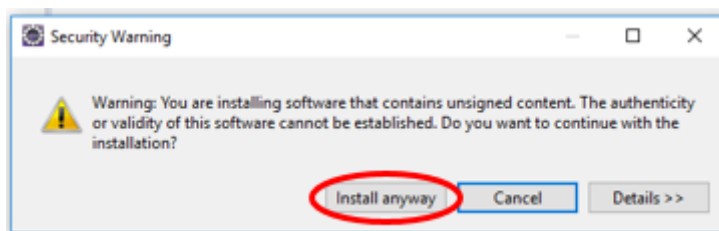
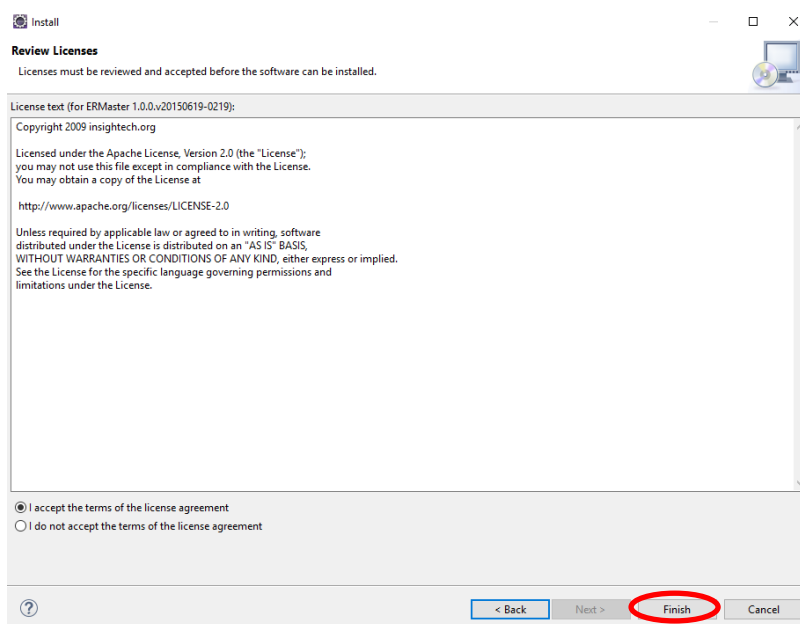
## 6. Plugin : ERMaster

### 6.1. Descargar e instalar el plugin ERMaster en Eclipse



El plugin se descargará de la siguiente URL: <http://ermaster.sourceforge.net/update-site/>





ERM Master es un *plugin* que contiene gran cantidad de funcionalidades, cuya documentación se puede encontrar en el siguiente enlace: <http://ermaster.sourceforge.net/content/manual/en/index.html>



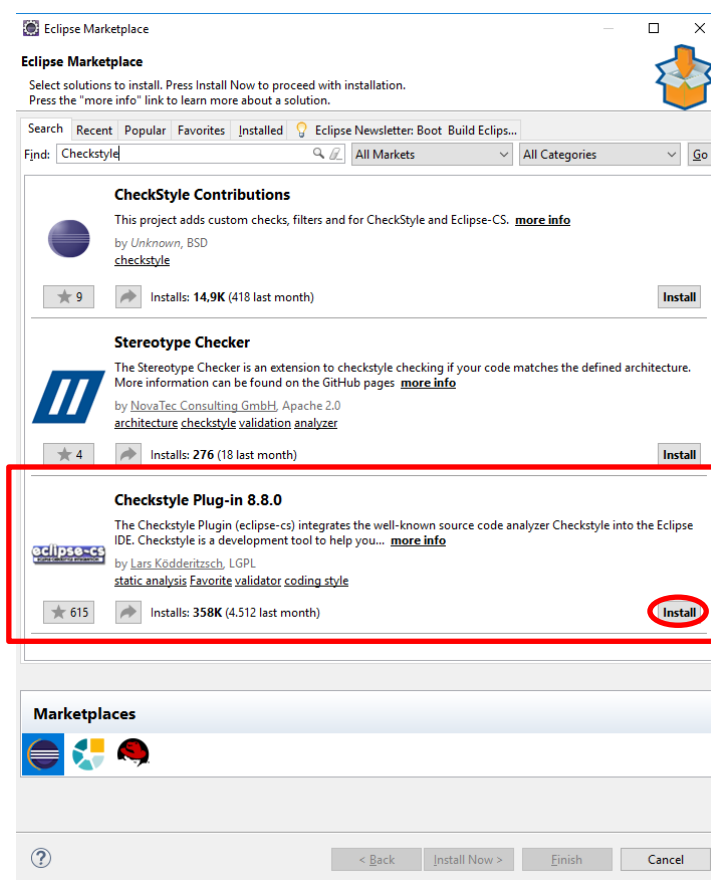
## 7. Plugin : Checkstyle

El *plugin* **CheckStyle** sirve para analizar el código según unos parámetros prefijados. Por defecto el *plugin* viene con opción de elegir entre las especificaciones de Google y Sun (ORACLE), pero elegiremos el primero.

### 7.1. Descargar e instalar el plugin Checkstyle en Eclipse

Para instalar el *plugin* **CheckStyle** es necesario ir a:

- Help > Eclipse MarketPlace

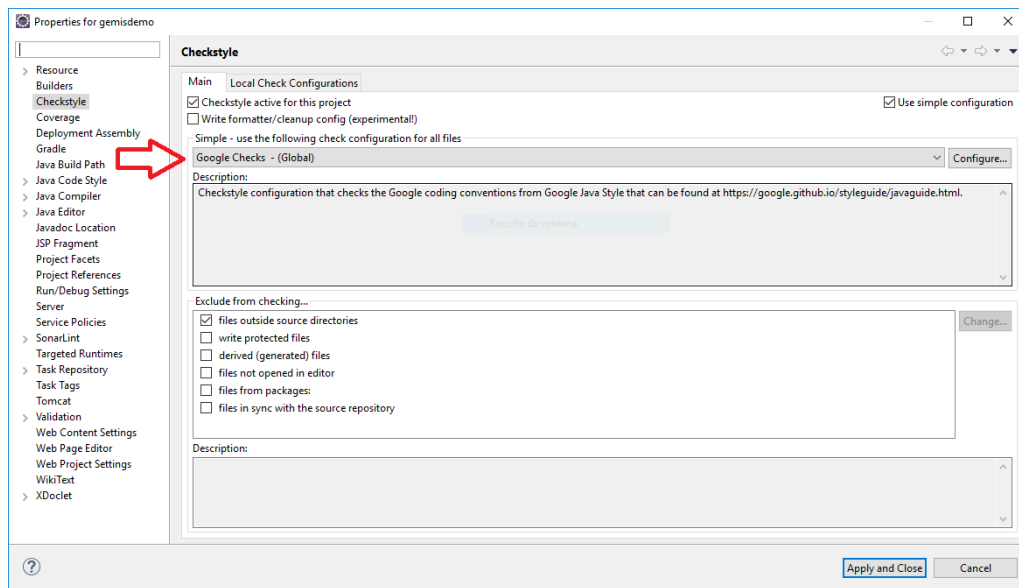


Una vez instalado el *plugin* se puede configurar, como ya se ha comentado anteriormente, si se quiere que analice el código según los valores de Google o Sun.

- Botón derecho sobre el proyecto > Properties > CheckList

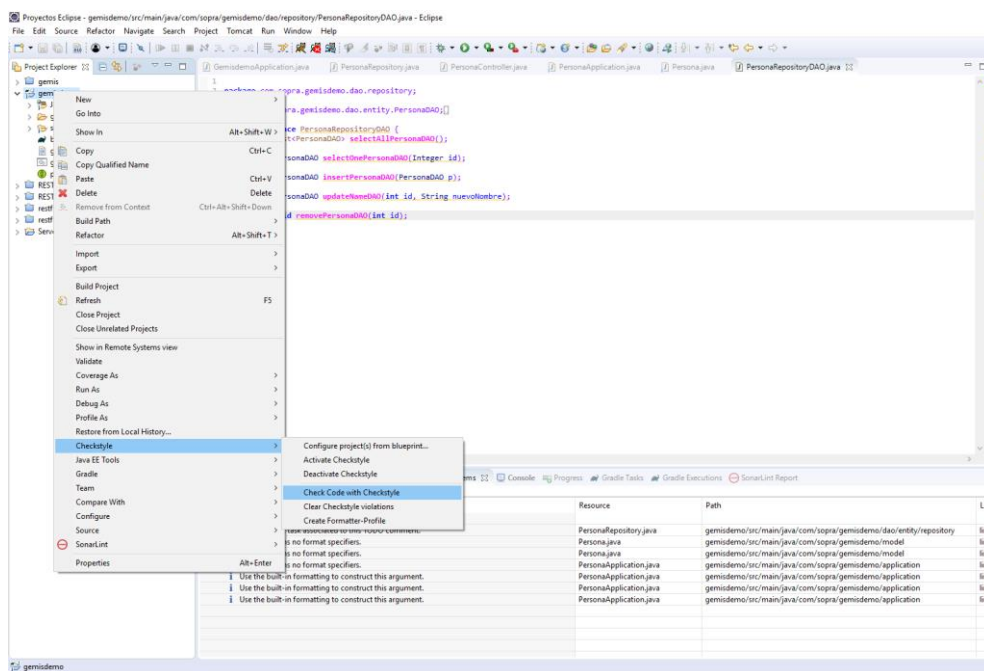
Ahí se puede configurar el *plugin*. En este ejemplo, en la pestaña central elegiremos cuál de las dos opciones vamos a usar (Google).



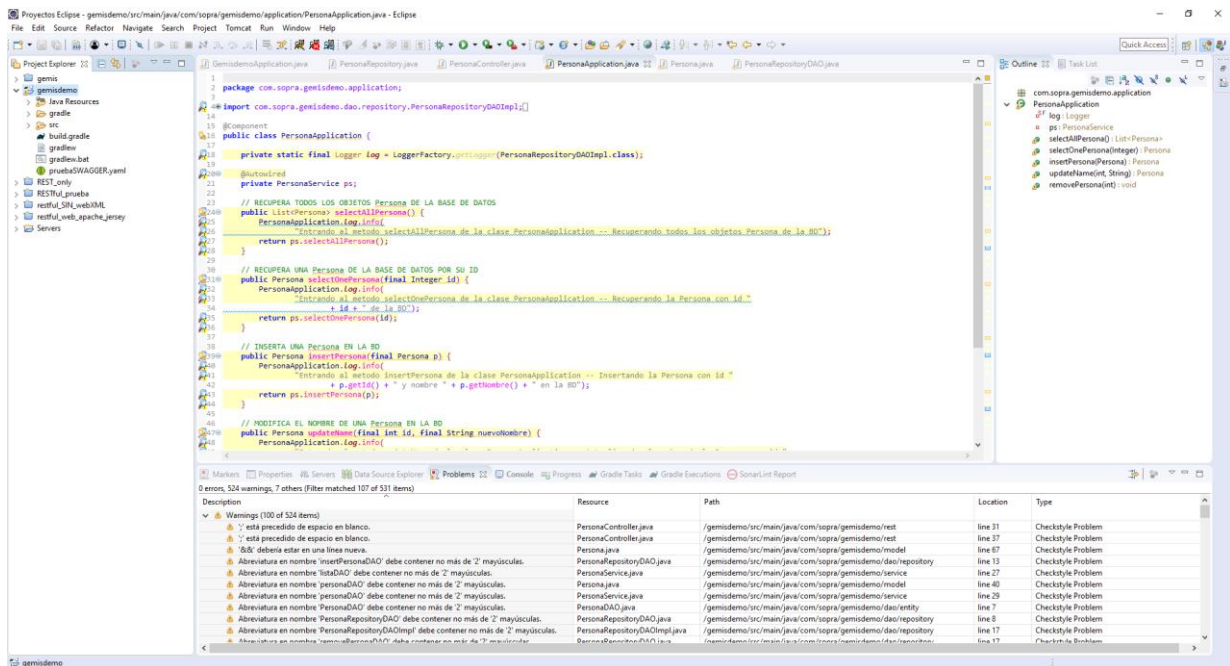


## 7.2. Uso de SonarLint

Para comprobar el código:



Un ejemplo de código resaltado se puede ver en la siguiente captura. Aclarar que únicamente indica los errores, así que se deben reparar manualmente.



Para desactivar las comprobaciones:

- Botón derecho sobre el proyecto > Properties > CheckList > Clear CheckStyle Violations

## 8. GIT

---

Para configurar GIT en local se tiene que hacer esto (entre otras cosas):

Para añadir el proxy de Sopra:

En consola de comandos

*git config --global http.proxy <http://vale.proxy.corp.sopra:8080/>*

(doble guión antes de "global"!)

Para añadir el origen de proyecto:

En consola de comandos

Navegar a la carpeta local (D:\Gemis)

*Git remote set-url origin <https://francisco-javier.chiarri@innersource.soprasteria.com/EcoleProjetValence/GemisProjet.git>*

En Amarillo el nombre de usuario de Innersource.



