

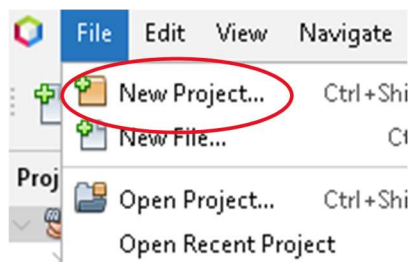
Contenido

Ejemplos de usos Netbeans	2
1. Crear Proyecto	2
2. Guardar proyecto	3
3. Versionar (Uso de Git, SVN, Mercurial)	4
4. Recuperar desde el repositorio	4
5. Instalar plugins.....	5
6. Desarrollar (Escribir código y administrar archivos)	7
7. Compilar.....	9
8. Ejecutar	10
9. Probar (manual y automática).....	10
10. Documentar	10
11. Generar documentación.....	11
12. Modelar (Diagramas UML).....	12
13. Depurar	12
14. Inspección de variables	12
15. Ejecución paso a paso.....	13
16. Comparar código	13
17. Refactorizar	13
18. Generar Código.....	13
19. Ingeniería inversa	14
20. Administrar Base de Datos.....	14
21. Ejecutar script sobre la base de datos	14

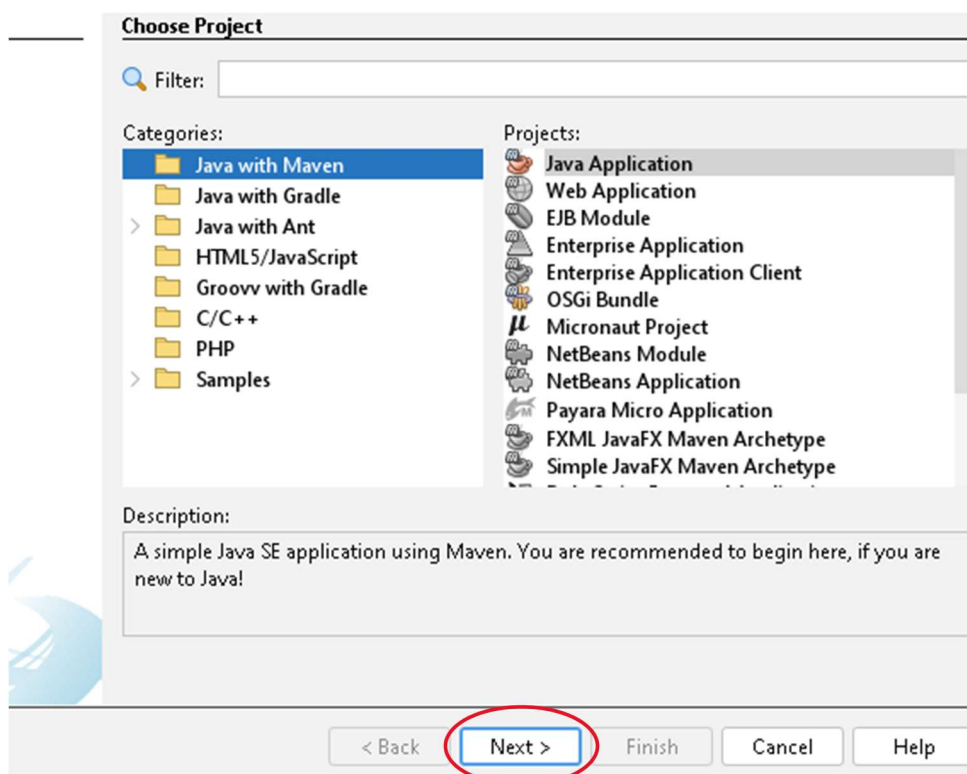
Ejemplos de usos Netbeans

1. Crear Proyecto

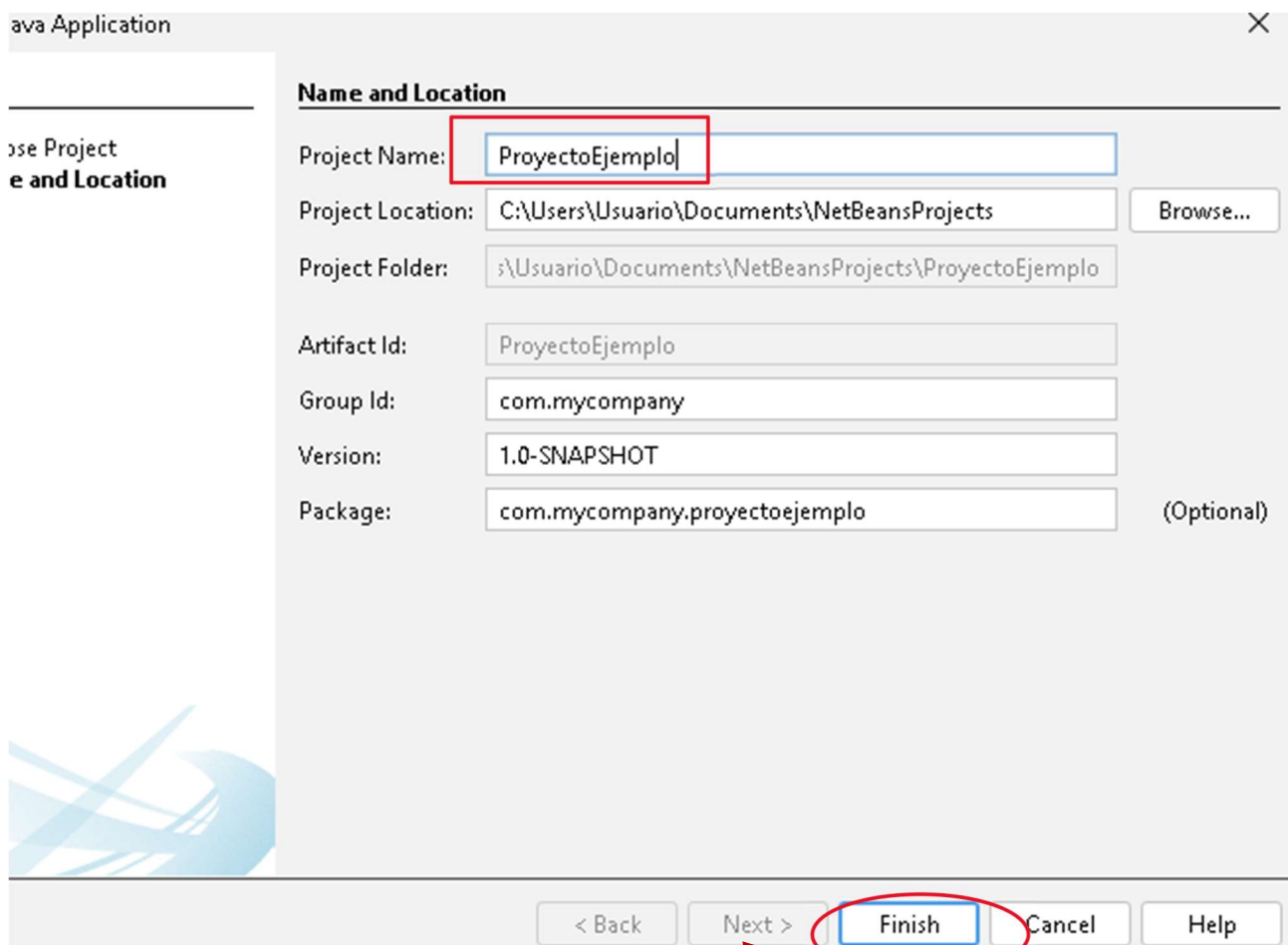
- Abre NetBeans.
- Ve a **File** → **New Project**.



- Selecciona **Category** (e.g., "Java" or "Maven") y **Projects** :Selecciona el tipo de proyecto, como "Java Application" (Ant) o "Maven Project".



- Pulsa **Next**, define el nombre y la ubicación del proyecto.

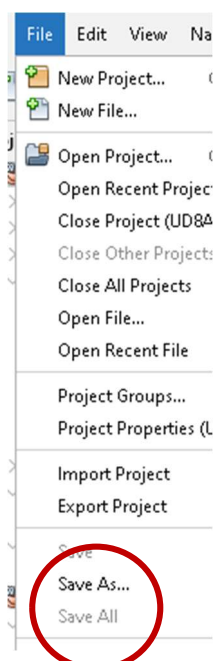


- Pulsa **Finish**.

2. Guardar proyecto

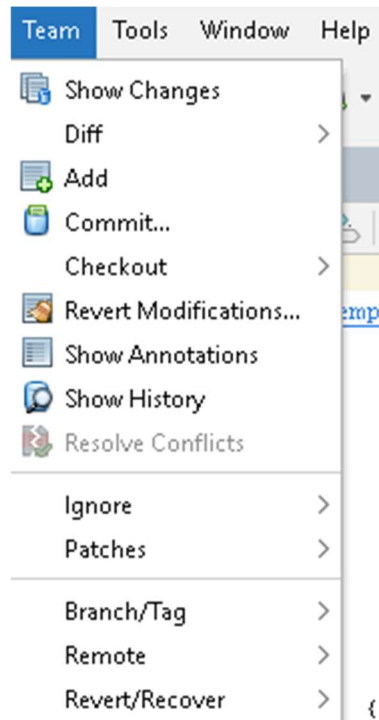
- **File** → **Save All**.

- O el tercer icono



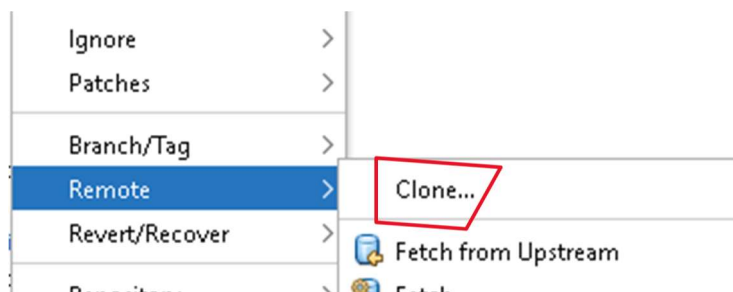
3. Versionar (Uso de Git, SVN, Mercurial)

- Ve a **Team** en la barra superior.
- Selecciona el sistema de control de versiones que usas (Git, SVN o Mercurial).
- Inicializa el repositorio o clona uno existente.
- Usa **Commit**, **Push**, y **Pull** para administrar cambios.

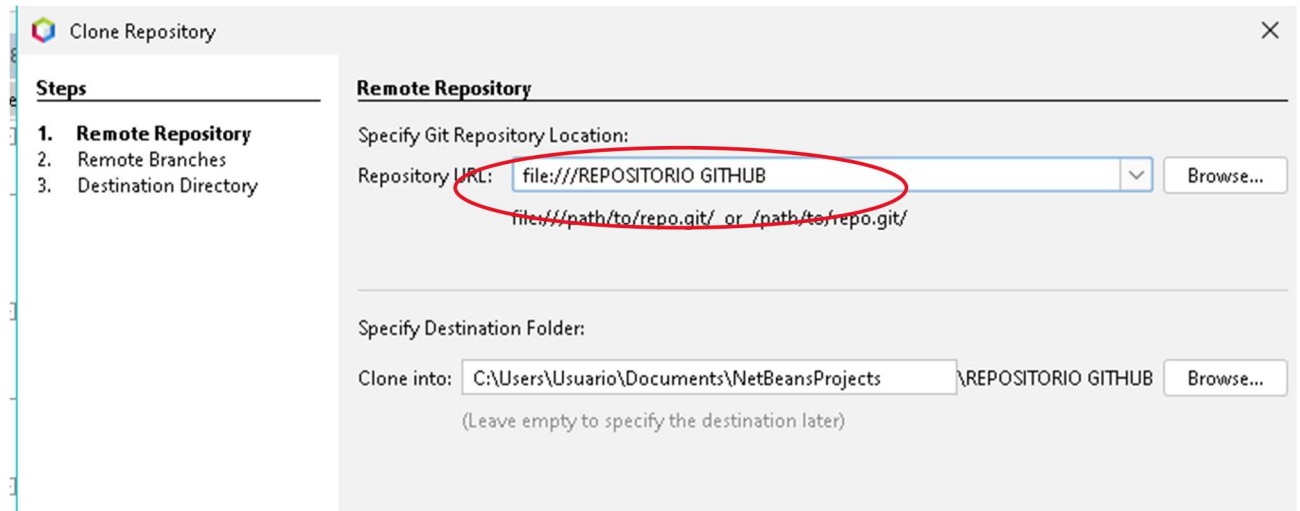


4. Recuperar desde el repositorio

- Ve a **Team** → **Remote** → **Clone**.



- Introduce la URL del repositorio y las credenciales.



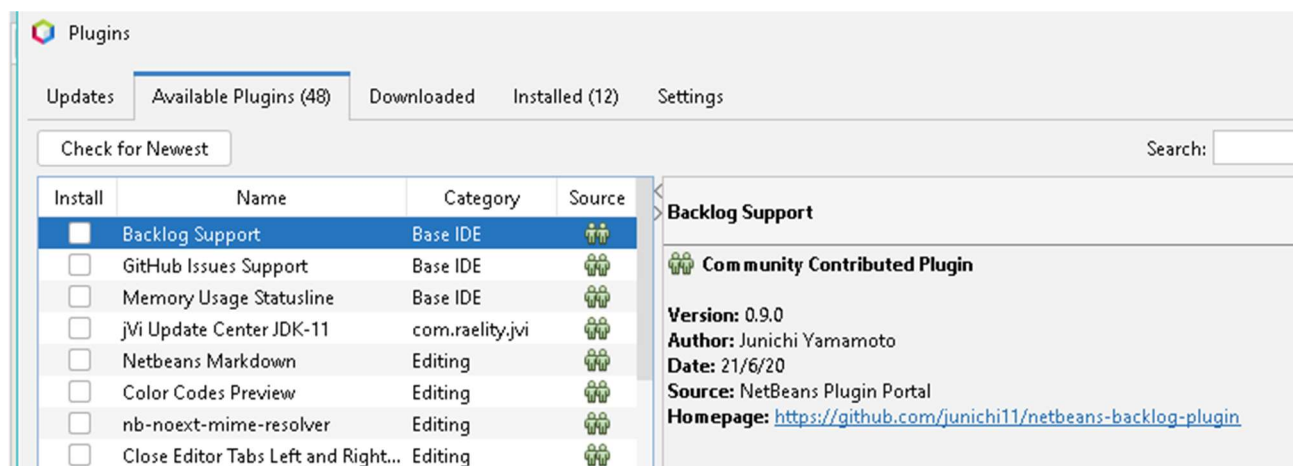
- Choose the branch and local directory.
- Click **Finish**.

5. Instalar plugins

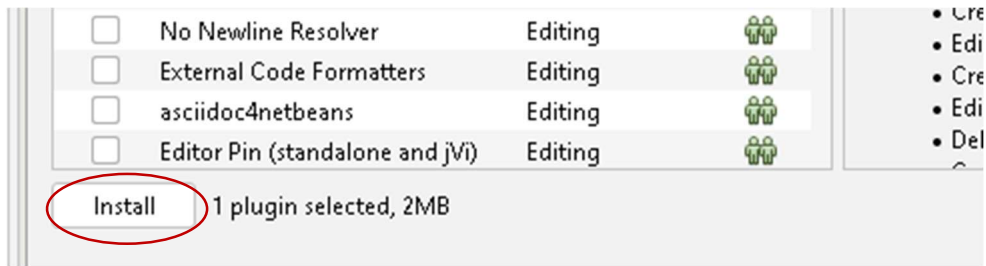
- Ve a **Tools** → **Plugins**.



- En la pestaña **Available Plugins** tab, busca el plugin deseado.



- Selecciona y pulsa **Install**.



- Reinicia NetBeans si es necesario.

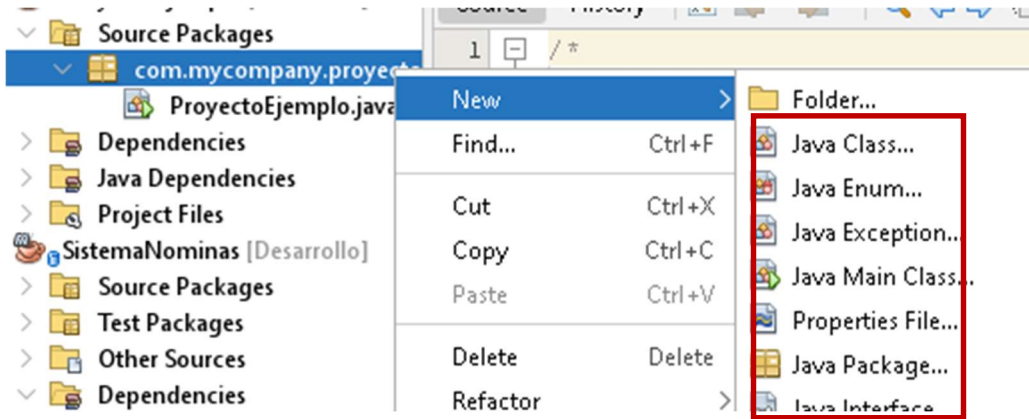
6. Desarrollar (Escribir código y administrar archivos)

Crear un Nuevo Proyecto

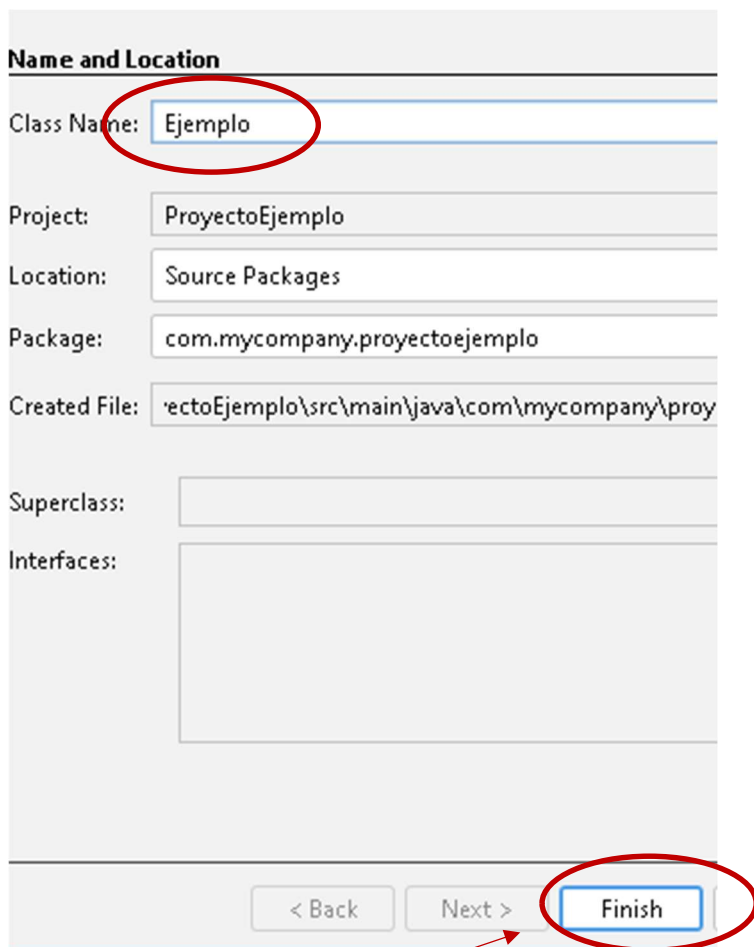
- Ve a **File** → **New Project**.
- En la sección **Categories**, selecciona **Java**.
- En la sección **Projects**, selecciona **Java Application**.
- Haz clic en **Next**.
- Escribe un Project Name y elige una ubicación.
- Haz clic en **Finish**.

Crear una Nueva Clase

- En la ventana **Projects**, busca tu proyecto.
- Haz clic derecho en la carpeta **com.mycompany**.
- Selecciona **New** → **Java Class**.



- Escribe el Nombre de la clase



- Haz clic en **Finish**.

Crear una Main Class

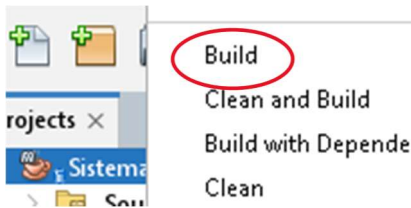
- Haz clic derecho en la carpeta **com.mycompany**.
- Selecciona **New** → **Java Main Class**.
- Escribe el Nombre de la clase
- Haz clic en **Finish**.

Ejecutar el Proyecto

- Haz clic derecho en **Main** class.
- Selecciona **Run File** o presiona **Shift + F6**.

7. Compilar

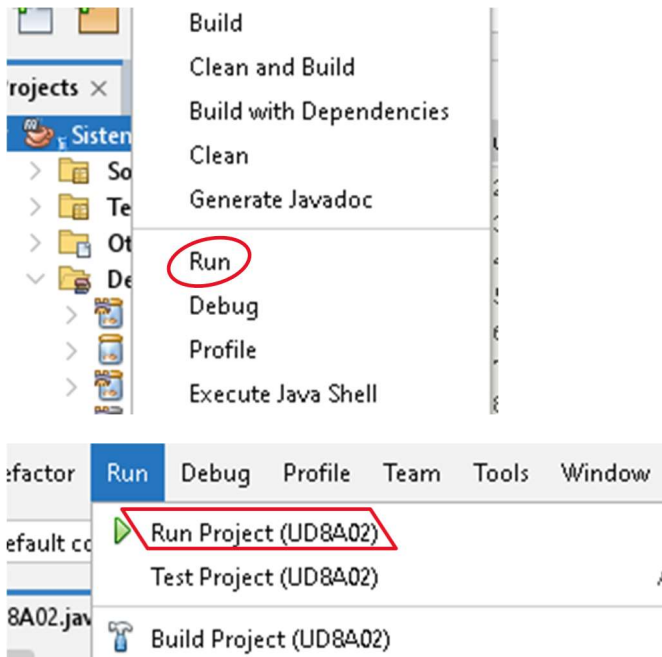
- Ve a **Run** → **Clean and Build Project**.



- NetBeans generará los archivos .class en la carpeta build.

8. Ejecutar

- Pulsa **F6** o ve a **Run** → **Run Project**.



9. Probar (manual y automática)

- **Manual:** Ejecuta la aplicación y verifica el comportamiento manualmente.
- **Automático:**

Crea pruebas JUnit: **File** → **New File** → **Unit Tests** → **JUnit Test**.

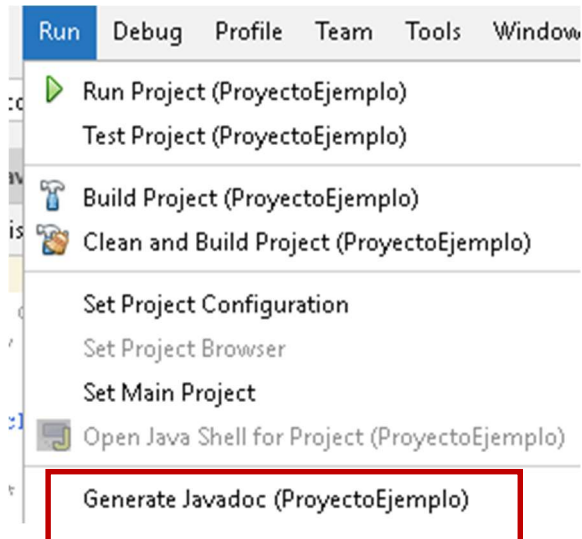
Ejecuta pruebas con **Shift + F6** o desde **Run** → **Test Project**.

10. Documentar

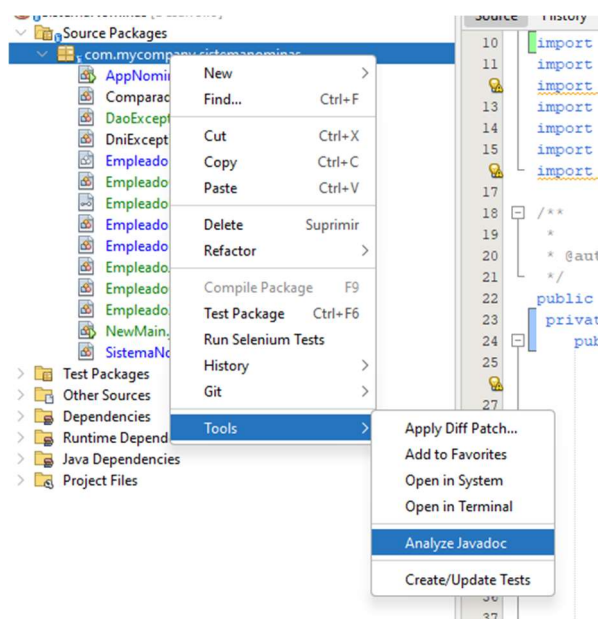
- Usa comentarios Javadoc:

11. Generar documentación

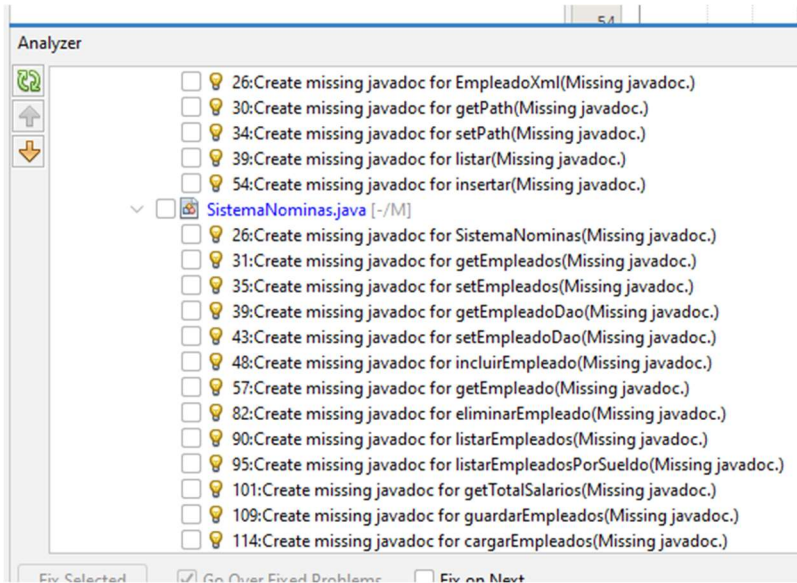
- Ve a **Run** → **Generate Javadoc**.



- NetBeans generará documentación en la carpeta del proyecto.
- Para que Netbeans te vaya insertando, los símbolos de comentarios. Haz clic con el botón derecho en la carpeta que contiene las clases y elegir tolos y Analyse Javadoc.



En la parte inferior en el cuadro que aparece se podrán seleccionar las clases que se quiera documentar.



12. Modelar (Diagramas UML)

- Instala el plugin **UML** desde **Tools** → **Plugins**.
- Ve a **File** → **New Project** → **UML**.
- Crea diagramas de clases, secuencia, etc.

13. Depurar

- Coloca un punto de interrupción haciendo clic en la línea de código.
- Ve a **Debug** → **Debug Project** or press **F5**.

14. Inspección de variables

- Empieza la depuración.
- Pasa el cursor sobre una variable para ver su valor.
- Usa la ventana **Variables** en **Window** → **Debugging** → **Variables**.

15. Ejecución paso a paso

- Usa **F7** para entrar en un método.
- Usa **F8** para avanzar línea a línea sin entrar en métodos.
- Usa **F5** para continuar hasta el siguiente punto de interrupción.

16. Comparar código

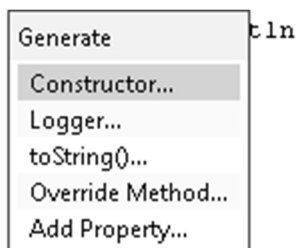
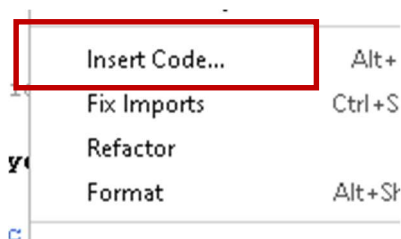
- Ve a **Team** → **Git** → **Diff to Revision**.
- Muestra las diferencias entre versiones.

17. Refactorizar

- Usa **Refactor** → **Rename** (Shift + F6) to change names.
- Usa **Refactor** → **Extract Method** para simplificar código.

18. Generar Código

- Usa **Alt + Insert** o botón derecho en el código y **insert Code** para generar constructores, getters y setters



19. Ingeniería inversa

- Instala el plugin UML.
- Importa una clase en el editor UML para generar diagramas automáticamente.

20. Administrar Base de Datos

- Ve a **Window** → **Services**.
- En **Databases**, agrega una conexión y administra datos.

21. Ejecutar script sobre la base de datos

- Conéctate a la base de datos en la ventana de **Services**.
- Haz clic derecho sobre la conexión y selecciona **Execute Command**.
- Escribe la consulta SQL y ejecútala.

22. Insertar Métodos de Dependencias

Haz clic en con el botón derecho en la dependencia y elegir download sources

