

Javadoc es una utilidad de Oracle para la generación de documentación de APIs en formato HTML a partir de código fuente Java. **Javadoc es el estándar para documentar clases de Java.** La mayoría de los IDEs utilizan javadoc para generar de forma automática documentación de clases. BlueJ también utiliza javadoc y permite la generación automática de documentación, y visionarla bien de forma completa para todas las clases de un proyecto, o bien de forma particular para una de las clases de un proyecto.

Veamos en primer lugar qué se debe incluir al documentar una clase:

a) **Nombre de la clase, descripción general, número de versión, nombre de autores.**

b) **Documentación de cada constructor o método** (especialmente los públicos) incluyendo: nombre del constructor o método, tipo de retorno, nombres y tipos de parámetros si los hay, descripción general, descripción de parámetros (si los hay), descripción del valor que devuelve.

Las variables de instancia o de clase no se suelen documentar a nivel de javadoc.

Para generar la documentación de un proyecto automáticamente hemos de seguir unas normas a la hora de realizar los comentarios dentro del mismo. Si las hemos seguido, en BlueJ disponemos de la opción del menú Tools --> Project Documentation, que nos abre la documentación del proyecto en un navegador web. Para ver la documentación de una clase específica en BlueJ, debemos abrir la ventana de código, y en la parte superior derecha cambiar la pestaña que pone "Source code" por la opción "Documentation".

Además de documentar las clases, **todo proyecto debería tener un archivo Leeme o README.** En BlueJ podemos acceder al readme.txt de proyecto haciendo doble click en el icono que representa una hoja en la parte superior izquierda del diagrama de clases. En el readme.txt sería adecuado incluir al menos: título del proyecto, descripción, versión, cómo arrancar el proyecto, autores e instrucciones para los usuarios. Haz doble click en el icono e introduce una descripción para tu proyecto.

Javadoc es una utilidad de Oracle para la generación de documentación de APIs en formato HTML a partir de código fuente Java. **Javadoc es el estándar para documentar clases de Java.** La mayoría de los IDEs utilizan javadoc para generar de forma automática documentación de clases. BlueJ también utiliza javadoc y permite la generación automática de documentación, y visionarla bien de forma completa para todas las clases de un proyecto, o bien de forma particular para una de las clases de un proyecto.

Veamos en primer lugar qué se debe incluir al documentar una clase:

a) **Nombre de la clase, descripción general, número de versión, nombre de autores.**

b) **Documentación de cada constructor o método** (especialmente los públicos) incluyendo: nombre del constructor o método, tipo de retorno, nombres y tipos de

parámetros si los hay, descripción general, descripción de parámetros (si los hay), descripción del valor que devuelve.

TAG	DESCRIPCIÓN	COMPRENDE
@author	Nombre del desarrollador.	Nombre autor o autores
@deprecated	Indica que el método o clase es obsoleto (propio de versiones anteriores) y que no se recomienda su uso.	Descripción
@param	Definición de un parámetro de un método, es requerido para todos los parámetros del método.	Nombre de parámetro y descripción
@return	Informa de lo que devuelve el método, no se aplica en constructores o métodos "void".	Descripción del valor de retorno
@see	Asocia con otro método o clase.	Referencia cruzada referencia (#método()); clase#método(); paquete.clase; paquete.clase#método()).
@version	Versión del método o clase.	Versión