

Adrián Tirado Ramos

1º De DAW

Entorno Desarrollo

TAREA DE INVESTIGACIÓN SOBRE
DOCUMENTACIÓN

ÍNDICE

PASOS PARA LA INSTALACIÓN	2
ACTUALIZAR UBUNTU	2
INSTALAR.....	2
PLANTILLAS DE CÓDIGO EN JAVADOC Y CADA UNO DE SUS COMPONENTES	3
RESUMEN JAVADOC Y PLANTILLAS DE CÓDIGO	3
COMPONENTES DE PLANTILLA DE CÓDIGO	3
<i>Cada uno de los componentes de la plantilla de código en el Javadoc contiene:</i>	3
<i>Etiquetas especiales.</i>	3

Pasos para la instalación

Actualizar Ubuntu

sudo apt update

sudo apt upgrade

Instalar

sudo apt install php-phpdocumentor

Verificamos la versión instalada si es la más nueva con:

phpdoc --version

Plantillas de código en Javadoc y cada uno de sus componentes

Resumen Javadoc y plantillas de código

Javadoc es una herramienta para crear documentación a partir del código fuente de Java. Las plantillas de código en Javadoc son comentarios especiales escritos directamente en el código fuente de Java para documentar sus elementos como clases, métodos, variables, etc.

Estos comentarios están escritos en un formato HTML simplificado y pueden contener etiquetas especiales reconocidas por Javadoc.

Componentes de plantilla de código

Cada uno de los componentes de la plantilla de código en el Javadoc contiene:

- **Comentarios de clase** (`/* */`): Estos comentarios se colocan directamente encima de la declaración de clase y proporcionan información sobre la clase en sí, como su propósito, su autor, versiones, etc.
- **Comentarios de método** (`/* */`): Ubicados directamente encima de la declaración del método, estos comentarios describen qué hace el método, qué parámetros acepta, qué devuelve y qué excepciones puede generar.
- **Comentarios de campo o variable** (`/* */`): Estos comentarios se colocan directamente encima de la declaración de variable o campo y proporcionan información sobre su propósito y cómo se puede utilizar.

Etiquetas especiales.

Los comentarios de Javadoc utilizan etiquetas especiales precedidas por un símbolo @ para proporcionar información adicional.

Las etiquetas comunes:

- **@author:** El autor de la clase.
- **@version:** La versión de la clase.
- **@see:** Referencias a otras clases o miembros.
- **@param:** Descripción de los parámetros de un método.
- **@return:** Descripción de lo que devuelve el método.
- **@throws:** Las excepciones que puede lanzar el método.
- **@deprecated:** Indica que el método está obsoleto.