Javadoc :

Java doc est permettant de créer une documentation d'API en format HTML depuis les commentaires présents dans un code source en Java.

Il est possible de décrire un paquetage avec le fichier package-info.java, dans lequel on inclut un commentaire Javadoc.

Format :

/\*\*

\* @author : Nom du développeur

\* @deprecated : Marque la méthode comme déconseiller

\* @exception : Documente une exception lancée par une méthode

\* @param : Définit un paramètre de méthode. Requis pour chaque paramètre

\* @return : Documente la valeur de retour

\* @see : Documente une association à une autre méthode ou classe

\* @since : Précise à quelle version de la SDK/JDK une méthode a été ajoutée à la classe

\* @throws : Documente une exception lancée par une méthode

\* @version : Donne la version d'une classe ou d'une méthode.

\*/

Exemple :

*/\*\**

*\* Classe de gestion d'étudiants*

*\* @author John Doe*

*\* @version 2.6*

*\*/*

**public** **class** **Etudiant**

Maven JXR Plugin

Une bibliothèque permettant d'analyser un ensemble de fichiers source Java et de produire de la documentation au format HTML à la Javadoc.

Caractéristique :

* Prend en charge le JDK 1.4+
* Outil complémentaire pour Javadoc
* Configuration facile pour la couleur, le style ou le modèle
* Entièrement intégré avec Maven 2/3

Exemple :

Transformer des paquets Java :

JXR gère plusieurs options comme un encodage d'entrée

JXR jxr = new JXR ();

jxr.setDest ("/ target / jxr");

jxr.setLog (nouveau DummyLog ());

jxr.xref (Collections.singletonList ("/ src / main / java /"), "templateDir",

"WindowsTitle", "DocTitle", "Bas");

La structure JXR générée devrait être la suivante :

/ target / jxr

| - allclasses-frame.html

| - index.html

| - overview-frame.html

| - overview-summary.html

...

Les Procédure stocker :

CallableStatement : L'interface CallableStatement, permet de faire appel aux procédures stockées et aux fonctions de manière standard

Pour utiliser CallableStatement il faut : méthodesprepareCall de Connection.

Le premier argument de ces méthodes est une chaîne de caractères définissant l'instruction SQL. Cette chaîne de caractères utilise la syntaxe d'échappement et peut avoir deux formes.

Exemple d’utilisation :

# String sql = "{call nomDeLaProcedure[(?, ?, ...)]}";

# //[(?, ?, ...)] sont les éventuels arguments de la procédure

# 

# //ces arguments peuvent être de type IN, OUT ou INOUT

# CallableStatement statement = connection.prepareCall(sql);