

Documentations des programmes

Javadoc - Doxygen - Ccdoc - Phpdoc

Lucas Chabalier - Lucas Roulette

2 Octobre 2014

Introduction

Javadoc

Doxygen

Conclusion

Table des matières

Introduction

Javadoc

Présentation

Utilisation

Exemple

Générateurs de documentation semblables à Javadoc

Doxygen

Présentation

Exemple

Conclusion

Pourquoi documentons-nous notre code ?

Un des points importants de la programmation est le fait de commenter son propre code. Cela permet de :

- ▶ De comprendre ce que l'on code
- ▶ Que les autres développeurs puissent comprendre votre code
- ▶ Parce que c'est toujours bien de commenter!

Exemple quand il n'y a pas de documentation



Le
programmeur
intello

Un programmeur crée un programme Java révolutionnaire possédant de nombreuses classes et méthodes. Un autre programmeur souhaite approfondir ce code.

Mais malheureusement le code source est compliqué, les noms des classes, méthodes ou encore paramètres ne permettent pas de connaître le fonctionnement du code.

Si l'auteur avait créé une documentation javadoc, ce problème n'aurait pas existé !

Présentation

Historique

L'outil Javadoc a été développé par Sun Microsystems (racheté par Oracle en 2009), l'entreprise à l'origine de Java. C'est le standard de la documentation en Java, et ce depuis Java 1 (1995). Bon nombre d'EDI, tels que NetBeans ou Eclipse génèrent automatiquement une Javadoc, grâce à cet outil.



Introduction

Javadoc

Doxygen

Conclusion

Présentation

Caractéristiques principales

Les principales caractéristiques de l'outil Javadoc sont :

- ▶ Utilisable uniquement pour Java.
- ▶ Standard de la documentation professionnelle.
- ▶ Génère une documentation en HTML.
- ▶ Inclus dans le SDK de Java, donc utilisable facilement et multi-plateforme.

Utilisation

Informations que peut contenir Javadoc

La documentation Javadoc générée contiendra des informations sur

- ▶ Les classes et interfaces :
 - ▶ Leurs hiérarchies.
 - ▶ Leurs déclarations.
 - ▶ Les résumés et détails de leurs constructeurs et méthodes.
- ▶ Les énumérations
 - ▶ Leurs déclarations et descriptions.
- ▶ L'organisation des packages et classes en arbre
- ▶ Les API déconseillées
 - ▶ Si le code utilise des API déconseillées, afin de les remplacer lors de futures versions.

Exemple

Attrapons des Pokémons

On va lancer javadoc sur un programme Java collectionnant les Pokémons.

Commande à utiliser

```
javadoc -d <repertoire de sortie> <nomFichier.java>
```

Utilisation

Les différents Flags

Les flags permettent de spécifier des informations pour la construction de la documentation.

- ▶ *@author* Nom du développeur.
- ▶ *@param* Définit un paramètre de méthode.
- ▶ *@return* Documente la valeur de retour d'une méthode.
- ▶ *@see* Indique une association à une autre méthode/classe.
- ▶ *@since* Précise à quelle version une méthode a été ajoutée.
- ▶ *@version* Donne la version d'une classe ou d'une méthode.
- ▶ *@deprecated* Marque la méthode comme dépréciée.
- ▶ *@exception* Documente une exception lancée par une méthode.

Ccdoc et Phpdoc

Ccdoc et Phpdoc sont deux générateurs de documentation pour le C++ et le PHP. Ils sont fortement inspirés de l'outil Javadoc et permettent aussi de générer une documentation en HTML et identique à celle que peut générer Javadoc.

Introduction

Javadoc

Doxygen

Conclusion

Présentation

Petit historique



Doxygen est un générateur de documentation libre (license GNU General Public License). Il a été créé par Dimitri van Heesch, codé en C++ et publié pour la première fois en Octobre 1997. La dernière version stable est la 1.8.7. Il est donc plus récent que Javadoc, et fait partie des générateurs de documentation les plus connus et utilisés.

Présentation

Caractéristiques et Utilisation

Doxygen est souvent opposé à Javadoc, et a quelques avantages comparé à ce dernier :

- ▶ Il supporte beaucoup de langages, dont le C++, C, C, Objective-C, Java, Perl, Python, Fortran and PHP.
- ▶ La documentation peut être générée dans de nombreux formats dont HTML, CHM, RTF, PDF, \LaTeX , PS, ou même de pages du manuel (avec la commande man).

En ce qui concerne l'utilisation, les flags sont les mêmes que pour Javadoc, mis à part certains flags qui sont utilisables uniquement pour un langage spécifique.

Exemple

Utiliser Doxygen

On peut utiliser Doxygen de 2 façons :

- ▶ En ligne de commande, en générant un fichier de configuration qu'on peut éditer par la suite.
- ▶ Avec doxywizard, en interface graphique.

Introduction

Javadoc

Doxygen

Conclusion

Conclusion

Ce qu'il faut retenir de cette présentation

- ▶ Peu importe le générateur utilisé, ils ont tous le même fonctionnement, c'est assez pratique.
- ▶ Qu'un générateur permet de documenter son code, que ce soit pour soi ou pour les autres, ce qui le rend indispensable.
- ▶ Qu'on ne voit pas forcément l'intérêt de documenter de petits programmes, mais que sur de gros projets, cela prend tout son sens (comme Java).
- ▶ Il est important de prendre l'habitude de bien commenter ses programmes, ainsi que d'utiliser des bonnes conventions de nommage des variables, classes, packages, etc...
- ▶ Commenter c'est le bien.