

Documentation du code

- ◆ Commentaires informels dans le code
 - pas de vision globale
 - obligation de lire tout le code
- ◆ Documentation au traitement de texte
 - fastidieuse à rédiger
 - jamais maintenue à jour

→des mentions spéciales dans le code
pour produire automatiquement la doc.

Commentaires spéciaux dans le code

- ◆ Reconnus par des outils dédiés
- ◆ Génération automatique
de la documentation technique
- ◆ Mis à jour en même temps que le code
 - Demande un peu d'effort
 - Garantit une documentation à jour
sans trop d'investissement

Documentation des sources

- ◆ index général et liens pour naviguer dans l'ensemble des éléments d'un projet java
- + très utile pour avoir une vision complète et détaillée des sources d'un projet
- + utile pour prendre en main et comprendre une API pour les développeurs et chefs de projets
- contraignant : ajouter systématiquement les éléments de documentation dans le code
- + documentation intégrée dans le code, donc facile à mettre à jour en même temps que le code

Élément de documentation

◆ Commentaire javadoc

```
/**  
    texte (html) de la documentation  
    balise(s) javadoc  
*/
```

et non

```
/*  
*/
```

Balises javadoc

@author Philippe
@version 1.2
@since 1.0
@param parametre commentaire
@return commentaire
@exception classeDException commentaire
@see Classe, Classe#champ, Classe#Classe
@see Classe#methode, Classe#methode(typeParam)
@see Interface, Interface#methode

Outillage javadoc

- ◆ En ligne de commande
 - > javadoc **-public** [-private ...]
 -d chemin_doc
 -classpath chemin_source
 package1 package2
- ◆ Dans BlueJ
 - > Menu Tools / Project Documentation
- ◆ Dans Eclipse
 - > Menu Project / Generate javadoc...