

Utilisation d'IntelliJ IDEA pour développer des programmes en JAVA

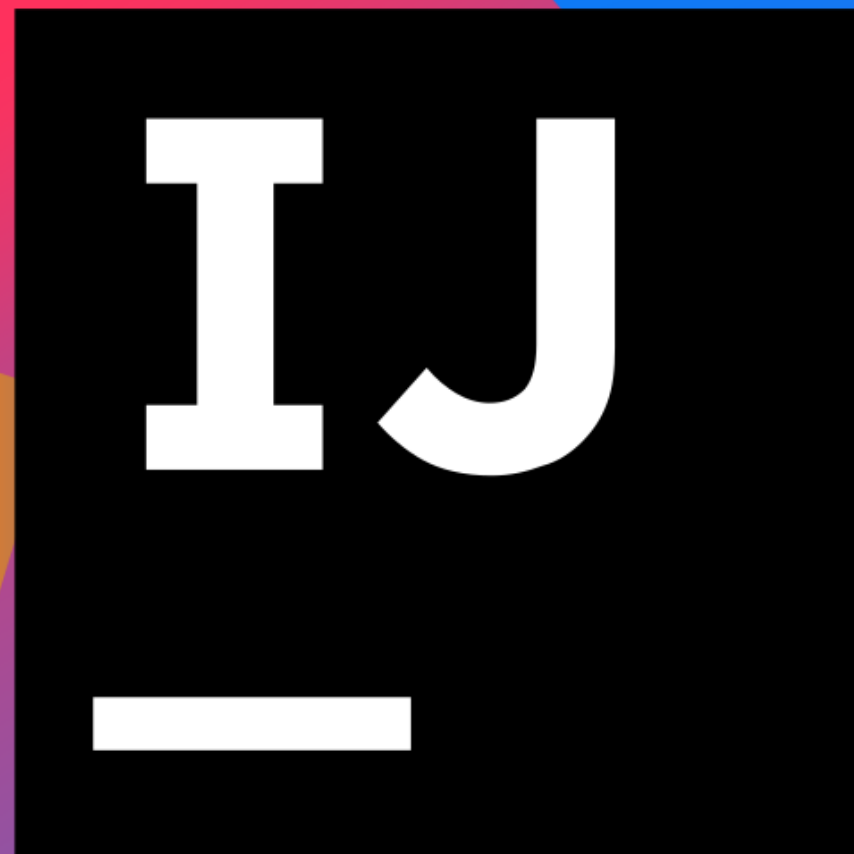


DAMEAN ALEXANRA

Pour faciliter le développement informatique, il existe des logiciels appelés environnement de développement, ou integrated development environment (IDE) en anglais.

Ces logiciels proposent de nombreux outils pour faciliter le développement de logiciels dont principalement :

- de l'auto-complétion syntaxique
- une vérification d'erreur en live
- différents outils de compilation
- des outils de debugging avancés



INTELLIJ IDEA

Il existe plusieurs IDE pour Java dont les plus connus sont Eclipse, NetBeans et IntelliJ. IntelliJ est à mon goût le plus moderne des 3 et il est aujourd'hui très répandu.

IntelliJ IDEA est l'un des IDE les plus populaires et les plus puissants pour le développement Java. IntelliJ IDEA est développé et maintenu par JetBrains.

INTRODUCTION À INTELLIJ IDEA

IntelliJ IDEA IDE est riche en fonctionnalités et permet un développement rapide de logiciels. Cet IDE améliore également la qualité du code.

IntelliJ IDEA IDE est disponible en deux éditions différentes:

- Community Edition: open source (gratuit)
- Ultimate Edition: commercial

Caractéristiques d'IntelliJ IDEA

IntelliJ IDEA est unique parmi tous les autres IDE Java grâce à sa compréhension approfondie du code et du contexte du codeur.

Cet IDE est conçu autour du principe de codage selon lequel les développeurs doivent pouvoir coder des applications avec le moins de distractions possible. Ainsi dans le cas d'IntelliJ IDEA, vous n'avez que l'éditeur visible à l'écran et pour toutes les autres fonctions non liées au codage, vous aurez des raccourcis dédiés.

Les caractéristiques importantes d'IntelliJ IDEA pour développer des programmes en JAVA:

- **Saisie de code intelligente** : IntelliJ IDEA fournit une liste de symboles applicables au code les plus pertinents dans le contexte actuel et prend en charge la complétion de code basée sur le contexte.
- **Achèvement du code de la chaîne** : c'est encore une autre fonctionnalité avancée. Les symboles fournis par les méthodes ou les getters et ceux applicables au contexte actuel sont répertoriés par la fonction de complétion de code de chaîne.



IntelliJ IDEA

- **Complétion du membre statique** : vous pouvez ajouter des méthodes et des constantes statiques et IntelliJ IDEA IDE ajoutera automatiquement les instructions d'importation requises au code afin d'éviter une erreur de compilation.
- **Détection des doublons** : il peut y avoir des fragments de code en double dans le code et cette fonctionnalité fournit des notifications ou des suggestions au programmeur à leur sujet.



- **Environnement axé sur l'éditeur** : lors du codage, vous obtenez également des fenêtres contextuelles rapides qui vous aident à vérifier des informations supplémentaires sans quitter le contexte actuel.
- **Inspections et correctifs rapides** : IntelliJ IDEA fait apparaître une petite ampoule sur la ligne de code où il a détecté que vous êtes sur le point de faire une erreur. Lorsque vous cliquez sur cette ampoule, vous verrez une liste de suggestions de code.

- **Raccourcis pour tout** : comme mentionné précédemment, comme les programmeurs ont moins de distractions, IntelliJ IDEA fournit des touches de raccourci (raccourcis clavier) pour presque tout, y compris des actions comme la sélection rapide et la commutation entre diverses fenêtres d'outils et de nombreuses autres opérations.
- **Débogueur en ligne** : vous pouvez utiliser le débogueur en ligne pour déboguer votre application dans l'EDI lui-même, rendant ainsi les processus de développement et de débogage plus efficaces.



Débogage de l'application

En utilisant le débogage, vous pourrez suivre le code en cours d'exécution, inspecter l'état dans lequel se trouve le code et également inspecter le flux d'exécution. Le débogueur proposé par IntelliJ IDEA vous aide à déboguer tout et n'importe quoi, du code le plus simple aux applications complexes.

```
5  ▶ public static void main(String[] args) {  
6    // write your code here  
7  ● | System.out.println("Hello,World!!");  
8    }  
9
```



Débogage de l'application

En utilisant le débogueur en ligne d'IntelliJ IDEA pour Java, vous pouvez effectuer diverses opérations comme arrêter l'exécution d'un programme à tout moment, entrer et sortir de n'importe quelle fonction, définir divers points d'arrêt, inspecter et surveiller les variables, évaluer une expression dans le programme et ainsi de suite.

Contrôle de version

IntelliJ IDEA offre une intégration avec les outils de contrôle de version les plus courants : Git , Mercurial , Perforce et Subversion .Vous pouvez consulter l'historique de l'ensemble de votre projet ou des fichiers séparés, comparer les versions de fichiers , gérer les branches et même traiter les demandes d'extraction GitHub sans quitter l'EDI.



Terminal

IntelliJ IDEA comprend un terminal intégré pour travailler avec un shell de ligne de commande depuis l'EDI. Par exemple, si vous avez l'habitude d'exécuter des commandes Git à partir de la ligne de commande, vous pouvez les exécuter à partir du terminal au lieu d'appeler ces actions à partir du menu. Les pistes Terminal avec votre shell par défaut du système, mais il prend également en charge un certain nombre d'autres coquilles, comme `cmd.exe`, `bash`, `sh` ainsi de suite.

Utiliser IntelliJ IDEA pour développer des programmes en JAVA est très pratique. IDE dispose de divers outils pour aider les développeurs. Il est très simple de créer des applications Java dans cet IDE.

”

