Developpez.com
Club des développeurs

JetBrains

Gérer et combiner langages et injections de langage dans IntelliJ IDEA

Par Elena Kozlova - Delphine Massenhove (traducteur)

Date de publication : 2 octobre 2020

Pour réagir au contenu de ce tutoriel, un espace de dialogue vous est proposé sur le forum.
Commentez

I - Préambule....3
II - Éditeur dédié au code injecté....4
III - Injecter des expressions régulières....4
IV - Injection de langage automatique....5
V - En résumé….....6
VI - Remerciements Developpez.com....7

Copyright ® 2020 Elena Kozlova. Aucune reproduction, même partielle, ne peut être faite de ce site et de l'ensemble de son contenu : textes, documents, images, etc. sans l'autorisation expresse de l'auteur. Sinon vous encourez selon la loi jusqu'à trois ans de prison et jusqu'à 300 000 € de dommages et intérêts.
https://jetbrains.developpez.com/tutoriel/jetbrains-intellijidea-gerer-combiner-langages-injections/

I - Préambule

Dans presque toutes les applications, il y a des situations dans lesquelles on doit travailler sur de petits éléments de code dans d'autres langages intégrés dans le code du projet. Il peut par exemple s'agir de fragments de code HTML en Java ou d'expressions régulières éparpillés dans votre application. Travailler avec de tels éléments de code est particulièrement facile dans IntelliJ IDEA grâce aux injections de langage. En effet, ces dernières apportent le surlignage et la saisir automatique du code, la détection d'erreurs et d'autres fonctionnalités d'assistance au codage pour les fragments de code intégrés. Voyons plus en détail comment fonctionnent les injections de langage.

Essayons d'intégrer un morceau de code HTML dans Java. Il vous suffit d'appuyer sur **Alt + Entrée**, de sélectionner **Inject language or reference** et de choisir **HTML** dans la liste.

Par défaut, l'EDI injecte un langage temporairement. Donc si vous supprimez l'élément intégré, l'injection sera perdue également. Si vous souhaitez conserver l'injection en permanence, appuyez à nouveau sur **Alt + Entrée** lorsque la fenêtre contextuelle qui vous invite à remplacer l'injection par une annotation s'affiche. L'EDI insère l'annotation **@Language** avec l'ID de langage correspondant. Pour les appels de méthode, l'EDI crée un modèle d'injection de langage qui ressemble à ceci : **// language=HTML**.

Désormais, même si vous supprimez l'injection originale, l'EDI saura déjà quel langage y est prévu et sera prêt à offrir une assistance au codage.

Copyright ® 2020 Elena Kozlova. Aucune reproduction, même partielle, ne peut être faite de ce site et de l'ensemble de son contenu : textes, documents, images, etc. sans l'autorisation expresse de l'auteur. Sinon vous encourez selon la loi jusqu'à trois ans de prison et jusqu'à 300 000 € de dommages et intérêts.
https://jetbrains.developpez.com/tutoriel/jetbrains-intellijidea-gerer-combiner-langages-injections/

II - Éditeur dédié au code injecté

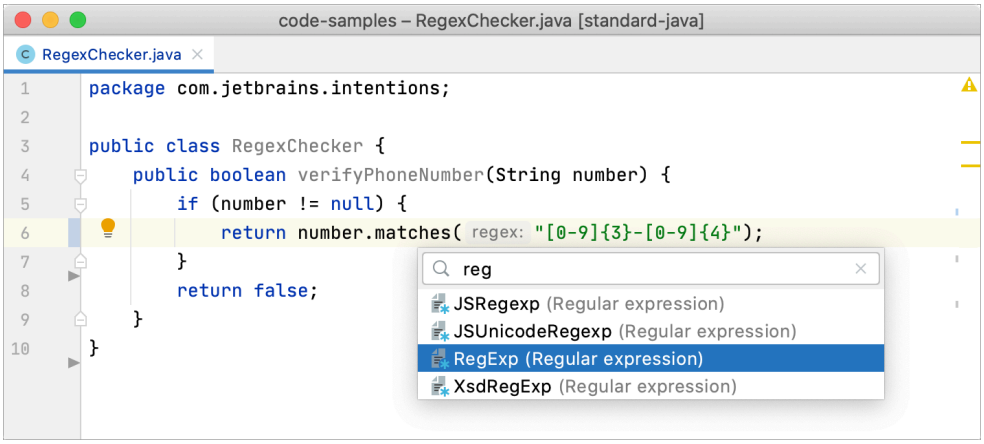
Si vous souhaitez vous concentrer uniquement sur le langage du fragment de code intégré, IntelliJ IDEA propose un éditeur dédié. Cela vous permet de travailler avec le code comme s'il était dans un document séparé. Cette fonctionnalité s'avère très pratique lorsque les morceaux de code intégrés sont volumineux ou difficiles à distinguer du reste du code.

Bien entendu, l'éditeur pour les injections de langage offre les fonctionnalités habituelles d'assistance au codage : saisie automatique du code, intentions, inspections, etc. Pour ouvrir l'éditeur, appuyez sur **Alt + Entrée** et sélectionnez **Edit Fragment**.

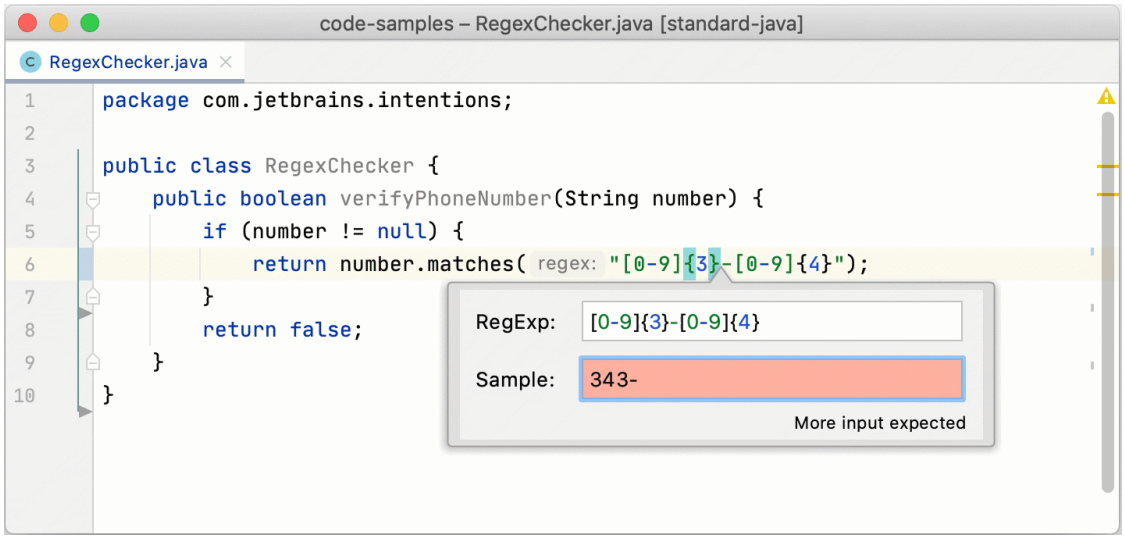
III - Injecter des expressions régulières

Les expressions régulières constituent un outil puissant, qui a de nombreuses applications, de la recherche à la validation des entrées, jusqu'aux refactorisations. Cependant, il nous arrive à tous d'avoir du mal à les composer correctement. Si vous utilisez des expressions régulières dans votre code, IntelliJ IDEA a un petit bonus à vous proposer. Tout d'abord, veillez à intégrer vos expressions régulières en utilisant des injections de langage.

Copyright ® 2020 Elena Kozlova. Aucune reproduction, même partielle, ne peut être faite de ce site et de l'ensemble de son contenu : textes, documents, images, etc. sans l'autorisation expresse de l'auteur. Sinon vous encourez selon la loi jusqu'à trois ans de prison et jusqu'à 300 000 € de dommages et intérêts.
https://jetbrains.developpez.com/tutoriel/jetbrains-intellijidea-gerer-combiner-langages-injections/



Lorsque des expressions régulières sont injectées, vous pouvez vérifier si elles sont compilées correctement. Appuyez sur **Alt + Entrée** et sélectionnez **Check RegExp**. Vous pouvez désormais saisir une séquence de caractères pour vous assurer que votre expression correspond à la combinaison voulue. Et voilà !

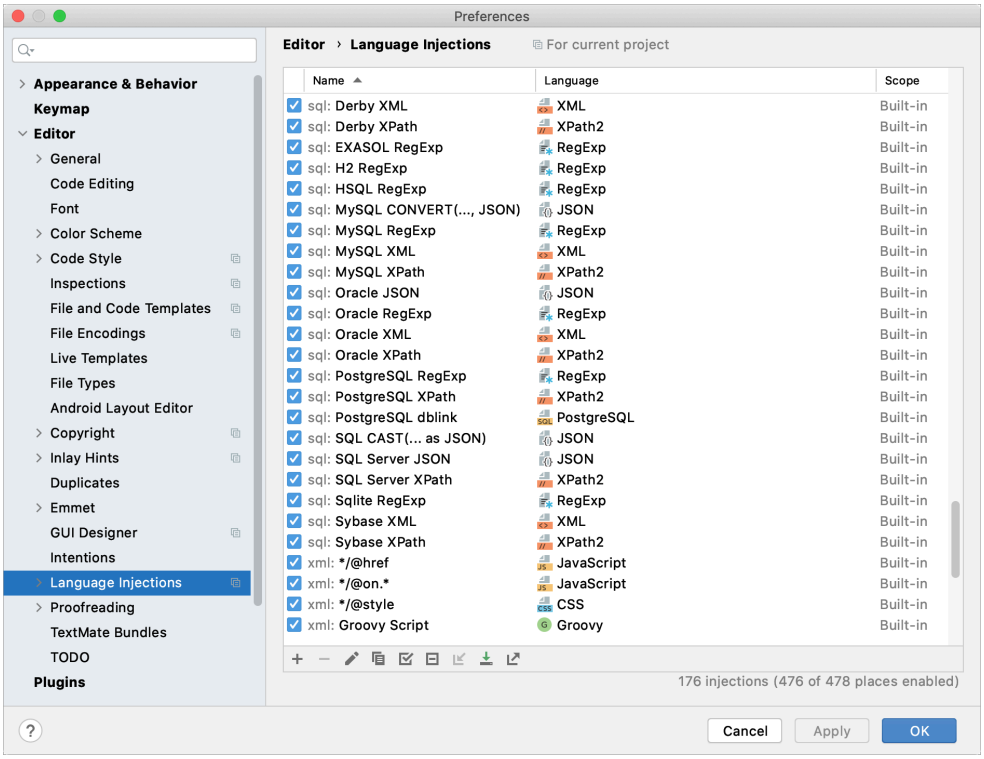


IV - Injection de langage automatique

Dans IntelliJ IDEA, il existe un ensemble de règles prédéfinies selon lesquelles l'EDI injecte automatiquement des langages à certains endroits dans votre code. Ces règles se trouvent dans **Settings/Preferences > Editor > Language Injections**.

Vous pouvez ainsi modifier les règles d'injection existantes, les dupliquer et désactiver celles dont vous n'avez pas besoin.

Copyright ® 2020 Elena Kozlova. Aucune reproduction, même partielle, ne peut être faite de ce site et de l'ensemble de son contenu : textes, documents, images, etc. sans l'autorisation expresse de l'auteur. Sinon vous encourez selon la loi jusqu'à trois ans de prison et jusqu'à 300 000 € de dommages et intérêts.
https://jetbrains.developpez.com/tutoriel/jetbrains-intellijidea-gerer-combiner-langages-injections/



Si aucune des règles existantes ne vous convient, vous pouvez créer les vôtres et les partager ensuite avec les membres de votre équipe.

Pour partager une règle personnalisée via un VCS, déplacez-la vers **Project** en utilisant l'option correspondante de la barre d'outils dans **Settings/Preferences > Editor > Language Injections**. Vous allez ainsi créer un nouveau fichier dans le dossier **.idea** nommé **IntelliLang.xml** avec vos règles personnalisées que vous pouvez placer sous contrôle de version.

Si vous n'utilisez pas de VCS, vous pouvez partager vos règles en les exportant dans des fichiers XML, puis en les important dans un autre projet.

V - En résumé…

Les injections de langage sont idéales pour gérer plusieurs langages dans un même projet. Elles fournissent la saisie automatique, l'analyser, la mise en évidence du code et même un éditeur dédié pour les morceaux de code intégrés. Et elles vous permettent d'injecter automatiquement les langages requis à des emplacements spécifiques dans votre code, selon des règles prédéfinies ou selon vos propres règles personnalisées.

Pour plus d'informations sur les injections de langage, consultez **la documentation** d'IntelliJ IDEA. Cette fonctionnalité est alimentée par le plugin IntelliLang, qui est fourni et activé par défaut, et fonctionne immédiatement.

Avez-vous déjà utilisé des injections de langage dans IntelliJ IDEA ? Qu'en pensez-vous ? Faites-le-nous savoir dans les commentaires.

En attendant, nous progressons avec IntelliJ IDEA 2020.2. **Rejoignez le programme d'accès anticipé** et testez en avant-première ses nouvelles fonctionnalités et améliorations.

Bon développement !

Copyright ® 2020 Elena Kozlova. Aucune reproduction, même partielle, ne peut être faite de ce site et de l'ensemble de son contenu : textes, documents, images, etc. sans l'autorisation expresse de l'auteur. Sinon vous encourez selon la loi jusqu'à trois ans de prison et jusqu'à 300 000 € de dommages et intérêts.
https://jetbrains.developpez.com/tutoriel/jetbrains-intellijidea-gerer-combiner-langages-injections/

VI - Remerciements Developpez.com

Nous tenons à remercier **Malick** pour la mise au gabarit et **Claude Leloup** pour la relecture orthographique.

Copyright ® 2020 Elena Kozlova. Aucune reproduction, même partielle, ne peut être faite de ce site et de l'ensemble de son contenu : textes, documents, images, etc. sans l'autorisation expresse de l'auteur. Sinon vous encourez selon la loi jusqu'à trois ans de prison et jusqu'à 300 000 € de dommages et intérêts.
https://jetbrains.developpez.com/tutoriel/jetbrains-intellijidea-gerer-combiner-langages-injections/