

Guide Pratique EDI NetBeans

Copyright © 2005 Sun Microsystems, Inc. All rights reserved.

Édition et Refactoring du Code

Table des Matières

Ouvrir l'Éditeur de Source.....	2
Gestion Automatique des Caractères Fermant.....	3
Terminer une Instruction.....	3
Répartir un String Sur Deux Lignes.....	3
Afficher les Numéros de Lignes.....	4
Génération d'Extraits de Code.....	4
Compléter les Instructions Java.....	4
Désactiver l'Apparition Automatique de la Fenêtre Flottante de Completion de Code.....	6
Modifier le Raccourci Clavier pour la Completion de Code.....	7
Proposer des Mots Déjà Présents dans le Fichier.....	7
Génération de Méthode à Override Depuis les Classes Étendues.....	7
Génération de Méthodes Depuis des Interfaces Implémentées.....	8
Générer les Propriétés Read/Write.....	8
Étendre les Abbreviations.....	10
Ajouter, Modifier et Supprimer des Abréviations.....	11
Création et Utilisations de Macros.....	12
Création et Personnalisation des Modèles de Fichiers.....	12
Gestion des Imports.....	14
Afficher la Documentation Javadoc Tout en Éditant.....	14
Formattage du Code.....	15
Indenter des Blocs de Code Manuellement.....	15
Modifier les Règles de Formattage.....	15
Mondifier les Polices et Couleurs.....	16
Raccourcis Sélection Texte.....	17
Naviguer dans le Fichier Java.....	18
Visualiser et Naviguer Entre les Membres d'une Classe.....	18
Déplacer le Point d'insertion et Déplacer le texte dans la Fenêtre.....	19
Mettre un Signet à une Ligne de Code.....	20
Cacher les Parties de Code.....	20
Naviguer Depuis l'Éditeur de Source.....	21
Se Déplacer Parmi les Fichiers Ouverts.....	21
Se Rendre au Code Concerné et la Documentation.....	21
Jumping Between Areas Where You Have Been Working.....	21
Se positionner sur le Noeud d'un Fichier Depuis l'Éditeur de Source.....	22
Rechercher et Remplacer.....	22
Trouver les Occurences du nom de la Classe, Méthode ou Champs sélectionné.....	22
Renommer Toutes les Occurrences du nom de la Classe, Méthode ou Champ Actuellement Sélectionné.....	23
Rechercher et Remplacer des Combinaisons de Caractères dans un Fichier.....	24
Autres Recherches de Fichiers.....	25
Déplacer une Classe dans un Différent Paquetage.....	26

Changer la Signature d'une Méthode.....	28
Commandes Refactoring.....	28
Consolider Notes à Vous-Même.....	29
Ajouter, Enlever et Modifier des Tags pour la Liste des Tâches.....	30
Afficher les Tâches Par Priorité.....	30
Filtrer les Entrées de la Liste de Tâches.....	30
Comparer les Différences Entre Deux Fichiers.....	31
Scinder l'Éditeur de Source.....	31
Maximiser l'Espace Pour l'Éditeur de Source.....	32
Modifier les Raccourcis Clavier de l'Éditeur de Source.....	32

L'EDI NetBeans fournit pas mal d'outils pour supporter le développement d'application Java [NdT: Heureusement, c'est tout de même le but de cet environnement. Sinon, à quoi servirait-il ? ;-)], mais c'est dans l'Éditeur de Source que vous y passerez la plus grande partie de votre temps. Cela étant dit, une énorme attention fut portée dans les fonctionnalités et petites touches subtiles pour simplifier et accélérer le coding, le rendant plus agréable.

Les fonctions de génération de code et de navigation sont disponibles par les raccourcis clavier, vous évitant ainsi de devoir rarement vous écarter du clavier. Les fonctionnalités de Refactoring vous permet de modifier la structure de votre code et de voir ces modifications propagées dans votre projet. De plus, il y a tout un tas de simples raccourcis pour la navigation de fichiers qui, pris ensemble, rend le codage plus agréable.

Du point de vue architecture, l'Éditeur de Source est en fait une collection de différents types d'éditeurs, chacun ayant des fonctionnalités spécifiques pour certains types de fichier. Par exemple, lorsque vous ouvrez un fichier Java, il y aura une coloration syntaxique spécifique aux fichiers Java, ainsi que le completion de code, refactoring et d'autres fonctionnalités spécifiques aux fichiers Java. Pareillement, lorsque vous ouvrez des fichiers JSP, HTML, XML, .properties, descripteurs de déploiement, et d'autres types, vous obtiendrez des fonctionnalités spécifiques à ces fichiers.

Peut-être encore plus important, l'Éditeur de Source est grandement intégré avec les autres parties de l'EDI, ce qui simplifie grandement votre *workflow*. Par exemple, vous pouvez spécifier des points d'arrêt directement dans l'Éditeur de Source et tracer le code lors de son exécution. Lorsque des erreurs de compilation sont reportées dans la fenêtre Output, vous pouvez positionner la source à l'endroit de ces erreurs en double-cliquant sur l'erreur ou en pressant F12.

Ouvrir l'Éditeur de Source

Avant que vous ne comenciez à travailler avec l'Éditeur de Source, vous devriez avoir configuré un projet dans l'EDI. Vous pouvez ensuite ouvrir un fichier existant ou créer un nouveau fichier depuis les modèles. Voyez le chapitre Fondamentales Du Projet EDI pour les informations de base sur la création des projets et fichiers ainsi qu'une description des nombreux modèles de fichiers.

Si vous désirez simplement créer un fichier sans devoir créer auparavant un projet pour cela, vous pouvez utiliser la fenêtre Favorites. La fenêtre Favorites vous permet de créer des répertoires et fichiers sur votre disque dur depuis votre EDI. La fenêtre Favorites n'est pas conçue pour le développement de projet à grande échelle, mais il peut être utile si vous désirez rapidement ouvrir et éditer quelques fichiers.

Pour utiliser l'Éditeur de Source sans devoir créer un projet:

1. Choisissez Window | Favorites pour ouvrir la fenêtre Favorites.

2. Ajouter le répertoire où vous désirez que votre fichier réside (ou il y réside déjà) en cliquant-droit sur le noeud Favorites et en choisissant Add To Favorites pour finalement choisir le répertoire depuis l'explorateur de fichier.
3. Dans la fenêtre Favorites, naviguez jusqu'au fichier que vous désirez éditer et double-cliquez dessus pour l'ouvrir dans l'Éditeur de Source.

Si vous désirez créer un nouveau fichier, cliquez-droit sur le noeud d'un dossier et choisissez New | Empty File et donnez un nom de fichier (y compris l'extension).

Gestion Automatique des Caractères Fermant

Lorsque vous êtes en train d'éditer dans l'Éditeur de Source, l'une des premières choses que vous remarquerez rapidement est que les caractères fermants sont automatiquement rajoutés lorsque vous tapez le caractère ouvrant. Par exemple, si vous tapez des guillemets, les guillemets fermants seront insérés à la fin de la ligne. Pareillement, les parenthèses (), crochets, ([]), et accolades ({}) seront également complétés pour vous.

Bien que vous pourriez trouver cela ennuyant de premier abord, la fonctionnalité fut conçue pour ne pas vous perturber. Ainsi, si vous tapez vous même le caractère fermant, le caractère qui fut inséré automatiquement sera supprimé. Vous pouvez également terminer une ligne en tapant un point-virgule (;) pour finir une instruction. Le point-virgule est inséré à la fin de la ligne, après les caractères générés automatiquement.

Voyez les points suivants pour des informations plus détaillées sur la façon d'utiliser l'insertion des caractères fermant correspondants.

Terminer une Instruction

Lorsque l'Éditeur de Source insère les caractères correspondants à la fin de la ligne, on pourrait penser que cela vous force à vous placez manuellement à la fin de la ligne avant que vous ne puissiez taper le point-virgule. En fait, vous pouvez juste taper le point-virgule sans déplacer le curseur et il se placera à la fin de la ligne automatiquement.

Par exemple, pour obtenir la ligne:

```
System.out.println("Test");
```

vous ne devez taper que ceci:

```
System.out.println("Test;
```

Répartir un String Sur Deux Lignes

Si vous avez une longue String que vous désirez répartir sur deux lignes, l'Éditeur de Source rajoutera la syntaxe de concaténation de String lorsque vous presserez la touche d'Entrée.

Par exemple, pour obtenir les lignes:

```
String s = "Though typing can seem tedious, reading long" +  
"and convoluted sentences can be even worse."
```

vous ne devez taper que ceci

```
String s = "Though typing can seem tedious, reading long  
and convoluted sentences can be even worse."
```

Les trois guillemets supplémentaires et le signe + sont rajoutés pour vous.

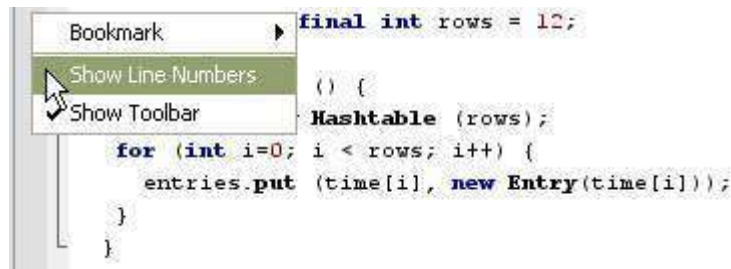
Si vous désirez passer à la ligne sans créer de concaténation, pressez Maj-Entrée.

Conseil EDI NetBeans

Lorsque vous introduisez un littéral, le 'point-virgule intelligent' ne fonctionne pas. You need to manually move the insertion point to the end of the line to type the semicolon. One way to get around this might be to create an abbreviation or macro for the code that contains the string to generate the semicolon and to place the insertion point between the quote marks.

Afficher les Numéros de Lignes

Par défaut, les numéros de ligne se sont pas affichés dans l'Éditeur de Source pour gagner de l'espace et réduire le trouble visuel. Si vous avez besoin des numéros de ligne, vous pouvez les rendre visible en cliquant-droit dans la marge de gauche de l'Éditeur de Source et choisir Show Line Numbers.



Génération d'Extraits de Code

L'Éditeur de Source a de nombreuses fonctionnalités pour réduire la frappe pour taper du code. En plus de réduire la frappe, ces fonctionnalités peuvent vous faire éviter les fautes de frappe et vous aider à trouver les bons noms de classe et de méthode. Cette partie se concentre sur les fonctionnalités pour les fichiers Java, mais la plupart de ces fonctionnalités (comme la completion de code et le replis de code) sont également disponibles pour d'autres types de fichiers..

Compléter les Instructions Java

Lorsque vous tapez des identifiants Java dans l'Éditeur de Source, vous pouvez utiliser la fenêtre flottante de completion de code de l'EDI pour vous aider à terminer vos expressions. De plus, une fenêtre flottante avec la documentation Javadoc apparaît et affiche la documentation pour l'option sélectionnée dans la fenêtre flottante de completion de code.



Pour ouvrir la fenêtre flottante de completion de code, faites l'une des opérations suivantes:

- Tapez les premiers caractères d'une expression et pressez Ctrl-Espace (ou Ctrl- \backslash).
- Marquez une pause après avoir tapé un point (.) dans une expression.
- Tapez le mot-clef `new` ou `import`, suivi d'un espace, et marquez une pause.

La fenêtre flottante de completion de code va apparaître avec une proposition de choix possibles selon ce que vous avez introduit jusque là.

Pour réduire les propositions, continuez à taper l'expression.

Pour compléter l'expression et fermer la fenêtre flottante de completion de code, faites l'une des choses suivantes:

- Continuer à taper jusqu'à ce qu'il n'y ait plus qu'une seule option et pressez la touche Entrée.
- Positionnez-vous dans la liste à l'aide des touches fléchées ou de la souris et pressez ensuite Entrée.

Pour compléter l'expression et laisser la fenêtre de completion de code ouverte, sélectionnez une proposition et pressez Maj-Entrée. Cela est très utile lorsque vous enchaînez les méthodes. Par exemple, si vous devez introduire:

```
getRootPane().setDefaultButton(defaultButton)
```

vous pouvez faire ce qui suit:

1. Tapez `getRo` (ce qui ne vous proposera que `getRootPane()` dans la fenêtre flottante de completion de code) et pressez Maj-Entrée.
2. Tapez `.setDef` (ce qui devrait sélectionner `setDefaultButton(defaultButton)` comme méthode dans la fenêtre flottante de completion de code) et pressez Entrée.

Conseil EDI NetBeans

Si la méthode que vous avez choisi dans la fenêtre de completion de code a un paramètre (comme dans l'exemple ci-dessus), un paramètre de remplacement est inséré. Vous pouvez écraser cet paramètre avec le nom de paramètre que vous désirez utiliser.

Si la méthode accepte plusieurs paramètres, un nom de paramètre est proposé pour le premier paramètre. Lorsque vous le remplacez par votre nom de paramètre et tapez ensuite une virgule, un paramètre de remplacement est alors proposé pour la valeur suivante.

Une fois que vous avez fini d'introduire vos paramètres, tapez un point-virgule (;) pour terminer la ligne. Le point-virgule est alors automatiquement inséré après la parenthèse fermante.

Pour compléter le texte qui est commun à toutes les propositions, pressez la touche de Tabulation. Par exemple, si vous travaillez sur une classe `Swing` et que vous vous commencez à taper `addW`, il y aura trois méthodes de listener qui vous seront proposées, commençant toute par `addWindow`. (`addWindowFocusListener`, `addWindowListener`, et `addWindowStateListener`). En pressant la touche de Tabulation, vous allez compléter avec le code commun (pour obtenir `addWindow`). Vous pouvez ensuite taper la lettre suivante (F pour `addWindowFocusListener(1)`, L pour `addWindowListener(1)`, ou S pour `addWindowStateListener(1)`) et pressez Entrée pour compléter l'expression.

Pour fermer la fenêtre flottante de completion de code sans choisir aucune des propositions, pressez la touche d'Échappement.

Désactiver l'Apparition Automatique de la Fenêtre Flottante de Completion de Code

Si vous êtes de ceux qui trouvent la fenêtre de completion de code comme étant plus une nuisance que d'une quelconque utilité, vous pouvez désactiver automatiquement l'apparition de cette fenêtre flottante. La completion de code fonctionnera toujours si vous forcez son apparition en pressant les touches `Ctrl-Espace` ou `Ctrl-\`.

Vous pouvez également laisser active l'apparition automatique de la fenêtre de completion de code, mais désactiver l'encombrante fenêtre Javadoc. La fenêtre flottante Javadoc peut toujours être invoquée manuellement en pressant `Maj-Ctrl-Espace`.

Pour désactiver l'apparition automatique de la fenêtre de completion de code:

1. Choisissez `Tools | Options`, étendez le noeud `Editor | Editor Settings settings`, et sélectionnez le noeud `Java Editor`.
2. Descendez jusqu'à la partie `Expert` de la feuille de propriétés et décochez la propriété `Auto Popup Completion Window`.

Pour désactiver l'apparition automatique de la fenêtre flottante Javadoc lorsque vous utilisez la fenêtre de completion de code:


1. Choisissez `Tools | Options`, étendez le noeud `Editor | Editor Settings settings`, et sélectionnez le noeud `Java Editor`.
2. Décochez la propriété `Auto Popup Javadoc Window`.

Conseil EDI NetBeans

Vous pouvez également seulement modifier le temps nécessaire avant que la fenêtre de completion de code n'apparaisse. Par défaut, le délai est de 500 millisecondes. Vous pouvez changer cette valeur dans la propriété `Delay of Completion Window Auto Popup`.

Modifier le Raccourci Clavier pour la Completion de Code

Si vous préférez utiliser différents raccourcis pour la completion de code, vous pouvez modifier ces raccourcis dans l'EDI NetBeans.

1. Choisissez Tools | Options et sélectionnez le noeud Editor | Editor Settings settings.
2. Cliquez sur le bouton  de la propriété Global Key Bindings pour ouvrir la boîte de dialogue Global Key Bindings.
3. Sélectionnez une commande que vous désirez modifier, cliquez sur Add et tapez le raccourcis que vous désirez utiliser.

Les commandes appropriées sont Show Code Completion, Hide Code Completion, et Show the Javadoc Window.

Proposer des Mots Déjà Présents dans le Fichier

Si vous tapez un mot qui apparait déjà ailleurs dans votre fichier, vous pouvez utiliser un raccourcis clavier pour compléter le mot selon le premier mot trouvé dans l'Éditeur de Source qui corresponde aux caractères que vous avez tapé. Cette fonctionnalité fonctionne pour n'importe quel texte dans le fichier.

Pour rechercher en arrière par rapport au curseur, pressez Ctrl-K.

Pour recherchez en avant par rapport au curseur, pressez Ctrl-L.

Par exemple, si vous avez défini la méthode `refreshCustomerInfo` en ligne 100 et que vous désirez appeler cette méthode à la ligne 50, vous pouvez taper `ref` et pressez Ctrl-L. En supposant qu'il n'y ait pas d'autres mots qui commencent par `ref` entre les lignes 50 et 100, le restant du mot `refreshCustomerInfo` sera rajouté.

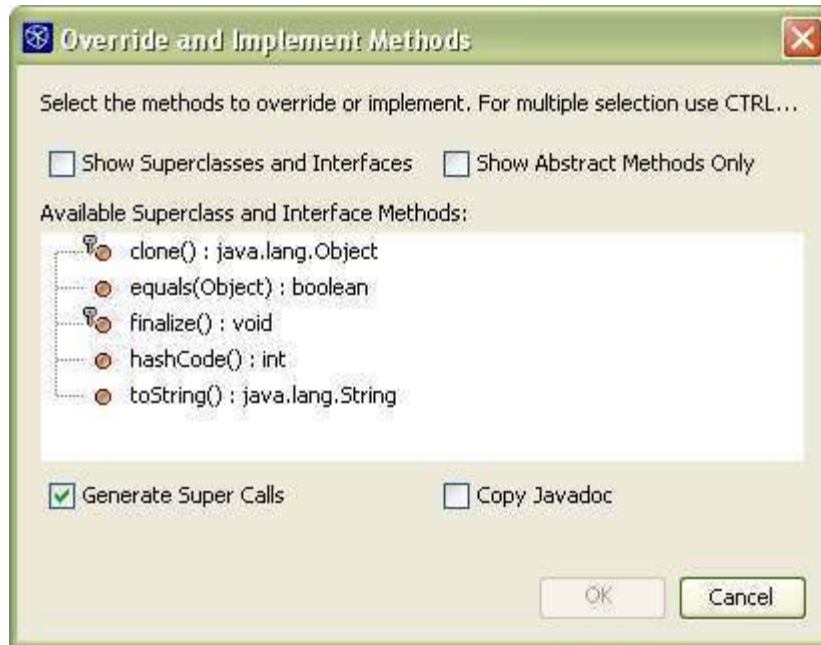
Génération de Méthode à Override Depuis les Classes Étendues

Lorsque vous étendez une classe, vous avez la possibilité de override les méthodes de la classe étendue. Pour vous simplifier la tâche, vous pouvez générer les méthodes à overrider dans votre nouvelle classe et ensuite modifier les méthodes dans votre classe.

Pour avoir les méthodes que vous désirez overridden insérées dans votre code:

1. Choisissez Tools | Override Methods ou pressez Ctrl-I.
2. Dans la boîte de dialogue Override Methods, sélectionnez les méthodes que vous désirez overridden. Vous pouvez pressez la touche Ctrl pour sélectionner plusieurs méthodes.

Vous pouvez décocher la case Generate Super Calls si vous ne désirez pas générer un appel de la méthode dans la classe super.



Génération de Méthodes Depuis des Interfaces Implémentées

Lorsque vous créez une classe qui implémente une interface, vous pouvez utiliser la fonctionnalité Synchronize de l'EDI pour générer les méthodes de cette interface dans la classe.

Pour générer les méthodes dans une classe qui implémente une interface:

1. Dans votre déclaration de classe, tapez `implements NomDUneInterface`.
2. Choisissez Tools | Synchronize.

La boîte de dialogue Synchronize apparaît et vous permet de spécifier pour laquelle des interfaces vous désirez générer les méthodes.


Conseil EDI NetBeans

Vous pouvez également paramétrer l'EDI pour qu'il vous propose de lui-même de générer les méthodes d'une interface après que vous ayez déclaré qu'une classe implémente une interface. Choisissez Tools | Options. Étendez le noeud Editing | Java Sources et sélectionnez le noeud Source Synchronization. Dans le panneau de propriétés de la fenêtre, cocher la case Synchronization Enabled.

Générer les Propriétés Read/Write

Vous pouvez créer les propriétés en respectant le model de composant JavaBeans à partir de rien, ou vous pouvez utiliser la fonctionnalité Encapsulate Fields pour créer un getter et un setter pour un champ déjà existant.

Pour créer une propriété read-write à partir de rien:

1. Dans la fenêtre Project, étendez le noeud du fichier en ensuite le noeud de la classe ().
2. Cliquez-droit sur le noeud Bean Patterns et choisissez Add | Property.
3. Dans la boîte de dialogue New Property Pattern, introduisez un nom et un type pour la propriété. Vérifier ensuite la boîte de dialogue Field.

Après avoir cliqué sur OK, un champs avec ses méthodes `get` et `set` sont générés.

Pour créer les méthodes d'accesseurs pour un champ existant:

1. Dans l'Éditeur de Source, cliquez-droit sur le champs et choisissez Refactor | Encapsulate Fields.
2. Dans la boîte de dialogue Encapsulate Fields, vérifiez que le champs que vous avez sélectionné est bien sélectionné comme devant être encapsulé. Indiquez ensuite la visibilité pour ce champ et pour les méthodes d'accesseurs (le getter et le setter).
3. Cliquez sur Next pour prévisualiser les modifications dans la fenêtre Refactoring.
4. Dans la fenêtre Refactoring, vérifiez les modifications proposées et cliquez sur Do Refactoring.



Conseil EDI NetBeans

La fonctionnalité Encapsulate Fields est l'une des nombreuses commandes de "refactoring" dans l'EDI. Les commandes de refactoring vous fournissent la possibilité de faire des changements de noms ou de structure et de les refléter dans le reste de votre projet. Voyez le sujet Commandes de Refactoring pour un résumé des commandes de Refactoring.

Étendre les Abbreviations

Pour les mots-clés et identifiants fréquemment utilisés, vous pouvez tirer avantages des abréviations dans l'Éditeur de Source, pour réduire le nombre de frappes. Les abréviations sont étendues lorsque vous tapez l'abréviation et que vous pressez ensuite la barre d'espacement.


Voyez le Tableau 1 pour une liste des abréviations pour les fichiers Java.

Si une abréviation est la même qu'un texte que vous désirez taper (et donc que vous ne désirez pas qu'il soit transformé en autre chose), pressez Maj-Espace pour éviter sa transformation.

Il y a également des abréviations définies pour les fichiers JSP ainsi que pour les fichiers XML.

Vous pouvez modifier des abréviations existantes (aussi bien l'abréviation que son extension), créer de nouvelles abréviations, et en supprimer, par l'intermédiaire de la boîte de dialogue Abréviations pour chaque type de fichier.

Pour modifier la liste des abréviations pour un type de fichier:

1. Choisissez Tools | Options.
2. Étendez Editing | Editor Settings et sélectionnez le type d'éditeur dont vous désirez modifier les abréviations.
3. Cliquez sur le bouton  dans la propriété Abréviations pour ouvrir la boîte de dialogue Abréviations.
4. Utilisez les boutons Add, Edit, et Remove pour modifier la liste des abréviations selon vos goûts.

Conseil EDI NetBeans

Dans l'extension de vos abréviations, vous aimeriez que le point d'insertion reste quelque part au milieu du texte. Par exemple, le point d'insertion apparaît entre les guillemets lorsque `sout` est transformé en `System.out.println("");` comme cela vous pouvez taper la chaîne de caractères sans avoir à déplacer le curseur.


Utilisez le caractère ligne verticale (|) dans le champ Expansion de l'abréviation pour indiquer où le point d'insertion devrait apparaître lorsque l'abréviation est étendue.

Tableau 1: Abréviations Java dans l'Éditeur de Source

Abréviation	Transformée en
En	Enumeration
Ex	Exception
Ob	Object
Psf	public static final
Psfb	public static final boolean
Psfi	public static final int
Psfs	public static final String
St	String
ab	abstract
bo	boolean
br	break
ca	catch (
cl	class
cn	continue
df	default:

Abbréviation	Transformée en
eq	equals
ex	extends
fa	false
fi	final
fl	float
fy	finally
ie	interface
im	implements
iof	instanceof
ir	import
le	length
pe	protected
pr	private
psf	private static final
psfb	private static final boolean
psfi	private static final int
psfs	private static final String
pst	printStackTrace();
pu	public
re	return
serr	System.err.println("");
sout	System.out.println("");
st	static
sw	switch {
sy	synchronized
tds	Thread.dumpStack();
th	throws
tw	throw
twn	throw new
wh	while (



Ajouter, Modifier et Supprimer des Abréviations

1. Choisissez Tools | Options, étendez le noeud Editing | Editor Settings, et sélectionnez le noeud pour le type d'éditeur pour lequel vous désirez modifier les abréviations.
2. Cliquez sur le bouton  dans la propriété Abbreviations pour ouvrir l'éditeur de propriétés Abbreviations.
3. Utilisez les boutons Add, Edit, et Remove pour modifier la liste des abréviations.

Création et Utilisations de Macros

Vous pouvez enregistrer des macros dans l'EDI pour réduire ce qui devrait être une longue série de frappe au clavier en un raccourcis clavier. Dans les macros, vous pouvez combiner la frappe de caractères dans l'Éditeur de Source et l'utilisation d'autres raccourcis clavier.

Pour enregistrer une macro:

1. Mettez le point d'insertion dans la partie d'un fichier dans l'Éditeur de Source, où vous désirez enregistrer la macro.
2. Cliquez sur l'icône  de la barre d'outils de l'Éditeur de Source (ou pressez Ctrl-J et tapez ensuite `s`) pour démarrer l'enregistrement.
3. Enregistrer la macro en utilisant n'importe quelle séquence de touche, que ce soit pour la frappe de caractères ou l'utilisation de raccourcis clavier. Les déplacements de souris ainsi que les clicks (comme la sélection d'un menu) ne sont pas enregistrés.
4. Cliquez sur l'icône  dans la barre d'outils de l'Éditeur de Source (ou pressez Ctrl-J et ensuite `e`) pour terminer l'enregistrement.
5. Dans la boîte de dialogue Recorded Macro qui apparaît, cliquez sur Add pour assigner un raccourcis clavier à la macro.
6. Dans la boîte de dialogue Add Keybinding, pressez les touches que vous désirez utiliser pour le raccourci clavier. (Par exemple, si vous désirez le raccourci Maj-Alt-Z, pressez les touches Alt, Maj, et Z.) Si vous pressez une mauvaise touche, cliquez sur le bouton Clear pour recommencer.

Soyez prudent de ne pas réutiliser un raccourci qui est déjà assigné. Si le raccourci que vous introduisez est un raccourci de l'éditeur, un message d'avertissement apparaîtra dans la boîte de dialogue. Mais si c'est un raccourci qui s'applique en dehors de l'Éditeur de Source, vous n'aurez pas d'avertissement.

Toutes les macros que vous enregistrez ne fonctionnent que pour le type de fichier dans lequel la macro est créée.

Conseil EDI NetBeans

Après avoir enregistré une macro, vous pouvez l'éditer manuellement dans la boîte de dialogue Recorded Macro.

Les raccourcis clavier que vous avez inclus sont indiqués par leur nom d'action. Vous pouvez obtenir la liste de ces noms d'actions dans la boîte de dialogue Key Bindings pour un type d'éditeur particulier, que vous utilisez lorsque vous assignez ou éditez des raccourcis clavier.

Création et Personnalisation des Modèles de Fichiers

Vous pouvez personnaliser les modèles que vous utilisez pour créer vos fichiers, ou même créer vos propres modèles. Cela peut être très utile si vous désirez ajouter des éléments standard dans tous vos fichiers (comme les notes de copyright) ou que vous désirez modifier la façon dont d'autres éléments sont générés.

Vous pouvez également créer vos propres modèles et les rendre disponibles dans l'assistant New File.

Il y a quelques macros qui sont à notre disposition dans les modèles pour générer dynamiquement le texte dans les fichiers créés. Ces macros sont identifiables par le deux underscore (__) qui apparaissent avant et après le nom de la macro. Voyez le Tableau 2 pour une liste des macros disponibles.

Pour éditer un modèle:

1. Choisissez Tools | Options et étendez le noeud Source Creation and Management | Templates node.
2. Étendez le noeud catégorie pour le modèle que vous désirez éditer.
3. Cliquez-droit sur le noeud du modèle et choisissez Open.
4. Éditez le modèle pour ensuite l'enregistrer.

Conseil EDI NetBeans

Tous les modèles qui sont repris sous le noeud Source Creation and Management | Templates ne représentent pas des objets qui peuvent être modifiés au niveau de l'utilisateur. Dans certains cas, les modèles sont disponibles dans l'assistant New File mais ne représentent pas une construction de fichier (comme ceux de la catégories Enterprise and Sun Resources, par exemple). D'un autre côté, des modèles qui ne servent pas à la création de fichiers, sont également repris (comme ceux des catégories Filesystems et Privileged).


Pour créer un nouveau modèle:

1. Choisissez File | New File et sélectionnez le modèle qui vous servira de base pour votre nouveau modèle.
2. Éditez le fichier, en y intégrant les macros de modèles que vous désirez utiliser (voir le tableau 2) et enregistrez-le.

Si le modèle est pour une classe Java, vous pouvez utiliser le nom de fichier pour le nom de la classe et le nom du constructeur. Ils seront automatiquement ajustés dans les fichiers que vous créez à partir du modèle.

3. Dans la fenêtre Projects, cliquez-droit sur le noeud du fichier et choisissez Save As Template dans le menu contextuel.
4. Dans la boîte de dialogue Save As Template, sélectionnez la catégorie à utiliser pour l'assistant New File.

Tableau 2: Macros pour les Modèles de Fichier

Macro	Information Substituée
__USER__	Votre nom d'utilisateur. Si vous désirez modifier la valeur de __USER__, Choisissez Tools Options et sélectionnez le noeud Editing Java Sources. Ensuite, cliquez sur le bouton  dans la propriété Strings Table et modifier la valeur de USER.
__DATE__	La date à laquelle le nouveau fichier a été créé.
__TIME__	L'heure à laquelle le nouveau fichier a été créé.
__NAME__	Le nom de la classe (sans l'extension de fichier). Il vaut mieux ne pas utiliser cette macro pour les noms de la classe et du constructeur dans le fichier (utilisez à la place le nom du fichier).
__PACKAGE__	Le nom du paquetage où la classe est créée.
__PACKAGE_SLASHES__	Le nom du paquetage de la classe avec des barres obliques (/) au lieu de points (.).
__PACKAGE_AND_NAME__	Le nom pleinement qualifié du fichier (par exemple com.mondomaine.monpaquetage.MaClasse).
__PACKAGE_AND_NAME_SLASHES__	Le nom pleinement qualifié du fichier avec une barre oblique (/) au lieu de points (.).

Macro	Information Substituée
<code>__QUOTES__</code>	Un guillemet ("). Utilisez cette macro si vous désirez que le texte substitué apparaisse entre guillemets dans le fichier généré. Si vous placez une macro entre guillemets dans le modèle, le texte n'est pas considéré comme étant une macro et ne sera donc pas substitué dans le fichier créé.

Gestion des Imports

L'EDI vous fournit une paire de commandes que vous pouvez utiliser pour gérer vos instructions d'import dans votre code.

- Fix Imports (Maj-Alt-F), qui insère automatiquement toutes les instructions d'import manquante pour l'entiereté du fichier. Les instructions d'import sont générées par classes (plutôt que par paquetage). Pour une gestion rapide de vos imports, utilisez cette commande.
- Fast Import (Maj-Alt-I), qui vous permet d'ajouter une instruction d'import ou de générer le nom de classe pleinement qualifié pour l'identifiant sélectionné. Cette commande est utile si vous désirez générer un instruction d'import pour tout un paquetage ou si vous désirez utiliser une classe pleinement qualifiée en-ligne plutôt qu'une instruction d'import.

Afficher la Documentation Javadoc Tout en Éditant

L'EDI vous donne différentes façons d'accéder à la documentation pour les classes du JDK et des bibliothèques.

Pour accéder à la documentation de la classe actuellement sélectionnée dans l'Éditeur de Source, pressez Maj-Ctrl-Espace. Une fenêtre flottante apparaîtra avec la documentation de la classe. Cette fenêtre flottante apparaît également lorsque vous utilisez la completion de code. Vous pouvez refermer la fenêtre flottante en cliquant en dehors de la fenêtre flottante.

Pour ouvrir la documentation dans un navigateur web pour la classe sélectionnée, cliquez-droit sur la classe et choisissez Show Javadoc (ou pressez Alt-F1).

Pour ouvrir la page d'index de la documentation d'une bibliothèque dans un navigateur web, choisissez View | Documentation Indices et choisissez l'index dans le sous-menu.

Conseil EDI NetBeans

La documentation de certaines bibliothèques est fournie avec l'EDI. Cependant, vous devrez enregistrer la documentation des autres bibliothèques dans l'EDI pour que la fonctionnalité de Javadoc fonctionne. Voyez le sujet Référencer la Documentation du JDK (Javadoc) Depuis le Projet du chapitre Principes Fondamentaux du Projet de l'EDI.

Paradoxalement, la documentation du JDK est disponible dans la fenêtre flottante de l'Éditeur de Source, mais pas depuis un navigateur. Cela s'explique par le fait que la fenêtre flottante de Javadoc dans l'Éditeur de Source va rechercher la documentation dans les sources qui sont incluses avec le JDK. Cependant, pour visualiser la documentation dans

le navigateur, il vous faut la documentation Javadoc au format HTML, que vous devez télécharger en plus du JDK.

Formattage du Code

Lorsque vous introduisez ou avez du code généré dans l'Editeur de Source, votre code Java est automatiquement formatté de façon suivante, par défaut:

- Les membres de classes sont indentées de 4 espaces.
- Les instructions continues sont indentées de 8 espaces.
- Toutes tabulation est convertie Any tabs that you enter are converted to spaces.
- Lorsque vous êtes dans un bloc de commentaire (commençant avec `/**`), une astérisque est automatiquement rajouté à la nouvelle ligne lorsque vous pressez le touche d'Entrée.
- L'accolade ouvrante est mise sur la même ligne que la déclaration de la classe ou de la méthode..
- Aucun espace n'est mis avant une parenthèse ouvrante.

Si votre fichier perd le formattage correct (ce qui est probable après une opération de copier/coller), vous pouvez reformater l'entiereté du fichier en pressant Maj-Ctrl-F. Si vous avez sélectionné des lignes, seules ces lignes seront reformatées.

Indenter des Blocs de Code Manuellement

Vous pouvez sélectionner plusieurs lignes de code et ensuite indenter ces lignes en pressant la touche de Tabulation, ou Ctrl-T.

Vous pouvez inverser l'indentation en sélectionnat certaines lignes et en pressant Maj-Tabulation ou Ctrl-D.

Modifier les Règles de Formattage

Vous pouvez ajuster la façon dont vos fichiers sont formattés en utilisant les ainsi nommés moteurs d'indentation, qui contiennent les paramètres pour le nombre d'espaces, le placement des accolades, etc. Par défaut, tous les fichiers Java utilisent le moteur d'indentation Java. Vous pouvez modifier ce moteur d'indentation, indiquer que les fichiers Java utilisent un autre moteur d'indentation, ou en créer un entièrement nouveau.

Pour ajuster les règles de formattage pour les fichiers Java:

1. Choisissez Tools | Options.
2. Étendez le noeud Editing | Indentation Engines et sélectionnez le noeud Java Indentation Engine.
3. Ajustez les propriétés du moteur d'indentation selon vos goûts.
4. Reformatter chaque fichier aux nouvelles règles en ouvrant le fichier et en pressant Maj-Ctrl-F (sans sélectionner du texte).

Il y a d'autres moteurs d'indentation prédéfinis à votre disposition, aussi bien génériques (comme les moteurs d'indentation "simple" et "line wrapping") et spécifiques aux types de fichier (notamment pour les fichiers JSP, HTML, et XML).

Pour changer le moteur d'indentation qui est utilisé pour un fichier Java:

1. Choisissez Tools | Options.
2. Étendez le noeud Editing | Editor Settings et sélectionnez le noeud Java Editor.
3. Sélectionnez le moteur d'indentation depuis la liste déroulante de la propriété Indentation Engine.
4. Reformatter chaque fichier selon les nouvelles règles en ouvrant le fichier et en pressant Maj-Ctrl-F (sans sélectionner du texte).

Pour créer un nouveau moteur d'indentation:

1. Choisissez Tools | Options.
2. Étendez le noeud Editing | Indentation Engines, cliquez-droit sur le noeud du moteur d'indentation qui vous servira de base, et choisissez Copy.
3. Cliquez-droit sur le noeud Indentation Engines et choisissez Paste | Copy.
4. Modifier le nom du moteur d'indentation et ajuster les propriétés selon vos goûts.
5. Dans la boîte de dialogue Options, étendez le noeud Editing | Editor Settings et sélectionnez le noeud pour l'éditeur (par exemple Java Editor ou HTML Editor) auquel vous désirez appliquer le moteur d'indentation.

Conseil EDI NetBeans

Toutes modifications que vous faites dans les règles de formatage ne s'applique pas au code des blocs protégés qui sont générés lorsque vous concevez des applications AWT ou Swing par l'intermédiaire de l'Éditeur Visuel.

Modifier les Polices et Couleurs


Vous pouvez modifier les polices qui sont utilisées dans l'Éditeur de Source ainsi que les couleurs et mise en évidence utilisées pour les éléments syntaxiques de votre code.


Les polices et couleurs sont déterminées par le type de fichier que vous éditez. Tout ajustement que vous faites sont donc par type de fichier.

Pour modifier la taille de la police pour un type d'éditeur:

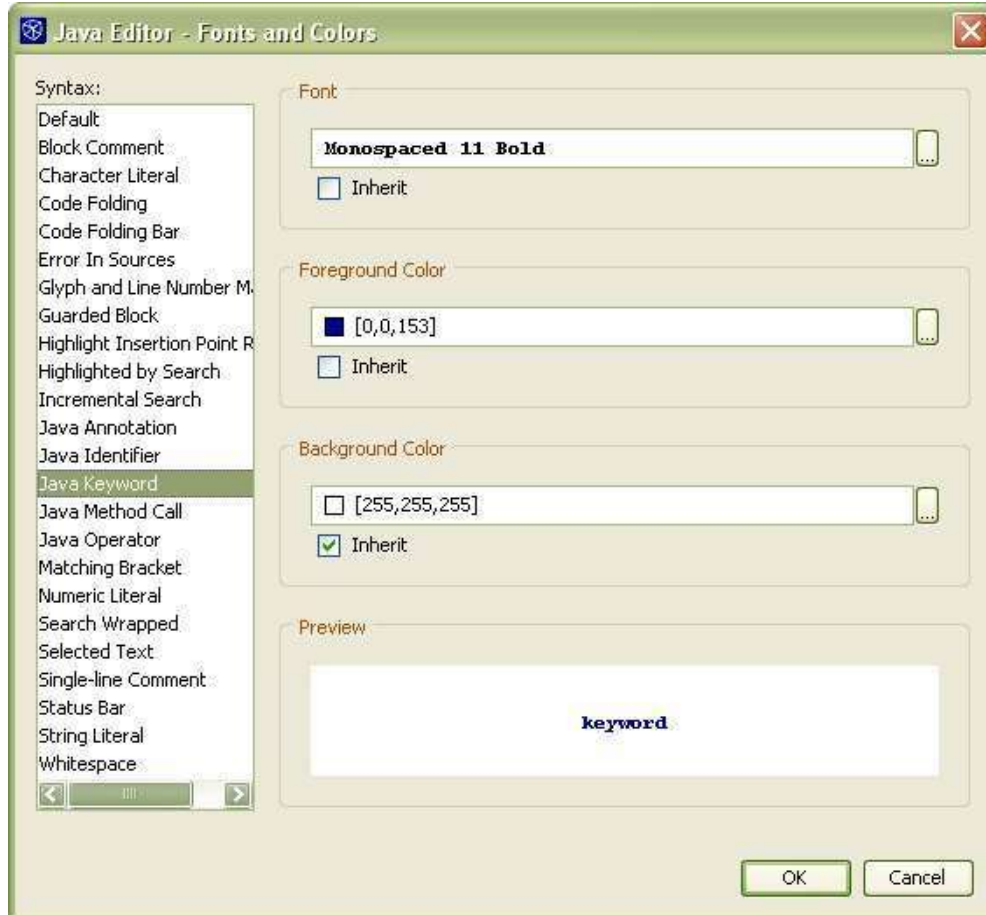
1. Choisissez Tools | Options.
2. Étendez Editing | Editor Settings et sélectionnez le noeud pour le type d'éditeur (c-à-d. Java Editor ou HTML Editor).
3. Modifier la propriété Font Size.

Pour modifier la palette de couleurs et de mise en évidence pour un type de fichier:

1. Choisissez Tools | Options.
2. Étendez Editing | Editor Settings et sélectionnez le noeud pour le type d'éditeur (c-à-d. Java Editor ou HTML Editor).
3. Cliquez sur le bouton  dans la propriété Fonts and Colors.
4. Dans la boîte de dialogue Fonts and Colors, sélectionnez un élément de la colonne Syntax.
5. Modifier la palette de couleur pour cet élément de syntaxe dans les champs Font, Foreground Color, et Background Color. Vous pouvez introduire les

valeurs directement ou cliquez sur le bouton  situé à côté de chaque champ pour faire apparaître une palette de couleur ou sélectionner une police.

Si vous avez coché la case Inherit pour l'un des trois paramètres, cet élément utilisera le paramètre de l'un des autres éléments de syntaxe, habituellement Default.



Raccourcis Sélection Texte

Pour vous permettre de garder les mains sur le clavier, il y a tout un tas de raccourcis qui vous permettent de sélectionner du texte, ne plus le sélectionner, et modifier le texte sélectionné. Voyez le tableau 3 pour une sélection de ces raccourcis.

Tableau 3: Raccourcis Sélection Texte

Description	Raccourcis
Sélectionne l'identifiant ou tout autre mot où se situe le curseur d'insertion.	Alt-J
Sélectionne tout le texte situé entre les parenthèses, crochets, ou accolades. Le curseur d'insertion doit se situer juste après la parenthèse/crochet/accolade ouvrant ou fermant.	Maj-Ctrl-[
Sélectionne le prochain (précédent) caractère ou étend la sélection d'un caractère.	Maj-Droit (Maj-Gauche)

Description	Raccourcis
Sélectionne le prochain (précédent) mot ou étend la sélection d'un mot.	Maj-Ctrl-Droit (Maj-Ctrl-Gauche)
Crée ou étend la sélection de texte à la ligne du dessous (du haut).	Maj-Bas (Maj-Haut)
Crée ou étend la sélection de texte jusqu'à la fin (début) de la ligne.	Maj-Fin (Maj-Début)
Crée ou étend la sélection de texte jusqu'à la fin (début) du document.	Maj-Ctrl-Fin (Maj-Ctrl-Début)
Crée ou étend la sélection de texte d'une page vers le bas (vers le haut).	Maj-PageBas (Maj-PageHaut)

Naviguer dans le Fichier Java

L'EDI fournit plusieurs mécanismes pour simplifier la visualisation et navigation dans un fichier Java:

- La fenêtre Navigator, qui apparaît en dessous de la fenêtre Projects, fournit une liste de membres (c-à-d. constructeurs, champs et méthodes) du fichier Java en cours d'édition.
- Signets, qui vous permettent de vous positionner facilement à des places bien spécifiques dans le fichier.
- Les raccourcis Alt-K et Alt-L “jump list” décrit dans le point Jumping Between Areas Where You Have Been Working ci-dessous.
- Raccourcis clavier pour faire dérouler la fenêtre. Voir le tableau 4.
- La fonctionnalité de pliage de code, qui vous permet de replier des parties de code (comme le corps des méthodes, les commentaires Javadoc, et les blocs d'instructions d'import) permettant de rendre visible une plus grande partie de la classe dans la fenêtre à un moment donné.

Visualiser et Naviguer Entre les Membres d'une Classe

La fenêtre Navigator de l'EDI fournit une liste de tous les membres (constructeurs, méthodes, et champs) de votre classe. Vous pouvez double-cliquez sur un membre dans cette liste pour que son code source apparaisse dans l'Éditeur de Source.



Vous pouvez utiliser les boutons de filtre au bas de la fenêtre pour cacher les membres non public, les membres statiques, les champs et/ou les membres hérités.

Déplacer le Point d'insertion et Déplacer le texte dans la Fenêtre

Il existe un large éventail de raccourcis que vous pouvez utiliser pour déplacer le point d'insertion ou déplacer le texte dans l'Éditeur de Source sans déplacer le point d'insertion. Voyez le tableau 4 pour une liste des raccourcis les plus utiles pour naviguer dans le fichier.

Tableau 4: Raccourcis Déplacement Curseur et Déplacement Texte

Description	Raccourcis
Déplacer le point d'insertion au mot suivant (mot précédent).	Ctrl-Droit (Ctrl-Gauche)
Déplacer le point d'insertion au début (à la fin) du fichier.	Ctrl-Début (Ctrl-Fin)
Déplacer le texte vers le haut (le bas) sans déplacer le point d'insertion.	Ctrl-Haut (Ctrl-Bas)
Déplace le texte dans la fenêtre pour que la ligne où se situe le curseur soit située au début de la fenêtre.	Alt-U, ensuite T
Déplace le texte dans la fenêtre pour que la ligne où se situe le curseur soit située au milieu de la fenêtre.	Alt-U, ensuite M
Déplace le texte dans la fenêtre pour que la ligne où se situe le curseur soit située au bas de la fenêtre.	Alt-U, ensuite B
Déplace le point d'insertion à la parenthèse/crochet/accolade correspondante à celle située juste avant le point d'insertion.	Ctrl-[

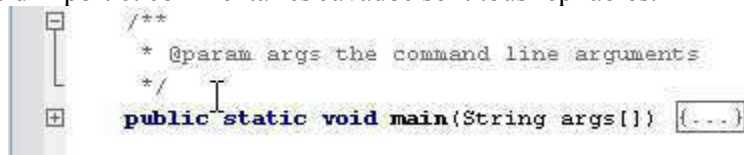
Mettre un Signet à une Ligne de Code




Vous pouvez rajouter des signets aux fichiers pour que vous puissiez retrouver l'emplacement facilement. Vous pourrez alors ensuite vous déplacer de signet en signet en pressant la touche F2.

Pour mettre un signet sur une ligne de fichier, cliquez sur la ligne et pressez Ctrl-F2. Pour enlever le signet, pressez également Ctrl-F2.

Cacher les Parties de Code

Vous pouvez “replier” des parties du fichier pour que seul une ligne de ce bloc ne soit visible dans l'Éditeur de Source, laissant plus d'espace pour voir d'autres lignes. Les méthodes, classes internes, blocs d'import et commentaires Javadoc sont tous repliables.



Les blocs repliables sont signalés par l'icône  dans la marge de gauche au niveau de la première ligne du bloc. Le restant du bloc est signalé par un ligne verticale qui s'étend depuis l'icône . Les blocs repliés sont signalés par l'icône . Vous pouvez cliquer sur l'une de ces icônes pour replier ou déplier le bloc qu'il représente.


Vous pouvez également replier et déplier plusieurs blocs de code grâce aux raccourcis clavier et aux lignes de menu dans le menu Edit | Code Folds et le sous-menu Code Folds dans l'Éditeur de Source. Voyez le tableau 5 pour une liste de ces commandes et raccourcis.

Tableau 5: Commande Pliage/Dépliage de Code

Commande	Raccourci
Replie le bloc	Ctrl-Moins-PavéNumérique
Déplie le bloc	Ctrl-Plus-PavéNumérique
Replie Tout	Maj-Ctrl-Moins-PavéNumérique
Déplie Tout	Maj-Ctrl-Plus-PavéNumérique
Replie Toutes les Javadoc	[aucun]
Déplie Toutes les Javadoc	[aucun]
Replie Tout le Code Java [replie tout sauf la documentation Javadoc]	[aucun]
Déplie Tout le Code Java [déplie tout sauf la documentation Javadoc]	[aucun]

Par défaut, aucun code n'est replié. Vous pouvez configurer l'Éditeur de Source pour replier du code par défaut lorsque vous créez un fichier ou que vous ouvrez un fichier qui ne l'était pas au préalable.

Pour configurer l'EDI pour replier automatiquement certaines parties de code:

1. Choisissez Tools | Options et sélectionnez le noeud Editing | Editor Settings | Java Editor.
2. Cliquez sur le bouton  dans la propriété Code Folding.
3. Cochez la case pour chaque type d'éléments dont vous désirez qu'ils soient repliés par défaut (les méthodes, classes internes, blocs d'import, commentaire Javadoc, commentaires d'en-tête).

Naviguer Depuis l'Éditeur de Source

L'EDI inclut des raccourcis pratiques pour naviguer parmi les fichiers, les différents bout de code et les différentes fenêtres. Au plus vous pouvez intégrer ces raccourcis dans votre workflow, au moins votre main devra quitter le clavier pour prendre la souris.

Se Déplacer Parmi les Fichiers Ouverts

Vous pouvez vous déplacer parmi les fichiers ouverts en utilisant les raccourcis clavier Alt-Gauche et Alt-Droit.

Se Rendre au Code Concerné et la Documentation

Les raccourcis du tableau 6 vous permet de vous rendre aux endroits du fichier actuel ou d'autres fichiers relatifs à l'identifiant sélectionné. Les 5 premiers de ces raccourcis sont disponibles depuis le menu Edit et le sous-menu Go To du menu contextuel de l'Éditeur de Source (clic-droit). La commande Show Javadoc est disponible directement depuis le menu contextuel de l'Éditeur de Source.

Tableau 6: Raccourcis Navigation Java

Commande	Raccourci	Description
Go to Source	Alt-O	Se rend dans le code source de la classe, méthode, ou champ sélectionné, si le source est disponible.
Go to Declaration	Alt-G	Se rend à la déclaration de la classe, méthode ou champ sélectionné.
Go to Super Implementation	Ctrl-B	Se rend à l'implémentation dans la classe héritée de la méthode sélectionnée (si la méthode sélectionnée overrides une méthode d'une autre classe ou si une implémentation de méthode est définie dans une interface).
Go to Line	Ctrl-G	Se rend à un numéro de ligne spécifique dans le fichier actuel..
Go to Class	Maj-Alt-O	Vous permet d'introduire le nom d'une classe et de vous rendre dans le code source de cette classe, s'il est disponible pour l'EDI.
Show Javadoc	Alt-F1	Affiche la documentation de la classe sélectionnée dans le navigateur web. Pour que cette commande puisse travailler, la Javadoc de la classe doit être rendue disponible à l'EDI via le Java Platform Manager (pour la documentation du JDK) ou la Library Manager (pour la documentation des autres bibliothèques).

Jumping Between Areas Where You Have Been Working

Lorsque vous travaillez sur plusieurs fichiers à la fois, ou dans différents endroits du même fichier, vous pouvez utiliser les raccourcis "jump list" pour naviguer directement entre les palces où vous êtes occupés à travailler au lieu de sélectionner une autre fenêtre et de déplacer l'ascenseur. La "jump list" est principalement un historique des lignes sur lesquelles vous avez travaillé dans l'Éditeur de Source.

Vous pouvez vous déplacer en avant ou en arrière entre les emplacements du même fichier avec les raccourcis Alt-K (arrière) et Alt-L (avant). Utilisez Maj-Alt-K et Maj-Alt-L pour naviguer à travers la jump list, mais pour tous les fichiers cette fois-ci.

Se positionner sur le Noeud d'un Fichier Depuis l'Éditeur de Source

Lorsque vous êtes en train de coder dans l'Éditeur de Source, vous pouvez vous positionner sur le noeud qui représente le fichier sur lequel vous travaillez dans d'autres fenêtres. Cela peut être utile, par exemple, si vous devez rapidement vous rendre vers un autre fichier appartenant au même paquetage, ou si vous désirez voir les informations de versioning pour le fichier en cours.

Voyez le tableau 7 pour la liste de ces raccourcis.

Tableau 7: Raccourcis pour Se positionner sur le noeud d'un fichier

Commande	Raccourci
Sélectionnez le noeud pour le fichier actuel dans la fenêtre Projects.	Maj-Ctrl-1
Sélectionnez le noeud pour le fichier actuel dans la fenêtre Files.	Maj-Ctrl-2
Sélectionnez le noeud pour le fichier actuel dans la fenêtre Favorites.	Maj-Ctrl-3

Rechercher et Remplacer

Il y a plusieurs types de recherches dans l'EDI pour différents besoins. Vous pouvez:

- Trouver et Remplacer les noms des classes, méthodes, et champs dans votre projet en utilisant la commande Find Usages.
- Trouver et Remplacer des combinaisons de caractères spécifiques dans un fichier ouvert en pressant Ctrl-F dans le fichier.
- Trouver des fichiers qui correspondent aux critères de recherche basés sur le nom des fichiers, leur contenu, type de fichiers, leur date, et/ou le statut de contrôle de version en cliquant-droit sur le dossier ou le noeud d'un projet dans la fenêtre Projects et choisissant Find (ou en pressant Ctrl-F).

Trouver les Occurrences du nom de la Classe, Méthode ou Champs sélectionné

Lorsque vous travaillez dans l'Éditeur de Source, vous pouvez rapidement savoir où un identifiant Java donné est utilisé dans votre projet en utilisant la commande Find Usages. La commande Find Usages améliore grandement la commande Find en étant sensible au contexte du langage Java.

La commande Find Usage affichera les lignes de votre projet qui contiennent:

- une déclaration de la classe, interface, méthode ou champ.
- (pour les classes et interfaces) une déclaration d'une méthode de la classe ou interface.
- (pour les classes et interfaces) l'utilisation du type, comme à la création d'une nouvelle instance, l'import d'une classe, l'extension d'une classe, l'implémentation d'une interface, cast d'un type ou le lancement d'une exception.
- (pour les classes et interfaces) un usage des méthodes ou champs du type.
- (pour les champs) getting ou setting de la valeur d'un champ.
- (pour les méthodes) l'appel de la méthode.
- (pour les méthodes) l'overriding de la méthode.
- Commentaires qui font références à l'identifiant.

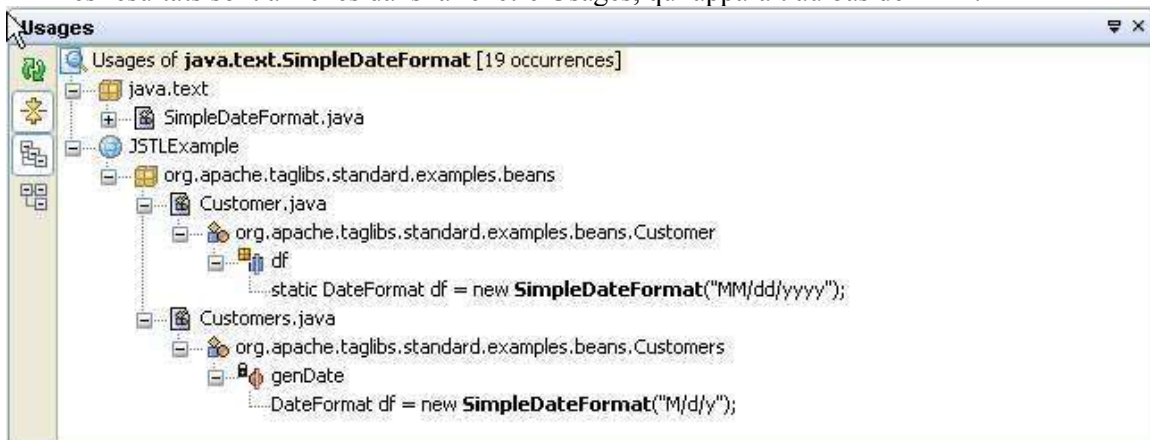
La commande Find Usage ne trouvera pas:

- Des Parties de mots.
- Des mots qui diffèrent du point de vue de la casse.

Pour trouver des occurrences d'un identifiant spécifique dans votre code:

1. dans l'Éditeur de Source, placer le point d'insertion sur le nom de la classe, méthode ou champ dont vous désirez retrouver les occurrences.
2. Cliquez-droit et choisissez Find Usages ou pressez Alt-F7.
3. Dans la boîte de dialogue Find Usages, sélectionnez les types d'occurrences que vous désirez afficher et cliquez sur Next.

Les résultats sont affichés dans la fenêtre Usages, qui apparaît au bas de l'EDI.



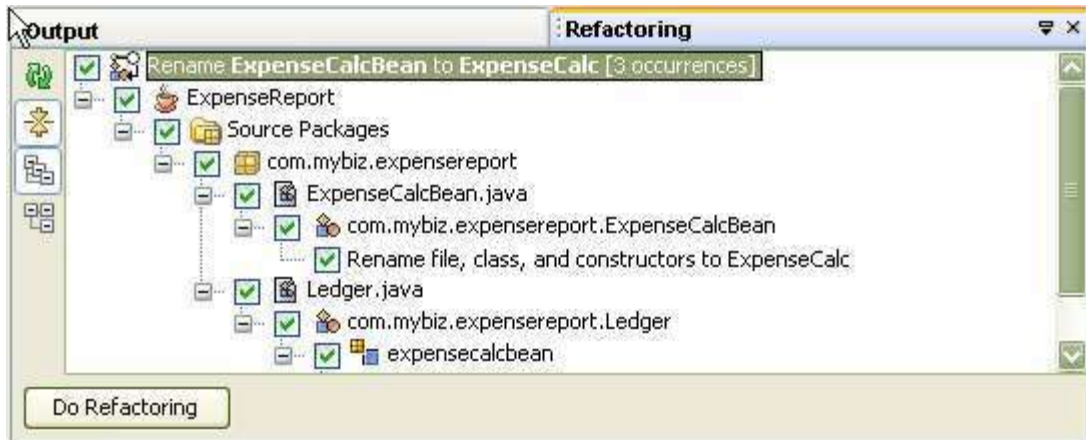
Vous pouvez naviguer jusqu'à une occurrence donnée d'une classe, méthode ou nom de fichier en double-cliquant sur la ligne de l'occurrence dans la fenêtre Usages.

Renommer Toutes les Occurrences du nom de la Classe, Méthode ou Champ Actuellement Sélectionné

Si vous désirez renommer une classe, méthode ou champ, vous pouvez utiliser la commande Refactor | Rename pour renommer toutes les occurrences de l'identifiant.

Pour renommer le nom d'une classe; méthode ou champ:

1. Dans l'Éditeur de Source, placer le point d'insertion sur une occurrence du nom de la classe, méthode et champ que vous désirez renommer.
2. Cliquez-droit sur cette occurrence et choisissez Refactor | Rename ou pressez Maj-Alt-R.
3. Si vous désirez que les occurrences du nom soient également modifiées dans les commentaires, cochez la case Apply Name on Comments dans la boîte de dialogue Rename.
4. Dans la boîte de dialogue Rename, cliquez sur Next.
5. Dans la fenêtre Refactoring, qui apparaît au bas de l'EDI, vérifiez les occurrences indiquées comme allant être modifiées. S'il y a une occurrence dont vous ne désirez pas sa modification, décochez la case de cette ligne.
6. Cliquez sur Do Refactoring pour effectuer les changements proposés.



Conseil EDI NetBeans

Vous pouvez initier le renommage d'une classe ou interface en les renommant en-ligne dans la fenêtre Projects. Après avoir renommé le noeud, la boîte de dialogue Rename vous invitera à renommer les autres occurrences. Vous pouvez cliquer sur Cancel si vous ne désirez pas que les autres occurrences soient renommées.

Rechercher et Remplacer des Combinaisons de Caractères dans un Fichier

Si vous désirez rechercher une combinaison de caractères dans votre fichier, cliquez dans le fichier dans lequel vous désirez effectuer une recherche et pressez Ctrl-F. Tapez le texte que vous désirez rechercher dans la boîte de dialogue Find.

Vous pouvez utiliser une expression régulière comme critère de recherche.

Contrairement à la commande Find Usages, vous pouvez utiliser la commande Find pour rechercher des parties de mots, faire des recherches insensibles à la casse, et surligner les résultats de la recherche dans le fichier.

Une fois terminé avec la boîte de dialogue Find, vous pouvez passer d'une occurrence à l'autre de la recherche en pressant F3 (prochaine occurrence) et Maj-F3 (occurrence précédente).

Pour rechercher après d'autres occurrences du texte actuellement sélectionné, ou du mot où se situe le point d'insertion, pressez Ctrl-F3.

Pour rechercher et remplacer du texte, cliquez dans le fichier où vous désirez remplacer du texte, pressez Ctrl-H, et remplissez les champs Find What et Replace With.



Conseil EDI NetBeans

Par défaut, le surlignement du résultat de la recherche reste activé même après avoir fermé la boîte de dialogue Find. Si cela vous gêne, vous pouvez toujours le désactiver en pressant les touches Maj-Alt-H.

Autres Recherches de Fichiers

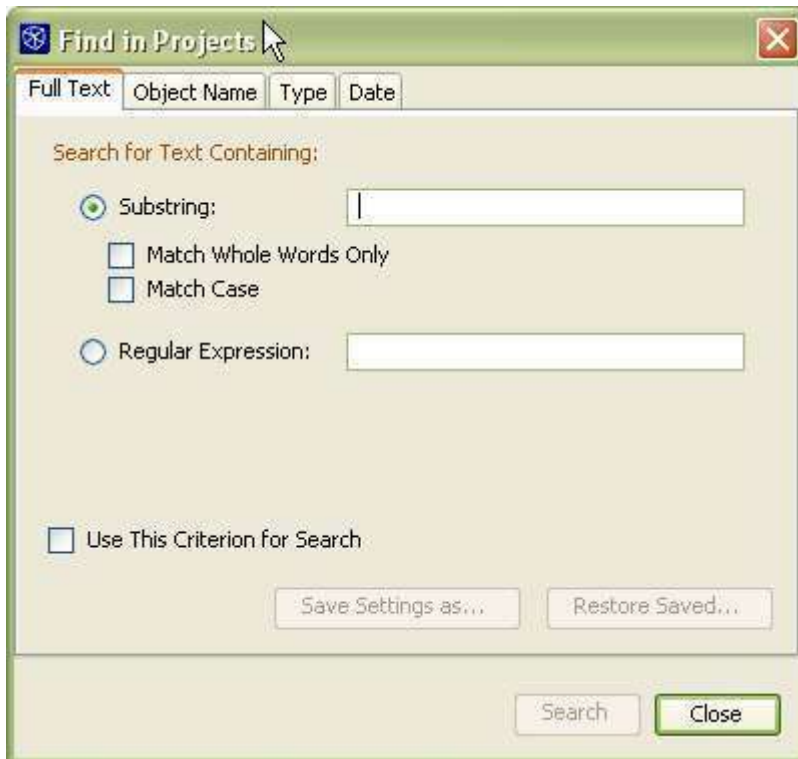
Si vous désirez faire une recherche sur de multiples fichiers pour autre chose que des occurrences d'un identifiant Java spécifique, vous pouvez utiliser la commande Find in Projects. La commande Find in Projects vous permet de rechercher des fichiers appartenant à un répertoire ou un projet à partir d'une combinaison des critères suivants:

- Correspondance à une chaîne de caractères ou expression régulière dans le texte du fichier.
- Correspondance à une chaîne de caractères ou expression régulière dans le nom du fichier.
- Date à laquelle les fichiers ont été modifiés.
- Type de Fichier.
- Statut dans le Version Control Engine (pour les fichiers qui sont gardés par un contrôleur de version et dans un répertoire enregistré dans le Gestionnaire de Versionning de l'EDI).

Pour initier une telle recherche sur les fichiers, faites l'une des possibilités suivantes:

- Dans la fenêtre Projects, cliquez-droit sur le noeud du répertoire ou du projet dans lequel vous désirez effectuer la recherche et choisissez Find (ou pressez Ctrl-F).
- Choisissez Edit | Find in Projects pour rechercher après des fichiers dans tous les projets ouverts.

Après avoir introduit les critères et cliqué sur Search, les résultats sont affichés dans la fenêtre Search Results avec un noeud pour chaque fichier répondant au critère de recherche. Pour les recherches en plein-texte, vous pouvez étendre les noeuds pour voir les lignes qui répondent à votre critère de recherche. Vous pouvez double-cliquer sur une ligne de résultat pour ouvrir le fichier dans l'Éditeur de Source (et dans le cas d'une recherche plein-texte, se positionner sur la ligne correspondante).



Conseil EDI NetBeans

La boîte de dialogue liée au raccourci clavier Ctrl-F dépendra de quelle partie de l'EDI a le focus. Si c'est l'Éditeur de Source qui possède le focus, la boîte de dialogue Find pour un fichier individuel apparaîtra. Si vous avez un noeud sélectionné dans la fenêtre Project, la boîte de dialogue pour une recherche dans de multiples fichiers apparaîtra.

Déplacer une Classe dans un Différent Paquetage

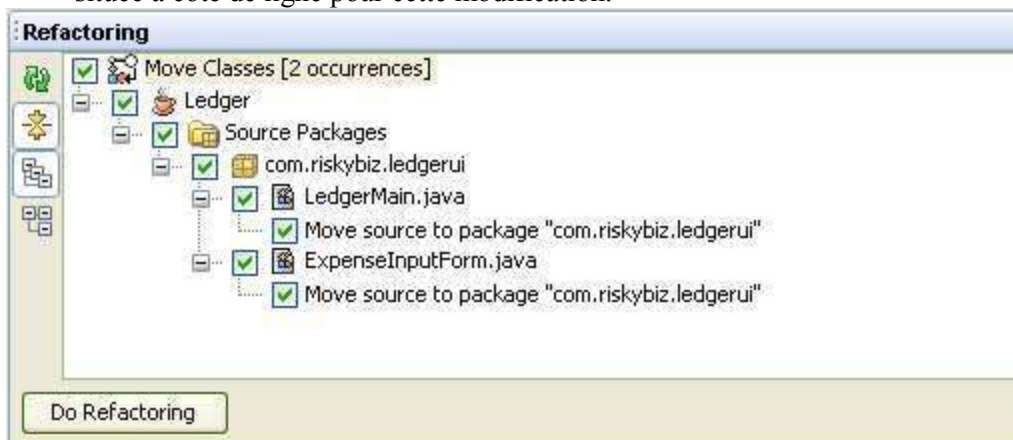
Si vous désirez déplacer une classe dans un autre paquetage, vous pouvez utiliser les fonctionnalités de refactoring de l'EDI pour mettre à jour automatiquement les références à cette classe à travers votre projet.

Pour déplacer une classe:

1. Dans la fenêtre Projects, prenez la classe de son paquetage actuel et déposez-la dans le paquetage où vous désirez la placer. (Vous pouvez également utiliser les commandes Couper et Coller du menu contextuel ou leurs raccourcis clavier correspondants.)
2. Dans la boîte de dialogue Refactor Code For Moved Class, cliquez sur Next après avoir vérifié que les champs To Package et This Class reflète le paquetage de destination et la classe que vous désirez déplacer.



3. Dans la fenêtre Refactoring, regardez à la prévisualisation du code à charger. S'il y a une modification que vous ne désirez pas faire, décochez la case située à côté de la ligne pour cette modification.



4. Cliquez sur Do Refactoring.

Si vous trouvez que le refactoring a eu des conséquences que vous aimeriez annuler, vous pouvez choisir Refactor | Undo

Conseil EDI NetBeans

Vous pouvez également renommer en-ligne un paquetage dans la fenêtre Project pour initier un Move Class refactoring. Cependant, vous ne pouvez pas faire un renommage en-ligne d'un dossier dans la fenêtre Files.

Changer la Signature d'une Méthode

Si vous désirez changer la signature d'une méthode, vous pouvez utiliser la commande Refactor | Change Method Parameters de l'EDI pour mettre à jour d'autres codes dans votre projet qui utilise cette méthode. Voici un aperçu de ce que vous pouvez modifier à la signature d'une méthode:

- Ajouter des paramètres.
- Modifier l'ordre des paramètres.
- Modifier l'accès de la méthode.

Vous ne pouvez pas utiliser la commande Change Method Parameters pour supprimer un paramètre d'une méthode. Si vous enlevez un paramètre d'une méthode, vous avez à mettre à jour manuellement tout code qui fait référence à cette méthode. .

Pour changer la signature d'une méthode:

1. Cliquez-droit sur la méthode dans l'Éditeur de Source ou la fenêtre Projects et choisissez Refactor | Change Method Parameters.
2. Cliquez sur Add si vous désirez rajouter des paramètres à la méthode. Éditez ensuite les cellules Name, Type et (optionnellement) Default Value cells pour le paramètre. Vous avez à double-cliquer sur la cellule pour la rendre éditable.
3. Pour modifier l'ordre des paramètres, sélectionnez un paramètre dans le tableau Parameters et cliquez sur Move Up ou or Down.
4. Sélectionnez l'accès de la méthode depuis la liste Visibility Modifier.
5. Cliquez sur Next.
6. Dans la fenêtre Refactoring, regardez au code qui devra être modifié. S'il y a des modifications que vous ne pouvez faire, décochez la case située à coté de la ligne.
7. Cliquez sur Do Refactoring.

Si vous trouvez que le refactoring a eu des conséquences que vous désirez annuler, vous pouvez choisir Refactor | Undo.

Commandes Refactoring

L'EDI NetBeans a un support spécifique pour le refactoring du code. Le terme refactoring se réfère à renommer et réarranger du code sans modifier ce que fait le code. Les raisons du refactoring inclut des choses comme le besoin d'une API séparée pour l'implémentation, rendant le code plus facile à lire, plus facile à réutiliser.

Le support de refactoring de l'EDI rend le refactoring plus aisé en vous permettant de mettre à jour automatiquement tout le code dans votre projet pour refléter les modifications que vous faites dans d'autres parties de votre projet.

Par exemple, si vous renommez une classe, les références à cette classe dans d'autres classes sont également mises à jour.

Vous pouvez accéder à la plupart des commandes depuis le menu Refactor ou en cliquant droit dans l'Éditeur de Source ou sur le noeud d'une classe dans la fenêtre Projects pour ensuite choisir le sous-menu menu Refactor. La commande Find Usages est dans le menu Edit et le menu contextuel de l'Éditeur de Source et de la fenêtre Projects.

Typiquement, l'identifiant sélectionné est considéré comme l'élément de code à refactoriser.

Le tableau 8 fournit un résumé des commandes de refactoring qui sont disponibles.

Tableau 8: *Commandes Refactoring*

Commande	Description
Rename	Renomme toutes les occurrences de la classe, interface, méthode ou champ sélectionné. Voir le sujet Renommer Toutes les Occurrences de la Classe, Méthode, Champ Sélectionné
Change Method Parameters	Vous permet de modifier les paramètres et l'accès d'une méthode donnée . Voir le sujet Modifier la Signature d'une Méthode
Encapsulate Fields	Génère les méthodes d'accès ("getters" et "setters") pour un champ. Voir le sujet Génération Propriétés Read/Write.
Move Class	Déplace une classe dans un paquetage différent et met à jour toutes les références pour cette classe avec le nom du nouveau paquetage. Voir le sujet Déplacer un Classe dans un Différent Paquetage
Find Usages	Affiche toutes les occurrences du nom d'une classe, méthode ou champ donné. Voir le sujet Trouver des Occurrences de la Classe, Méthode ou Champ Sélectionné

Consolider Notes à Vous-Même

L'EDI a une fonctionnalité de liste de tâches qui vous fournit un moyen d'écrire des notes dans votre code et de voir toutes ces notes dans une seule liste de tâches (ou "à faire"). Vous pouvez utiliser la liste de tâches comme centre d'opérations lorsqu'il vous faut nettoyer quelque peu votre code.

Une ligne est affichée dans la liste de tâches si elle est "taggée" avec l'un des textes suivants:

- @todo
- TODO
- FIXME
- XXX
- PENDING
- <<<<<<<<

Conseil EDI NetBeans

Lorsque vous tapez un tag dans votre code, il doit être tapé en un seul mot pour être reconnu par l'EDI. Par exemple, si vous ne mettez pas un espace entre le tag et la note, la note n'apparaîtra pas dans la liste des tâches.

Pour voir la liste des tâches, choisissez Window | To Do (ou pressez Ctrl-6).




Une fois la fenêtre To Do affichée, vous pouvez voir les tâches concernant le fichier en cours d'édition, tous les fichiers ouverts, ou ceux d'un répertoire spécifique en pressant le bouton correspondant en haut de la fenêtre To Do.

Vous pouvez trier les tâches par leur nom, situation ou priorité, en cliquant sur les titres des colonnes correspondantes. Voir ci-dessous pour comment afficher la colonne de priorité.

Vous pouvez vous rendre depuis une entrée dans la liste des tâches directement à la bonne ligne de code dans le fichier, en double-cliquant sur l'entrée.


Ajouter, Enlever et Modifier des Tags pour la Liste des Tâches

Pour modifier les tags utilisés pour la liste des tâches:

1. Choisissez Tools | Options et sélectionnez le noeud Editing | To Do Settings.
2. Cliquez sur le bouton  de la propriété Task Tags.
3. Dans la boîte de dialogue To Do Settings, utilisez les boutons Add, Change, et Delete pour modifier le contenu de la table Tag List.


Afficher les Tâches Par Priorité

Vous pouvez également afficher les priorités pour chacune des tâches. Les priorités disponibles sont High, Medium-High, Medium, Medium-Low, et Low.

Par défaut, la colonne Priority n'est pas affichée. Vous pouvez afficher la colonne Priority en cliquant sur l'icône  et en sélectionnant la case Priority dans la boîte de dialogue Change Visible Columns.

Des valeurs de priorité sont assignées à chaque tag. Par défaut, tous les tags ont reçu la priorité Medium excepté pour le tag <<<<<<, qui a reçu la priorité High.


Pour modifier la valeur de priorité d'un tag:

1. Choisissez Tools | Options et sélectionnez le noeud Editing | To Do Settings.
2. Cliquez sur le bouton  de la propriété Task Tags.
3. Dans la boîte de dialogue To Do Settings, sélectionnez la nouvelle priorité dans la liste déroulante de la colonne Priority pour le tag dont vous désirez changer la priorité.

Filtrer les Entrées de la Liste de Tâches

Vous pouvez configurer l'affichage de la liste des tâches en créant et utilisant des filtres. Lorsque vous utilisez un filtre, seuls les entrées correspondantes au critère de recherche spécifié par le filtre sont affichés. Les critères incluent le texte qui apparaît dans la note, la priorité de la tâche, et/ou le nom du fichier.

Pour créer un filtre:

1. Cliquez sur l'icône  dans la barre d'outils de la fenêtre To Do .
2. Dans la boîte de dialogue Edit Filters, cliquez sur le bouton New et tapez ensuite un nom pour le filtre dans le champ Name.
3. Remplissez les détails pour le critère.
4. Optionnellement, ajouter un critère supplémentaire en cliquant sur le bouton More et en remplissant les détails pour les filtres. Vous pouvez indiquer que le filtre s'applique à tous ou quelques critères en utilisant les boutons radios en haut de la boîte de dialogue.

Une entrée pour le filtre nouvellement défini apparaît dans une liste déroulante dans la barre d'outils de la fenêtre To Do.

Comparer les Différences Entre Deux Fichiers

Vous pouvez générer une comparaison côte à côte de deux fichiers avec les lignes différentes surlignées. Pour comparer deux fichiers, sélectionnez les noeuds des deux fichiers dans la fenêtre Projects et choisissez Tools | Diff.

Les “différences” apparaissent en tant qu'onglet dans l'Éditeur de Source.

Conseil EDI NetBeans

La commande Diff n'apparaît dans le menu Tools que lorsque deux (et uniquement deux) fichiers sont sélectionné dans la fenêtre Projects, Files ou Favorites.

Scinder l'Éditeur de Source

Vous pouvez scinder l'Éditeur de Source pour voir deux fichiers simultanément ou voir différentes parties du même fichier.

Pour scinder la fenêtre de l'Éditeur de Source:



1. Assurez-vous qu'au moins deux fichiers soient ouverts.
2. Cliquez sur l'onglet d'un fichier, maintenez enfoncé le bouton de la souris, et déplacez l'onglet à l'extrême gauche, extrême droite ou vers le bas de la fenêtre Éditeur de Source.
3. Lâchez le bouton de la souris lorsque le cadre rouge indiquant l'emplacement de la fenêtre scindée apparaît.

Pour voir différentes parties du même fichier simultanément:

1. Cliquez sur l'onglet du fichier dans l'Éditeur de Source et choisissez Clone Document pour créer un second onglet pour le même document.
2. Tirez-lâchez un des onglet du fichier pour créer une deuxième fenêtre de l'Éditeur de Source (voir la procédure à suivre ci-dessus).


Maximiser l'Espace Pour l'Éditeur de Source

Il y a tout un nombre de choses que vous pouvez faire pour donner plus d'espace à l'éditeur dans l'EDI comme:


- Maximiser un fichier dans l'Éditeur de Source de l'EDI en double-cliquant sur l'onglet du fichier. Lorsque vous faites cela, le fichier occupe tout l'espace de l'EDI excepté le menu principal et les barres d'outils. Vous pouvez faire réapparaître les autres fenêtres comme auparavant en double-cliquant à nouveau sur l'onglet.
- Faire "coulisser" les autres fenêtres comme cela elles n'apparaissent que lorsque vous cliquez ou déplacez le curseur de souris sur le bouton représentant cette fenêtre situé sur le contour de l'EDI. Vous pouvez rendre une fenêtre "coulissante" en cliquant sur son icône . Vous pouvez ramener la fenêtre dans son état normal en cliquant sur le bouton  dans la fenêtre "coulissante".
- Cachez les barres d'outils de l'EDI. Vous pouvez cacher les barres d'outils principales en choisissant View | Toolbars et ensuite sélectionnez chaque barre d'outils que vous désirez cacher. Vous pouvez cacher la barre d'outil de l'Éditeur de Source en choisissant Tools | Options, sélectionnant Editing | Editor Settings, et décochez la case pour la propriété Show Toolbar.

Modifier les Raccourcis Clavier de l'Éditeur de Source

Vous pouvez modifier les raccourcis clavier existant ou faire correspondre d'autres commandes disponibles aux raccourcis. Pour modifier les raccourcis clavier de l'Éditeur de Source:

1. Choisissez Tools | Options et sélectionnez le noeud Editor | Editor Settings.
2. Cliquez sur le bouton  dans la propriété Global Key Bindings pour ouvrir la boîte de dialogue Global Key Bindings.
3. Sélectionnez une combinaison de touches que vous désirez modifier, cliquez sur Add, et tapez un raccourci que vous devez utiliser.

Conseil EDI NetBeans

Il est également possible de rajouter des raccourcis clavier pour des types de fichiers spécifiques. Dans la fenêtre Options, étendez le noeud Editor Settings et sélectionnez le noeud pour l'éditeur du type de fichier dont vous désirez ajouter un raccourci. Cliquez sur le bouton  de la propriété Key Bindings pour ouvrir la boîte de dialogue Key Bindings pour ce type d'éditeur.

Les différents types d'éditeurs héritent les raccourcis du niveau global et il n'est pas possible d'override ces raccourcis globaux pour un type d'éditeur spécifique.

Vous pouvez modifier des raccourcis clavier qui ne sont pas liés à un éditeur en choisissant Tools | Keyboard Shortcuts et en ajoutant et/ou supprimant des raccourcis clavier.

Conseil EDI NetBeans

Certaines combinaisons de touches, comme ceux pour le clipboard et d'autres opérations courantes, sont assignées comme raccourcis aussi

bien dans l'Éditeur de Source qu'en dehors de celui-ci. Par exemple, Ctrl-F invoque une boîte de dialogue Find différente selon qu'elle est invoquée depuis un fichier dans l'Éditeur de Source ou un dossier dans la fenêtre Projects.

Si vous modifiez de tels raccourcis clavier dans l'Éditeur de Source, il n'affectera pas le raccourcis clavier pour les autres parties de l'EDI et vice versa.