

Exécution d'une révision de code

Ce tutoriel présente certaines des caractéristiques et fonctions de la révision de code. Il s'adresse aux développeurs de logiciels.

Durée nécessaire

Pour une simple lecture de ce tutoriel, comptez environ **15 minutes**. Pour effectuer les exercices en utilisant le projet exemple fourni, prévoyez environ **30 minutes**.

Conditions préalables

Pour pouvoir suivre ce tutoriel, vous devez être familiarisé avec le développement d'applications Java. Il est aussi utile de savoir comment utiliser les perspectives et les vues d'IBM Rational Software Development Platform.

Objectifs de la formation

Ce tutoriel est divisé en sections qu'il convient de suivre dans l'ordre. Vous lirez d'abord une présentation des avantages des révisions de code automatisées, puis vous apprendrez à accomplir les tâches suivantes :

- Exécuter une révision de code
- Appliquer une correction rapide fournie pour résoudre un problème dans le code

Lorsque vous êtes prêt, commencez par la section "Aperçu de la révision de code."

Aperçu de la révision de code

Rôle

Une révision de code est un ensemble de règles qui automatisent le processus de révision de code par un développeur ou un architecte logiciel. Tandis que la révision manuelle peut s'avérer longue et donner lieu à des discussions subjectives, la révision automatisée est efficace, rapide et cohérente. La révision automatisée complète la révision manuelle. Elle ne la remplace pas.

Avantages

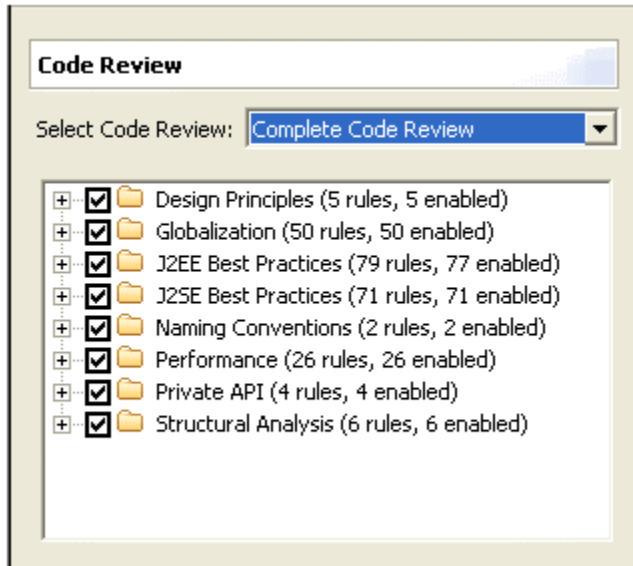
En exécutant les tâches suivantes, l'outil de révision de code automatisée valorise le processus de développement de logiciels de plusieurs manières :

- Il détecte les bogues dans le code.
- Il vérifie la conformité aux pratiques recommandées.
- Il explique chaque anomalie constatée et fournit des solutions pour celle-ci.
- Il fournit une correction automatique pour certaines anomalies typiques.
- Il vous permet de créer des règles pour garantir que vous suivez les standards de conception de l'application à mesure que vous écrivez le code.

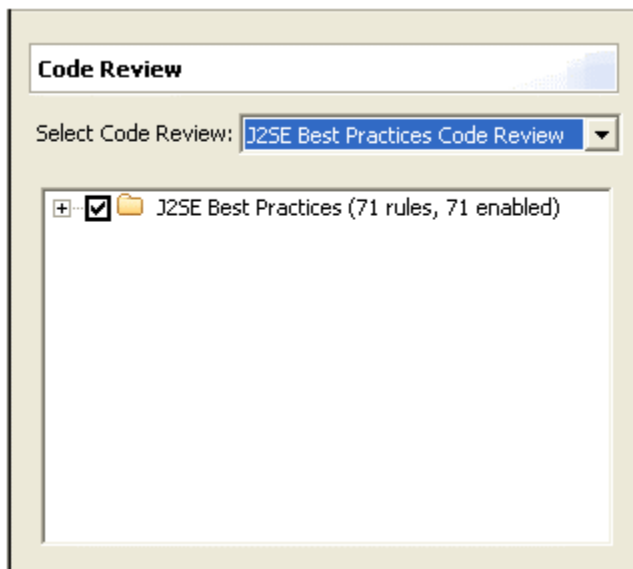
Comme le processus automatisé est rapide, vous pouvez exécuter des révisions de code aussi souvent que vous le souhaitez. Les constatations faites par la révision de code permettent de détecter et de corriger les problèmes très tôt dans le cycle de développement, à un stade où l'application de modifications est encore simple et peu coûteuse.

Révisions de code fournies

Plusieurs révisions de code sont fournies. Chacune applique un jeu de règles différent, organisé en dossiers. Selon l'état d'avancement de votre processus de développement et vos objectifs de révision, vous pouvez choisir la révision de code qui convient le mieux à vos besoins. La plus globale d'entre elles est la révision de code complète. Elle applique les règles de toutes les catégories, comme le montre la capture d'écran suivante :



Certaines catégories sont aussi associées à une révision de code. Par exemple, vous pouvez sélectionner la révision intitulée Pratiques recommandées dans J2SE lorsque vous souhaitez n'appliquer que les règles de cette catégorie, comme le montre la capture d'écran ci-dessous. Vous pouvez ainsi exécuter des révisions concentrées sur un aspect particulier du code.






Révisions de code définies par l'utilisateur

Vous pouvez créer des règles à l'aide d'un assistant prévu à cet effet. Celui-ci vous permet de choisir entre deux types de règles : Contrôle de l'architecture et Général. Les architectes logiciels peuvent ainsi étendre les capacités de la révision de code en créant leurs propres règles pour vérifier l'intégrité de leurs conceptions.

Niveaux de gravité des règles

A chaque règle est associé un niveau de gravité. Vous pouvez modifier le niveau de gravité affecté aux règles fournies. Lorsque vous créez une règle à l'aide de l'assistant, vous spécifiez également son niveau de gravité. Les trois niveaux de gravité disponibles sont symbolisés par les icônes suivantes :

- Problème () : Cette anomalie doit être traitée.
- Avertissement () : Cette anomalie est probablement un problème qui doit être traité.
- Recommandation () : Cette anomalie n'est pas encore grave, mais il est vivement recommandé de la traiter dans l'immédiat.

Bien que le niveau de gravité "Recommandation" soit le plus bas, ne sous-estimez pas son importance et la nécessité de traiter les anomalies correspondantes. Les recommandations reflètent un ensemble de pratiques conseillées et de normes de l'industrie auxquelles doivent se conformer les équipes de développement. Même si ces anomalies ne sont pas des problèmes immédiats, elles peuvent le devenir dans le futur.

La capture d'écran suivante montre les règles présentes dans le dossier Comparaison de la révision de code Pratiques recommandées dans J2SE. Ensemble, ces règles couvrent les trois niveaux de gravité.



Corrections automatisées pour certains problèmes

Certaines anomalies courantes s'accompagnent d'une correction rapide, laquelle est une solution automatisée. S'il existe une correction rapide pour une anomalie constatée dans la révision de code, elle est indiquée par l'une des icônes illustrées ci-après :



Résumé

Dans le cycle de développement de logiciels, la révision de code automatise le processus d'examen manuel du corps d'un code. Les révisions de code fournies en standard permettent de procéder aux révisions suivantes :

- Révision globale ou complète, appliquant un large éventail de règles de toutes les catégories à une base de code.
- Révision ciblée, appliquant les règles d'une ou de plusieurs catégories spécifiques, telles que la globalisation et les principes de conception.

Vous disposez aussi d'un assistant pour créer vos propres règles visant à garantir l'intégrité de la structure de conception de votre application.

Comme les révisions de code sont rapides à exécuter, vous pouvez détecter au plus tôt les problèmes et incohérences dans une base de code. Vous pouvez dès lors corriger ces problèmes avant qu'ils n'affectent la maintenabilité, l'extensibilité et les performances de votre application.

Vous voici prêt à passer à la section “Exercice 1.1 : Importation des ressources requises.”

Exercice 1.1 : Importation des ressources requises

Cet exercice vous explique comment importer le projet exemple, CodeReview_Examples. Vous utiliserez ce projet pour effectuer l'Exercice 1.2 : Exécution d'une révision de code et application d'une correction rapide."

Décompression du projet exemple

Le projet exemple accompagnant ce tutoriel est inclus dans un fichier ZIP. Les étapes suivantes vous montrent comment extraire les fichiers de ce fichier ZIP pour les placer dans le dossier de votre espace de travail.

1. Naviguez jusqu'à
<rép_install>\rad\eclipse\plugins\com.ibm.r2a.rad.tutorial.doc_6.0.0\resources,
emplacement du fichier ZIP CodeReview_Examples.
2. Extrayez le contenu de CodeReview_Examples dans
<rép_install>\updater\eclipse\workspace. Les fichiers du projet exemple sont extraits
dans le dossier de votre espace de travail afin que vous puissiez les importer.

Ouverture de la vue Révision de code

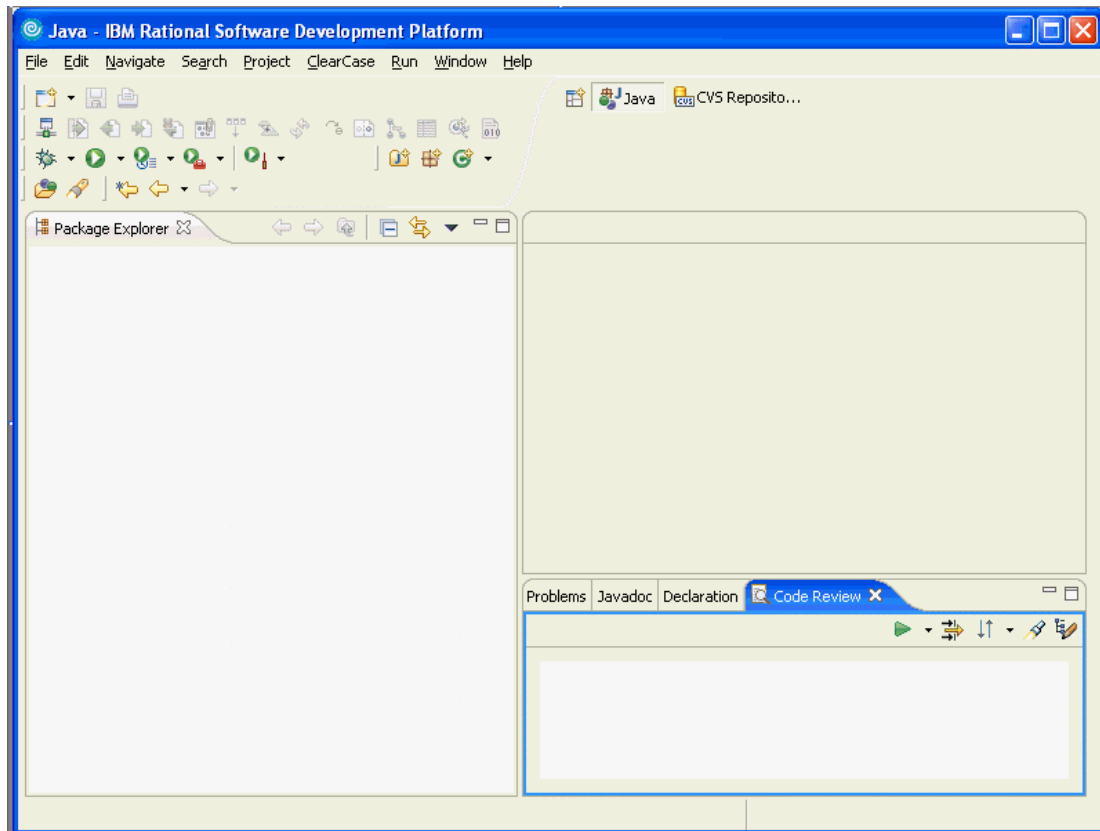
Procédez comme suit pour ouvrir une perspective montrant la vue Révision de code :

1. Démarrez IBM Rational Software Development Platform 6.0.



2. Cliquez sur **Fenêtre > Préférences**.
3. Dans le panneau de gauche, développez la branche **Plan de travail** et cliquez sur **Capacités**.
4. Dans la liste **Capacités**, cliquez sur **Développeur Java**. Cliquez ensuite sur **OK**.
5. Cliquez sur **Fenêtre > Ouvrir la perspective > Java**.
6. Cliquez sur **Fenêtre > Afficher la vue > Autre > Java > Révision de code**.
7. Cliquez sur **Fenêtre > Afficher la vue > Autre > Java > Vue Packages**.

Une fois la perspective Java ouverte et les vues Révision de code et Packages affichées, l'écran doit se présenter comme celui illustré ci-après. Il est toutefois possible que votre agencement soit différent, c'est-à-dire que les vues peuvent être affichées à d'autres endroits. Dans le présent tutoriel, l'emplacement des vues est décrit en tenant compte de l'agencement de cette capture d'écran.



Importation du projet exemple

Pour importer le projet exemple dans l'espace de travail :

1. Cliquez avec le bouton droit sur la vue Packages pour ouvrir son menu contextuel. Cliquez ensuite sur **Importer** pour ouvrir l'assistant Importation.
2. Dans la liste **Sélection**, cliquez sur **Projet existant dans l'espace de travail**. Cliquez ensuite sur **Suivant**.
3. Cliquez sur le bouton **Parcourir** associé à la zone de saisie **Contenu du projet** et sélectionnez `<rep_install>\updater\eclipse\workspace\CodeReview_Examples`.
4. Cliquez sur **Fin**. Le projet exemple et tous les fichiers associés sont importés dans l'explorateur de packages (vue Packages).

Début de l'exercice

Pour commencer l'exercice, allez à la section "Exercice 1.2 : Exécution d'une révision de code et application d'une correction rapide."

Exercice 1.2 : Exécution d'une révision de code et application d'une correction rapide

On suppose ici que vous avez effectué les étapes de l'“Exercice 1.1 : Importation des ressources requises.” Dans le présent exercice, vous commencez par lire un scénario mettant en jeu plusieurs utilisateurs. Puis vous prenez le rôle du développeur de logiciels décrit dans ce scénario.

Scénario utilisateurs

Une grande équipe de développeurs dispersés géographiquement travaille à l'écriture d'une nouvelle application logicielle. Il est important que les développeurs exécutent régulièrement des révisions de code afin de détecter les problèmes dans leur code.

L'un des développeurs souhaite exécuter une révision de code pour avoir un aperçu général de son travail. Pour passer en revue le code qu'il vient d'écrire et évaluer sa conformité aux pratiques recommandées dans plusieurs domaines, il exécute une procédure automatisée appelée Révision de code rapide. Cette révision applique au code plusieurs catégories de règles fournies en standard avec le produit. Chaque catégorie de règles contrôle la qualité du code dans un domaine spécifique tel que les performances.

Une fois la révision de code terminée, une liste de constatations, ou "anomalies", s'affiche à l'écran. Chaque anomalie représente une chaîne de code qui ne respecte pas strictement une règle appliquée. Pour l'une des anomalies constatées, une correction rapide est disponible. Le développeur applique donc la solution automatisée et corrige immédiatement le problème constaté.

Dans la première partie de l'exercice, vous accomplissez les tâches suivantes en vue d'exécuter une révision de code :

1. Sélectionner une révision de code à exécuter.
2. Visualiser les règles appliquées dans la révision de code.
3. Choisir sur quel code exécuter la révision.
4. Exécuter la révision de code.
5. Visualiser les anomalies constatées par l'exécution de la révision de code.
6. Sélectionner une anomalie pour afficher les informations suivantes la concernant :
 - Code source.
 - Description, exemples et solutions.

Ensuite, pour appliquer une correction rapide à l'une des anomalies constatées dans la révision de code, vous accomplissez les tâches suivantes :

1. Déterminer si une correction rapide est disponible pour une anomalie.
2. Visualiser la liste des modifications qui seront apportées au code par la correction rapide.

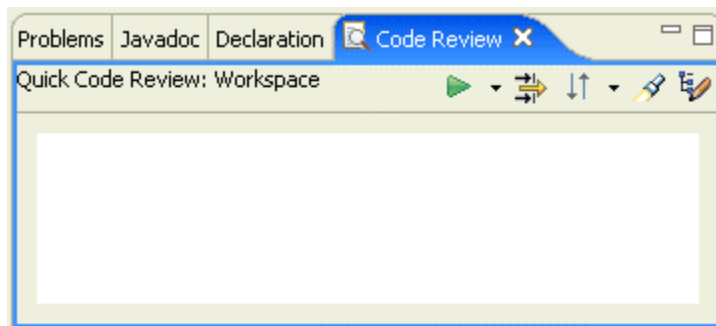
3. Prévisualiser la version initiale et la version restructurée du code avant d'appliquer la correction rapide.
4. Appliquer la correction rapide pour restructurer le code.
5. Obtenir une confirmation de l'application de la correction.

Exercice

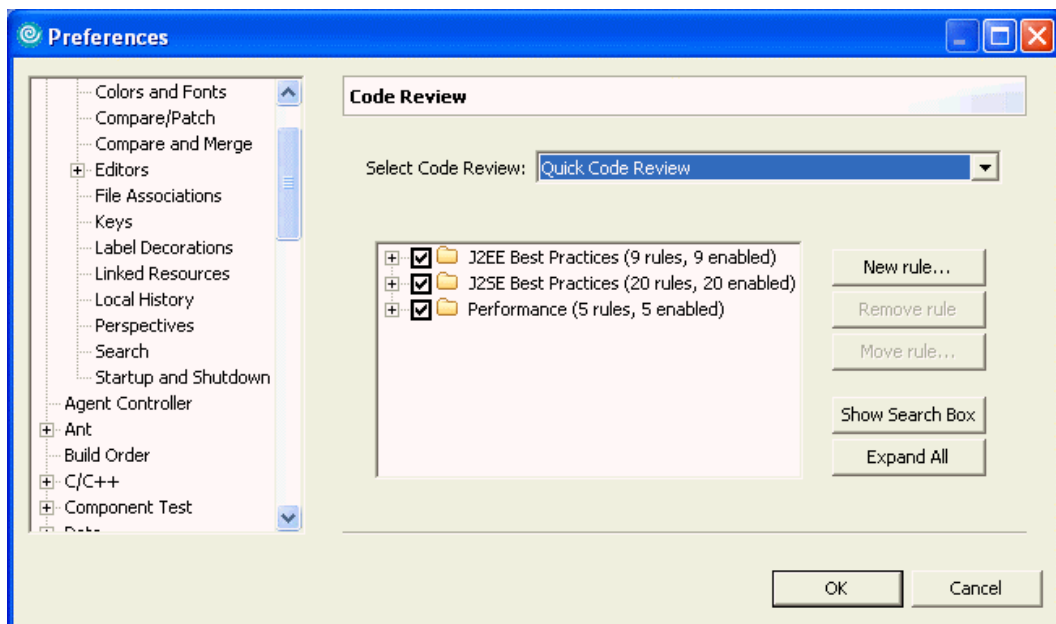
Sélection d'une révision de code

Pour sélectionner une révision de code rapide :

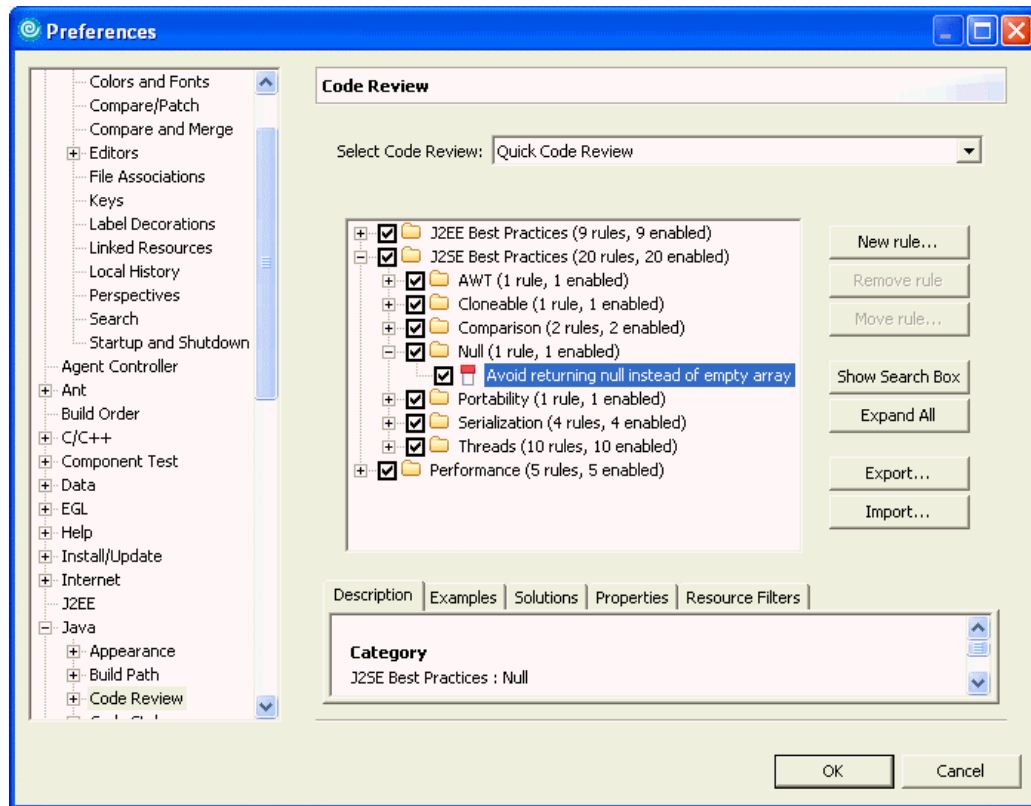
1. Sur la barre d'outils de la vue Révision de code, cliquez sur l'icône **Gérer les règles**, .



2. Dans la liste **Sélectionnez la révision de code**, cliquez sur **Révision de code rapide**. Les dossiers de règles de la révision de code sélectionnée s'affichent, comme l'illustre la capture d'écran suivante :



3. Pour voir l'une des règles qui sera appliquée dans la révision de code, développez la branche du dossier **Pratiques recommandées dans J2SE**, puis celle du sous-dossier **Valeur NULL**. Ce sous-dossier contient une seule règle, dont la gravité est de niveau Problème (voir l'illustration ci-dessous).



Pour mémoire, les icônes symbolisant les différents niveaux de gravités sont les suivantes :

Icon	Severity Level
	Problem
	Warning
	Recommendation

4. Cliquez sur **OK** pour choisir la Révision de code rapide.

Sélection d'une base de code à réviser

Procédez comme suit pour sélectionner le projet de la base de code à réviser :

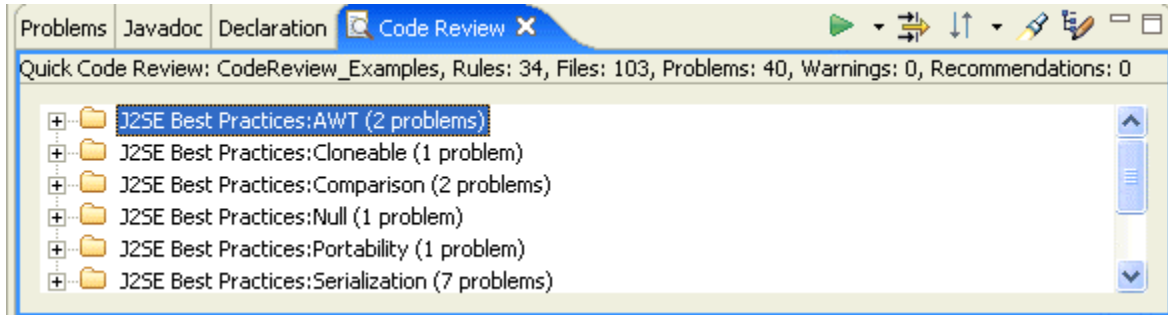
- Sur la barre d'outils de la vue Révision de code, cliquez sur l'icône **Révision** () > **Projets** > **Révision de CodeReview_Examples**.

Exécution de la révision de code

Une fois que vous avez sélectionné la base de code à réviser, la révision de code s'exécute. Vous pouvez suivre son état en observant la barre de progression, dans l'angle inférieur droit de la vue.

Visualisation des anomalies constatées par la révision de code

Une fois la révision de code terminée, les anomalies constatées sont présentées dans la vue Révision de code, comme le montre l'illustration ci-après :



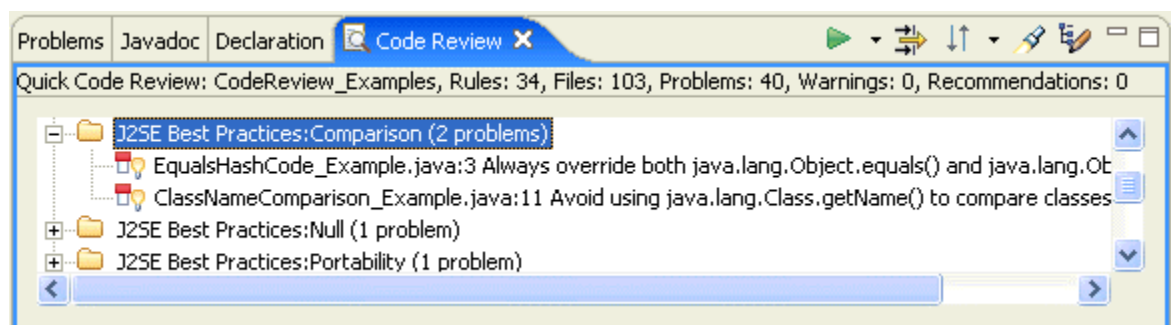
La vue Révision de code fournit les informations suivantes :

- Statistiques : la ligne située au-dessus des anomalies constatées donne des détails sur la révision de code la plus récente : type, portée, nombre de règles et de fichiers inclus, nombre et gravité des anomalies constatées.
- Anomalies constatées : les anomalies constatées par la révision de code sont répertoriées sous forme de dossiers dans la vue Révision de code. Chaque nom de dossier mentionne la catégorie de règles appliquées ainsi que le nombre d'anomalies.

Obtention d'informations complémentaires sur une anomalie constatée

Procédez comme suit pour obtenir plus d'informations sur une anomalie constatée dans la révision de code :

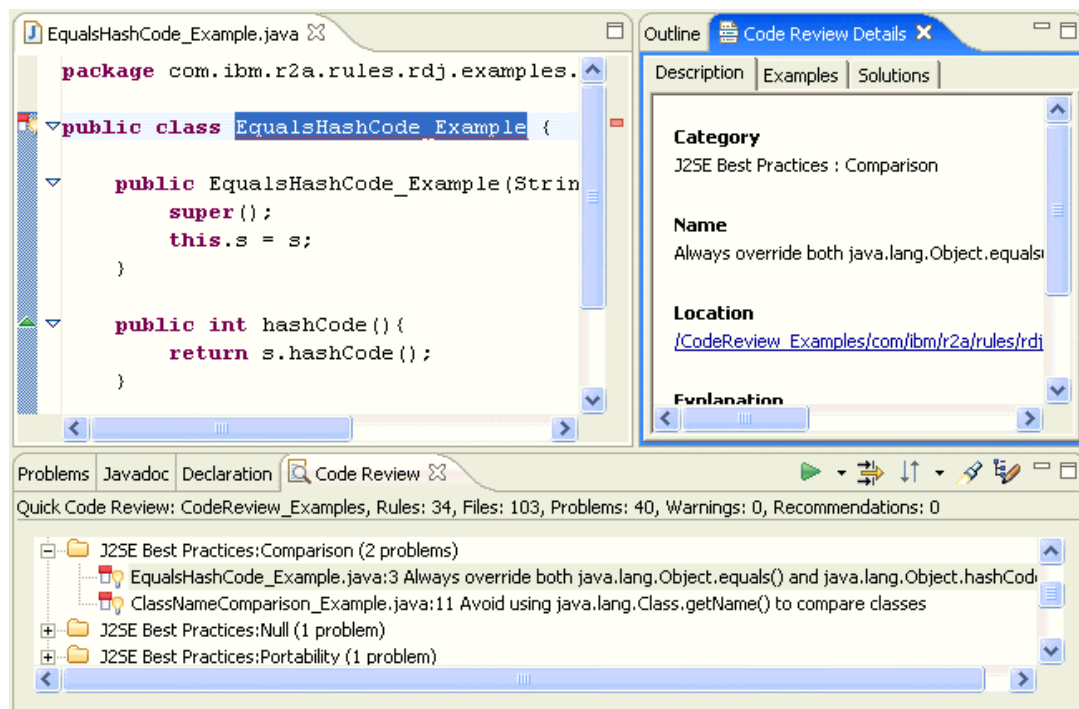
1. Faites défiler la vue Révision de code pour atteindre le dossier **Pratiques recommandées dans J2SE : Comparaison**. Développez la branche de ce dossier pour révéler les anomalies qu'il contient, comme illustré ci-après :



2. La première anomalie constatée commence par EqualsHashCode_Example.java. La règle appliquée figure après :

Remplacez toujours `java.lang.Object.equals()` et `java.lang.Object.hashCode()`

3. Cliquez deux fois sur la première anomalie. Les détails la concernant apparaissent en deux endroits, comme décrit et illustré ci-après :
 - Code source : affiche le code dans lequel l'anomalie est constatée et met en évidence l'endroit précis où elle se situe.
 - Vue Détails de la révision du code : décrit l'anomalie plus en détail et fournit des exemples et des solutions pour la corriger.



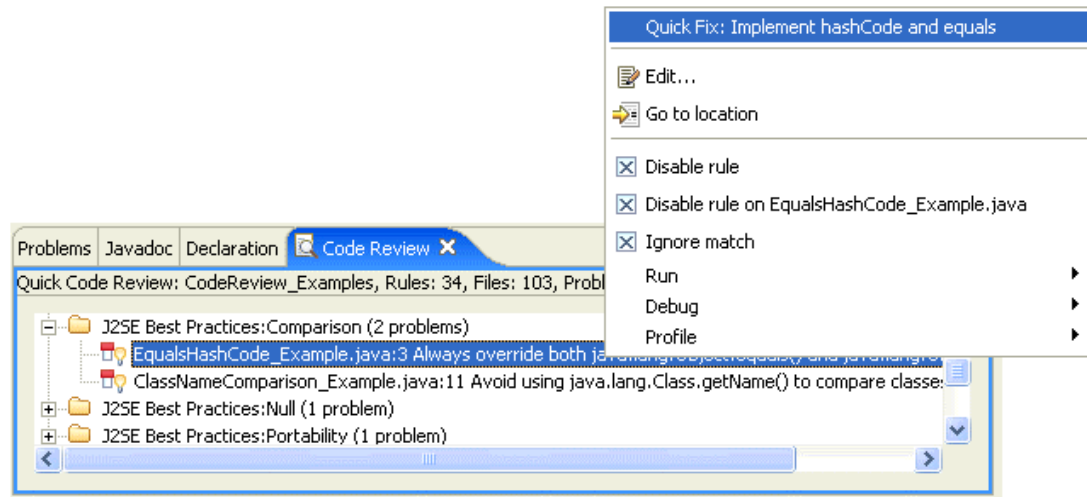
Sélection d'une anomalie pour laquelle il existe une correction rapide

Les icônes symbolisant les deux anomalies dans le dossier Pratiques recommandées dans J2SE : Comparaison montrent qu'il existe une correction rapide pour celles-ci. Pour mémoire, les icônes symbolisant les corrections rapides sont les suivantes :



1. Cliquez avec le bouton droit sur la première anomalie répertoriée dans la liste (voir l'illustration suivante).

2. Les choix proposés dans le menu contextuel **Correction rapide** varient en fonction de la solution. Pour l'anomalie sélectionnée ici, la correction consiste à implémenter hashCode et equals.



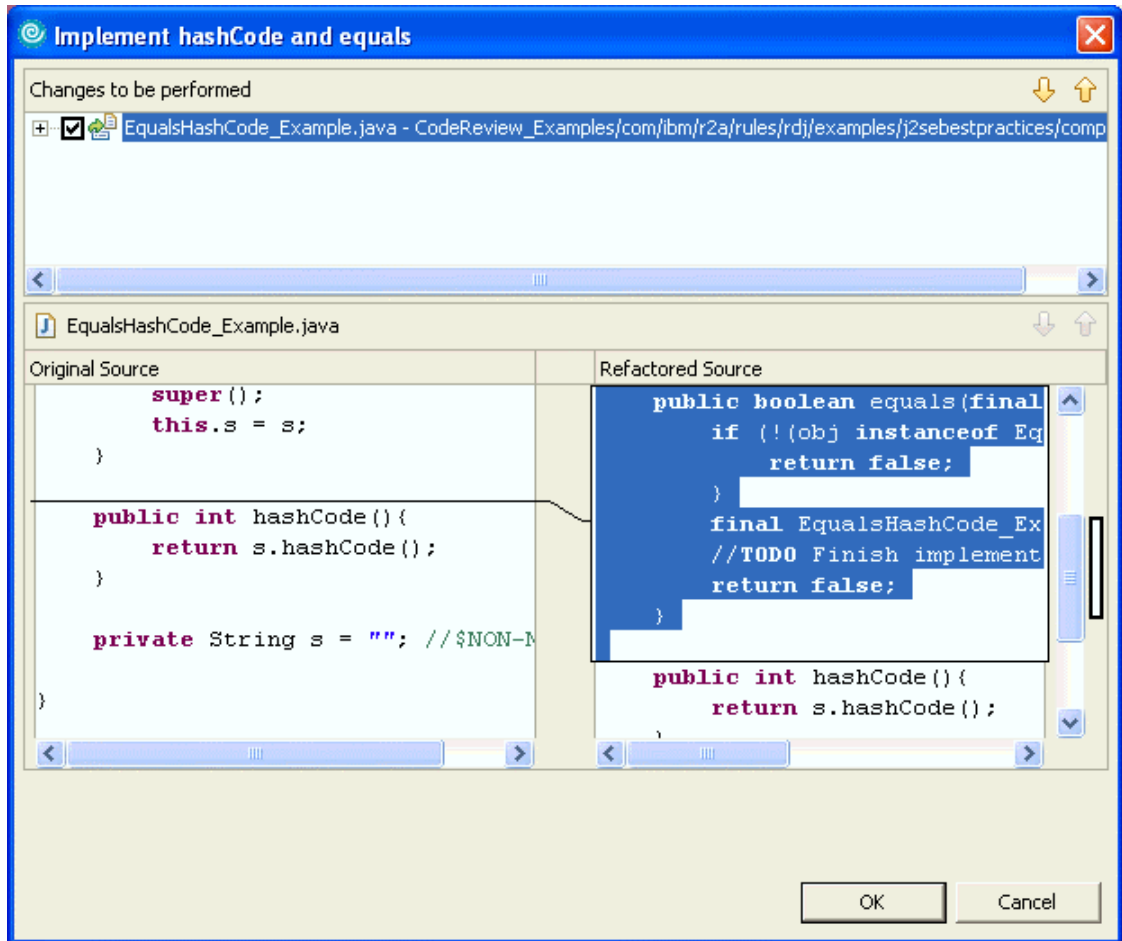
3. Cliquez sur **Correction rapide : Implémentez hashCode et equals**.

Application de la correction rapide

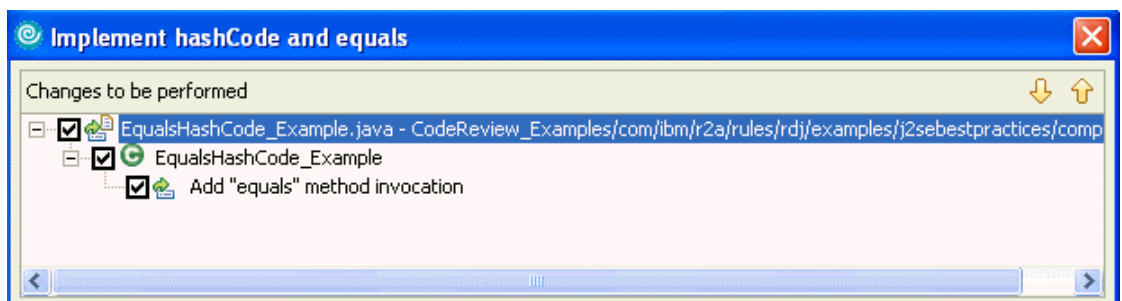
Pour l'anomalie que vous avez sélectionnée, la correction rapide consiste à implémenter hashCode et equals.

Pour examiner la correction rapide et l'appliquer à l'anomalie :

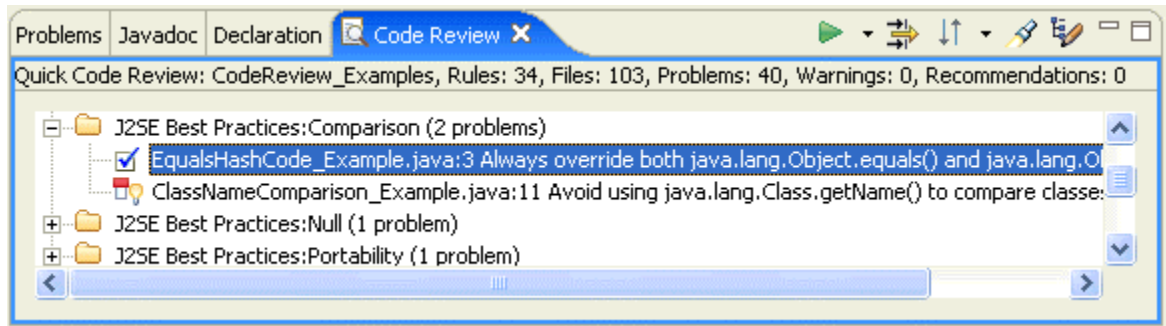
1. La vue affichée, illustrée ci-après, présente deux sous-fenêtres de code placées l'une à côté de l'autre. Le code source d'origine se trouve à gauche, tandis que le code restructuré qui serait créé par la correction rapide se trouve à droite. Si vous décidez d'appliquer la correction rapide, les lignes mises en évidence dans la sous-fenêtre de droite seront ajoutées au code d'origine.



2. Dans la section **Modifications à apporter**, développez la liste pour voir précisément ce que la correction rapide changera dans le code et de quelle manière elle le fera (voir l'illustration suivante) :



3. Examinez les modifications dans la liste. Cliquez ensuite sur **OK** pour appliquer la correction rapide à toutes les modifications sélectionnées dans la liste.
4. Une fois la correction rapide appliquée, une marque de sélection (coche) figure en regard de l'anomalie que vous venez de résoudre.



A côté de cette marque de sélection figurent les informations suivantes :

- La correction rapide appliquée
- Le numéro de ligne où se situe l'anomalie dans le code source
- La règle qui était spécifiée dans la révision de code et qui n'était pas respectée

L'exercice Exécution d'une révision de code et application d'une correction rapide est terminé.

Conclusion de l'exercice

Vous avez accompli toutes les tâches de l'exercice Exécution d'une révision de code et application d'une correction rapide.

Tâches à accomplir pour exécuter une révision de code

Pour exécuter cette révision de code, vous avez accompli les tâches suivantes :

1. Sélectionner une révision de code à exécuter.
2. Visualiser les règles appliquées dans la révision de code.
3. Choisir sur quel corps de code exécuter la révision.
4. Exécuter la révision de code.
5. Visualiser les anomalies constatées par l'exécution de la révision de code.
6. Sélectionner une anomalie pour afficher les informations suivantes la concernant :
 - Code source.
 - Description, exemples et solutions.

Tâches à accomplir pour appliquer une correction rapide

Pour appliquer la correction rapide, vous avez accompli les tâches suivantes :

1. Déterminer si une correction rapide est disponible pour une anomalie.
2. Visualiser la liste des modifications qui seraient apportées au code par la correction rapide.
3. Prévisualiser le code d'origine et le code restructuré.
4. Appliquer la correction rapide pour restructurer le code.
5. Obtenir une confirmation de l'application de la correction.

Mettre à profit la puissance d'une révision de code

En exécutant régulièrement des révisions de code, vous êtes en mesure d'identifier très tôt les anomalies. Cela signifie que vous pouvez aussi les traiter immédiatement, avant qu'elles ne conduisent aux problèmes suivants :

- Affectation des performances, de la maintenabilité et de l'extensibilité de votre application.
- Augmentation du temps et des ressources nécessaires au développement, et donc du coût pour votre entreprise.

Mettre à profit la puissance d'une correction rapide

En appliquant une correction rapide fournie, vous disposez d'un moyen automatisé de résoudre une anomalie courante. Les corrections rapides présentent les avantages suivants :

- Elles corrigent les problèmes de manière cohérente et identique chaque fois qu'ils surviennent.
- Elles vous soulagent d'une part du codage et vous permettent de passer moins de temps à corriger les bogues.

Terminez ce tutoriel en passant en revue les objectifs de formation, dans la section "Résumé : Exécution d'une révision de code."

Résumé : Exécution d'une révision de code

Ce tutoriel vous a montré comment exécuter une révision de code.

Objectifs de formation atteints

Si vous avez effectué l'exercice, vous devez maintenant être en mesure d'accomplir les tâches suivantes :

- Exécuter une révision de code.
- Appliquer une correction rapide fournie pour résoudre un problème.

Autres sources d'informations

Si vous souhaitez en apprendre plus sur les sujets traités dans ce tutoriel, reportez-vous à l'aide en ligne consacrée à l'exécution des révisions de code.