

[Java development user guide](#) > [Reference](#)

# Refactoring Support

The goal of Java program refactoring is to make system-wide code changes without affecting the behavior of the program. The Java tools provide assistance in easily refactoring code.

The refactoring tools support a number of transformations described in Martin Fowler's book *Refactoring: Improving the Design of Existing Code*, Addison Wesley 1999, such as *Extract Method*, *Inline Local Variable*, etc..

To get an overview of all offered refactorings look at the [Refactor menu](#). Refactoring commands are also available from the context menus in many views or appear as [quick assists](#).

When performing a refactoring operation, you can optionally preview all of the changes resulting from a refactoring action before you choose to carry them out. When previewing a refactoring operation, you will be notified of potential problems and will be presented with a list of the changes the refactoring action will perform. If you do not preview a refactoring operation, the change will be made in its entirety and any resultant problems will be shown. If a problem is detected that does not allow the refactoring to continue, the operation will be halted and a list of problems will be displayed.

Refactoring commands are available from the context menus of several Java views (e.g. Package Explorer, Outline) and editors. Many "apparently simple" commands, such as **Move** and **Rename**, are actually refactoring operations, since moving and renaming Java elements often require changes in dependent files.

Refactorings can not only be performed interactively, but also from refactoring scripts. Most refactorings available in the **Refactor** menu are stored in the workspace refactoring history in order to be used in refactoring scripts afterwards. The refactoring tools support the creation of refactoring scripts based on refactorings in the workspace refactoring history. Refactoring scripts can then be applied to an arbitrary workspace. Applying a refactoring script launches a refactoring wizard which is able to replay the refactorings as if they had been initiated by the user which originally had created them.

Related to refactoring scripts, the refactoring tools offer a refactoring to migrate a JAR File to a newer version, using refactoring information to avoid breaking changes in your workspace after the migration.

## Related references

[Refactoring actions](#)

[Refactoring wizard](#)

[Java preferences](#)