

Replace Temp with Query

This refactoring allows you to extract the variable's initializer expression into a method, and replace all references to the variable with the calls to the extracted method. The declaration of the variable will be removed and the query method can be used in other methods.

Instead of `int size = getActualSize()` and using `size` throughout the code, we just operate with `getActualSize()` method. Though the resulting code has more invocations, it is much cleaner and helps identify precisely where the bottlenecks in the code can appear.

Example

Before	After
<pre>public void method() { String str = "str"; String aString = returnString().concat(str); System.out.println(aString); }</pre>	<pre>public void method() { String str = "str"; System.out.println(aString(str)); } private String aString(String str) { return returnString().concat(str); }</pre>

To replace temp with query

1. In the editor, position the caret at the name of the local variable you want to be refactored.
2. On the main menu, or on the context menu of the selection, choose **Refactor | Replace Temp with Query**. [Replace Temp with Query](#) dialog box appears.

Selecting a local variable, make sure that its value is not modified later in the code. Otherwise the error message appears.

3. Specify the name for the extracted method.
4. To declare the method static, select the **Declare static** check box. This option is enabled when the initial expression is static.
5. In the **Parameters** section, select the parameters to be used in the extracted method. The parameters are all checked by default. If unchecked, the appropriate value will be used as a local variable in the extracted method.
6. Check the result in the **Signature Preview** pane and click **OK** to create the method.

See Also

Reference:

- [Replace Temp with Query Dialog](#)

Web Resources:

- [Developer Community](#) 