**Руководство по Eclipse IDE**

Данный пост является переводом с английского[руководства от Ларса Фогеля](http://www.vogella.de/articles/Eclipse/article.html). Мне, как человеку, который в программировании не новичек, а просто переходит с одной платформы (.Net) на другую (Android+Java+Eclipse), данное руководство дало самую необходимую краткую информацию об отличиях и особенностях среды разработки. Поэтому захотелось перевести его – может еще кому-то будет полезно. Заодно и сам перечитал некоторые моменты, упущенные при первом быстром ознакомлении – читаю я часто слишком уж быстро… Разрешение на перевод и публикацию получено от самого Ларса. Кстати, на [его сайте](http://www.vogella.de/)немало других коротких, но информативных руководств, так что не исключаю, что еще что-нибудь переведу.

Все картинки можно посмотреть в оригинальном размере, просто кликнув на них.

## Ларс Фогель (Lars Vogel)

Version 2.0

## Eclipse Java IDE

Данное руководство описывает использование Eclipse как среды разработки для Java. В нем описана установка Eclipse, создание Java-программ и полезные советы по использованию Eclipse. Данное руководство основано на Eclipse 3.7 (Indigo).

### Содержание

1. Обзор Eclipse  
2. Начало работы  
2.1. Установка  
2.2. Запуск Eclipse  
3. Обзор пользовательского интерфейса Eclipse  
3.1. Workspace  
3.2. Perspective  
3.3. Views и Editors  
4. Создание вашего первого Java-приложения  
4.1. Создание проекта  
4.2. Создание пакета  
4.3. Создание Java класса  
4.4. Запуск проекта в Eclipse  
4.5. Запуск Java-программы извне Eclipse (создаем jar-файл)  
4.6. Запуск вашей программы не из-под Eclipse  
5. Content Assists и Quick Fix  
5.1. Content assist  
5.2. Quick Fix  
6. Использование jars (библиотек)  
6.1. Добавление внешних библиотек (.jar ) в Java classpath  
6.2. Показ исходников jar-библиотеки  
6.3. Добавление Javadoc к библиотеке jar  
7. Обновление и установка плагинов  
7.1. Менеджер обновлений Eclipse  
7.2. Ручная установка плагинов (папка dropins)  
8. Дополнительные подсказки  
8.1. Вид «Problems»  
8.2. Важные настройки  
8.3. Управление заданиями  
8.4. Рабочие множества  
8.5. Синхронизация отображения эксплорера пакетов с редактируемым кодом  
8.6. Шаблоны кода  
9. Дальнейшие шаги  
10. Спасибо вам  
11. Вопросы и обсуждения  
12. Ссылки и литература  
12.1. Исходные коды  
12.2. Ресурсы Eclipse  
12.3. vogella Resources

### 1. Обзор Eclipse

Многие люди знают Eclipse как интегрированную среду разработки (IDE) для Java. Eclipse создан сообществом Open Source и используется в нескольких различных областях, таких как среда разработки для Java или [Android](http://www.vogella.de/articles/Android/article.html) или же как платформа для разработки приложений [Eclipse RCP](http://www.vogella.de/articles/EclipseRCP/article.html) (Rich Client Platform).  
Здесь мы рассмотрим только первый вариант использования Eclipse – как интегрированной среды разработки для Java.

### 2. Начало работы

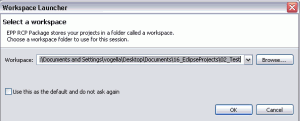
#### 2.1. Установка

Eclipse требует установленную requires [Java](http://www.vogella.de/articles/JavaIntroduction/article.html) Runtime. Я рекомендую использовать Java 6 (также известную как Java 1.6).

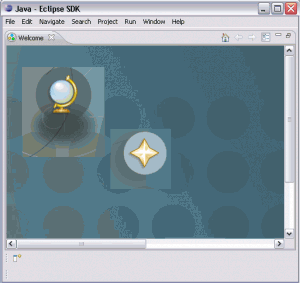
Загрузите “Eclipse IDE for Java Developers” с сайта [Eclipse Downloads](http://www.eclipse.org/downloads/) и распакуйте его. Используйте каталог, путь к которому не содержит пробелов – это иногда приводит к проблемам в Eclipse. После распаковки загруженного Eclipse, он готов к работе. Никакой дополнительной установки не требуется.

#### 2.2. Запуск Eclipse

Для запуска Eclipse, выполните двойной клик на файле “eclipse.exe” (Microsoft Windows) или “eclipse” (Linux / Mac) в папке с распакованным Eclipse. Система предложит вам выбрать рабочее пространство (workspace). Под рабочим пространством подразумевается место, где будут размещаться ваши Java проекты. Выберите пустой каталог и нажмите Ok.

[](http://sotnyk.com/wp-content/uploads/2011/10/01-xstart10.png)

Eclipse запустится и будет показана стартовая страница. Закройте её, нажав “X” рядом с надписью “Welcome”.

[](http://sotnyk.com/wp-content/uploads/2011/10/02-xstart20.png)

### 3. Обзор пользовательского интерфейса Eclipse

Eclipse предоставляет перспективы (perspectives), виды (views) и редакторы (editors). Виды и редакторы сгруппированы в перспективы. Все проекты находятся в рабочем пространстве (workspace).

#### 3.1. Workspace

Рабочее пространство представляет собой физическое расположение того, с чем вы работаете (путь в файловой системе). Вы можете выбрать его во время старта Eclipse либо через меню (File-> Switch Workspace-> Others). Все ваши проекты, исходные файлы, изображения и другие артефакты будут сохранены и записаны в вашем рабочем пространстве.

Вы можете предварительно задать рабочее пространство, используя параметр командной строки -data путь\_к\_workspace, т.е. “c:\eclipse.exe -data “c:\temp”. Пожалуйста, учтите, что вы должны поместить путь в кавычки. Чтобы увидеть каталог текущего рабочего пространства в заголовке Eclipse, используйте дополнительный параметр -showLocation.

#### 3.2. Perspective

Перспектива является визуальным контейнером для набора видов и редакторов. Вы можете изменить компоновку элементов, используя перспективу (close / open views, editors, change the size, change the position, etc.). Eclipse Позволяет переключаться на другую перспективу, используя меню Window->Open Perspective -> Other. Для Java-разработчиков, самой используемой обычно будет “Java Perspective”.

*Обычной проблемой является та, что когда вы закрываете какой-то вид, вы не знаете, как можно его переоткрыть. Для того, чтобы привести перспективу в первоначальное состояние, используйте команду меню “Window” -> “Reset Perspective”.*

#### 3.3. Views и Editors

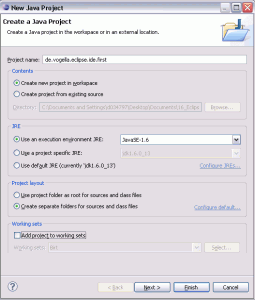
Вид как правило используется для навигации по иерархической информации или для открытия редактора. Изменения в виде, непосредственно применяются к привязанным к нему данным. Редакторы используются для модификации элементов. Редакторы могут иметь возможности автозавершения кода, отката и повтора, другие. Чтобы применить изменения в редакторе к данным, к которым он привязан (например, исходные коды Java-программы), как правило необходимо выполнить команду сохранения.

### 4. Создание вашего первого Java-приложения

Далее описывается, как создать минимальное Java-приложение, используя Eclipse. Это будет классическое “Hello World”. Наша программа выведет “Hello Eclipse!” в консоли.

#### 4.1. Создание проекта

Выберите в меню File -> New-> Java project. Введите “de.vogella.eclipse.ide.first” в качестве имени проекта. Также выберите “Create separate source and output folders”.

[](http://sotnyk.com/wp-content/uploads/2011/10/03-xfirstgany10.png)

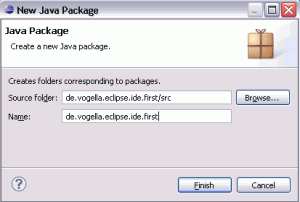
Нажмите finish для создания проекта. Новый проект создан и будет показан как папка. Откройте папку “de.vogella.eclipse.ide.first”

#### 4.2. Создание пакета

Теперь создайте пакет. Хорошей практикой будет использование для пакета того же самого имени, что и для проекта. Соответственно, создаем пакет с именем “de.vogella.eclipse.ide.first”.

Выберите папку src, кликните по ней правой кнопкой мыши и выберите New -> Package.

[http://sotnyk.com/wp-content/uploads/2011/10/04-xfirstgany30-300x61.png](http://sotnyk.com/wp-content/uploads/2011/10/04-xfirstgany30.png)

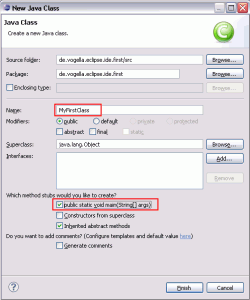
[](http://sotnyk.com/wp-content/uploads/2011/10/05-xfirstgany40.png)

#### 4.3. Создание Java класса

Кликните правой кнопкой на вашем пакете и выберите New -> Class

[](http://sotnyk.com/wp-content/uploads/2011/10/06-xfirstgany50.png)

Создайте MyFirstClass, пометьте опцию “public static void main (String[] args)”

[](http://sotnyk.com/wp-content/uploads/2011/10/07-xfirstgany60.png)

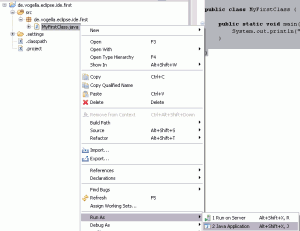
В открывшемся редакторе введем следующий код.

[?](http://sotnyk.com/2011/10/09/rukovodstvo-po-eclipse-ide/)

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9 | package de.vogella.eclipse.ide.first;    public class MyFirstClass {      public static void main(String[] args) {      System.out.println("Hello Eclipse!");    }    } |

#### 4.4. Запуск проекта в Eclipse

Теперь запустим полученный код. Для этого удобно кликнуть правой кнопкой над вашим Java-классом и выбрать Run-as-> Java application

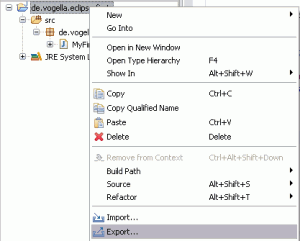
[](http://sotnyk.com/wp-content/uploads/2011/10/08-xfirstgany70.png)

Все готово! Сейчас вы можете увидеть вывод в консоли.

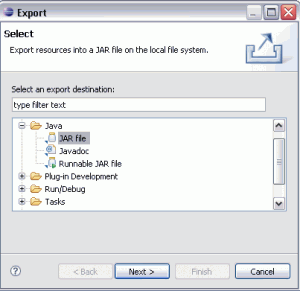
[](http://sotnyk.com/wp-content/uploads/2011/10/09-xfirstgany80.png)

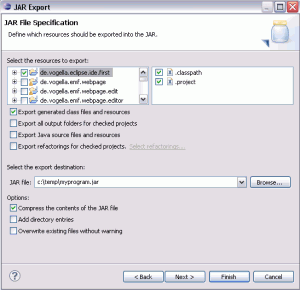
#### 4.5. Запуск Java-программы извне Eclipse (создаем jar-файл)

Для того, чтобы запустить вашу Java-программу без Eclipse, необходимо экспортировать её как jar-файл. Выберите проект, кликните его правой кнопкой и выберите “Export”.

[](http://sotnyk.com/wp-content/uploads/2011/10/10-xfirstgany90.png)

Выберите «JAR file», затем нажмите «next». Выберите ваш проект и введите куда его необходимо экспортировать, указав имя для jar-файла. Я назвал его “myprogram.jar”.

[](http://sotnyk.com/wp-content/uploads/2011/10/11-xfirstgany100.png)

[](http://sotnyk.com/wp-content/uploads/2011/10/12-xfirstgany110.png)

Жмите «Finish». Это запустит процесс создания jar-фала в выбранном каталоге.

#### 4.6. Запуск вашей программы не из-под Eclipse

Откройте командную оболочку, например, в Microsoft Windows выберите Start -> Run (или нажмите сочетание Win+R) и введите в этом диалоге «cmd». Это приведет к запуску консоли.  
Перейдите в каталог, в который вы сохранили jar. Например, если он расположен в папке “c:\temp”, наберите “cd c:\temp”.

Для запуска программы, необходимо включить jar-файл в classpath. Подробности вы можете посмотреть в разделах [Classpath](http://www.vogella.de/articles/JavaIntroduction/article.html" \l "classpath) и [Java JAR Files](http://www.vogella.de/articles/JavaIntroduction/article.html" \l "jarfiles).

java -classpath myprogram.jar de.vogella.eclipse.ide.first.MyFirstClass

[http://sotnyk.com/wp-content/uploads/2011/10/13--300x16.png](http://sotnyk.com/wp-content/uploads/2011/10/13-.png)

Поздравляю! Вы создали ваш первый Java-проект, крохотную Java-программу и запустили её из-под среды разработки Eclipse и вне нее.

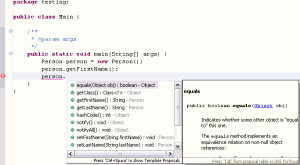
### 5. Content Assists и Quick Fix

*Чтобы получить список наиболее важных клавиатурных сокращений Eclipse, пожалуйста, обратитесь к руководству [Eclipse Shortcuts](http://www.vogella.de/articles/EclipseShortcuts/article.html)*

#### 5.1. Content assist

Функция «content assistant» позволяет получить помощь по вводу контента прямо из редактора. Она может быть вызвана при помощи клавиатурного сокращения CTRL + Space.

Например, наберите «syso» и нажмите [Ctrl + Space]. Только что введенный текст будет заменен на «System.out.println(“”)». Или, если вы имеете объект, скажем Person P и вам нужно посмотреть методы данного объекта, вы можете набрать «p.» (или нажать CTRL + Space), что также активирует «content assist».

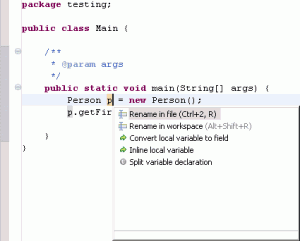
[](http://sotnyk.com/wp-content/uploads/2011/10/14-xcontentassists10.png)

#### 5.2. Quick Fix

Всякий раз, когда Eclipse находит проблему в коде, он подчеркивает проблемный участок. Выделите его и нажмите (Ctrl+1).

Например, наберите “myBoolean = true;”. Если переменная myBoolean еще не определена, Eclipse подсветит её как ошибку. Выделите переменную и нажмите “Ctrl+1″ и Eclipse посоветует создать поле или локальную переменную.

Функция Quick Fix очень интеллектуальна, она позволяет создавать новые локальные переменные, поля, методы, классы, обрамлять код с исключениями конструкциями try / catch, приводить выражение к типу переменной и т.п.

[](http://sotnyk.com/wp-content/uploads/2011/10/15-xquickfix10.png)

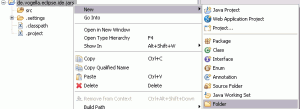
### 6. Использование jars (библиотек)

#### 6.1. Добавление внешних библиотек (.jar ) в Java classpath

Ниже идет описание, как можно добавить jars в ваш проект. Мы подразумеваем, что вы имеете необходимую jar-библиотеку.

*Если вам необходим пример работы с jar-библиотеками, вы можете использовать руководство [JFreeChart Tutorial](http://www.vogella.de/articles/JFreeChart/article.html)*

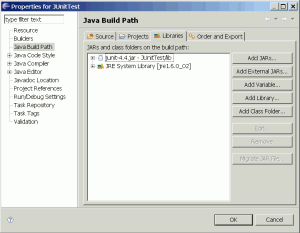
Создайте новый Java-проект “de.vogella.eclipse.ide.jars”. Создайте новую папку с именем “lib” (или используйте уже существующую папку) при помощи клика правой кнопкой на проекте и выбрав New -> Folder

[](http://sotnyk.com/wp-content/uploads/2011/10/16-xexternaljars02.png)

Выберите в меню File -> Import -> File system. Выберите нужный jar-файл и выберите папку «lib» в качестве целевой.

Далее, клик правой кнопкой на вашем проекте, выберите пункт «свойства» (properties). На закладке «libraries» выберите “Add JARs”.

Следующий пример показывает, как может выглядеть результат, если в проект добавить библиотеку «junit-4.4.jar».

[](http://sotnyk.com/wp-content/uploads/2011/10/17-xexternaljars10.png)

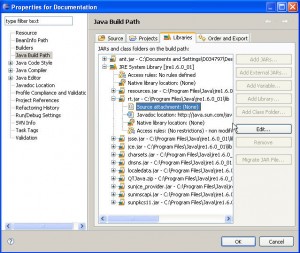
#### 6.2. Показ исходников jar-библиотеки

Чтобы просмотреть исходные коды типов, содержащихся в библиотеке, можно присоединить архив исходников или папку с исходниками к библиотеке. После этого редактор будет показывать исходный код вместо декомпилированного кода. Настройки присоединенных исходников позволяют настраивать уровень вхождения в них при отладке.

Диалог присоединения исходников может быть достигнут следующим путем:

Откройте страницу проекта «Java Build Path» (Projects > Properties > Java Build Path). На странице «Libraries», раскройте узел необходимой библиотеки и выберите атрибут «Source attachment». Нажмите кнопку «Edit».

Введите местонахождение исходников.

[](http://sotnyk.com/wp-content/uploads/2011/10/18-xadd_source_to_jar.jpg)

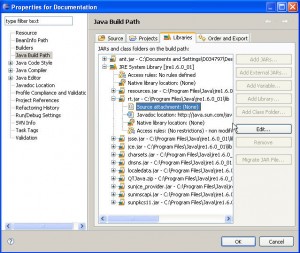
Для этого в поле «Location path» введите путь к архиву, либо к папке, содержащей исходные файлы.

#### 6.3. Добавление Javadoc к библиотеке jar

Загрузите javadoc для jar и поместите её где-то в файловой системе.

Откройте страницу проекта «Java Build Path» (Projects > Properties > Java Build Path). На закладке «Libraries page» раскройте узел библиотеки, выделите атрибут «Javadoc location» и нажмите кнопку «Edit».

Введите расположение фалов с описанием API.

[](http://sotnyk.com/wp-content/uploads/2011/10/19-xadd_source_to_jar.jpg)

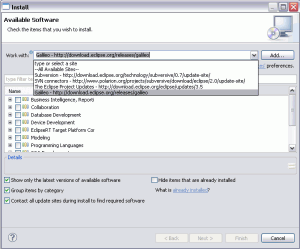
### 7. Обновление и установка плагинов

#### 7.1. Менеджер обновлений Eclipse

Eclipse предлагает функционал автоматического обновления (включая обновление плагинов). Eclipse 3.5 включает в себя «Software Update Manager», который позволяет обновить уже установленные плагины и инсталлировать новые.

Для того, чтобы обновить установленное, выберите в меню Help -> Check for Updates. Система проверит, имеются для установленных плагинов обновления или нет.

Для того, чтобы расширить функционал за счет новых плагинов, выберите пункт меню Help-> Install New Software.

[](http://sotnyk.com/wp-content/uploads/2011/10/20-xUpdateManager10.png)

Выберите из списка сайт, откуда вы собираетесь установить новые плагины. Например, для установки новых плагинов от Galileo, выберите «Galileo Update Site».

*Иногда необходимо снять выделение с пункта “Group items by category” – не все доступные плагины категоризированы. Соответственно, они не будут показаны. Подробности можно увидеть здесь – [Eclipse bug](https://bugs.eclipse.org/bugs/show_bug.cgi?id=281764).*

Для того, чтобы добавить новый сайт в список обновлений, нажмите кнопку “Add” и введите соответствующий URL. Данная операция позволит вам сделать данный сайт доступным и инсталлировать с него плагины.

#### 7.2. Ручная установка плагинов (папка dropins)

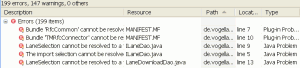
Для того, чтобы установить плагин, у которого нет соответствующего сайта обновления, вы можете использовать папку Dropins, расположенную внутри вашей инсталляции Eclipse.

Для этого, просто поместите соответствующий плагин в папку “dropins” и перезапустите Eclipse. Eclipse должен обнаружить новый плагин и установить его.

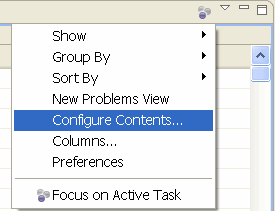
### 8. Дополнительные подсказки

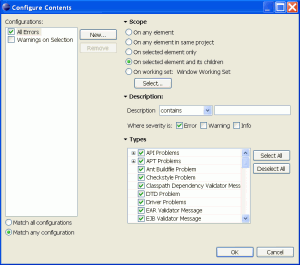
#### 8.1. Вид «Problems»

Вид «problems» показывает проблемы ваших проектов. Его можно открыть через пункт меню Windows -> Show View -> Problems

[](http://sotnyk.com/wp-content/uploads/2011/10/21-xproblems10.png)

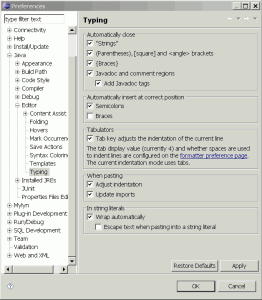
Вы можете настроить этот вид. Например, для того, чтобы настроить отображение проблем только выделенного проекта, выберите “Configure Contents”.

[](http://sotnyk.com/wp-content/uploads/2011/10/22-xproblems20.png)

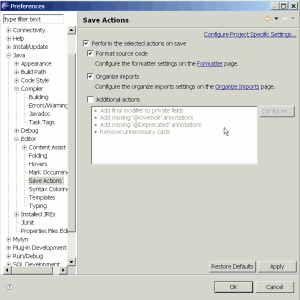
[](http://sotnyk.com/wp-content/uploads/2011/10/23-xproblems30.png)

#### 8.2. Важные настройки

Eclipse позволяет автоматически добавлять точку с запятой (и некоторые другие элементы).

[](http://sotnyk.com/wp-content/uploads/2011/10/24-xwayofworking10.png)

Также, Eclipse может автоматически форматировать исходный код и организовывать секцию импорта в момент записи на диск.

[](http://sotnyk.com/wp-content/uploads/2011/10/25-xwayofworking30.png)

*Вы можете экспортировать ваши настройки текущего рабочего пространства, используя File -> Export -> General -> Preferences. Аналогично, вы затем можете импортировать их обратно в это, либо другое активное рабочее пространство.*

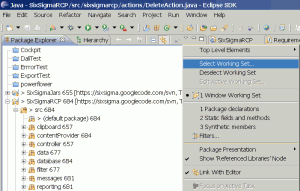
#### 8.3. Управление заданиями

Вы можете использовать конструкцию // TODO в процессе кодирования, чтобы указать на задание для Eclips. Впоследствии вы легко можете найти его, используя вид «task» в Eclipse.

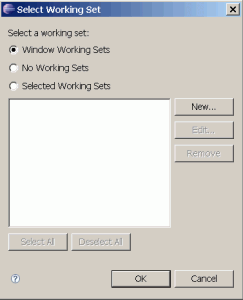
Для расширенных задач вы можете использовать руководство [Eclipse Mylyn Tutorial](http://www.vogella.de/articles/Mylyn/article.html).

#### 8.4. Рабочие множества

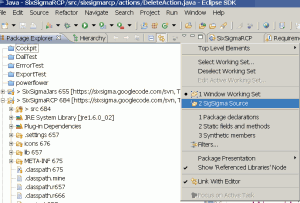
Обычной проблемой в Eclipse является то, что ваши данные в вашем рабочем пространстве растут и постепенно оно становится плохо структурированным. Для дополнительной организации используемых проектов и данных, вы можете использовать рабочие множества. Чтобы настроить ваше рабочее множество, выберите команду меню Package Explorer -> View Menu -> Select Working Sets. Меню вида находится в пределах виджета вида Package Explorer – небольшой треугольник рядом с командой сворачивания и разворачивания в правом верхнем углу.

[](http://sotnyk.com/wp-content/uploads/2011/10/26-xworkingset10.png)

В открывшемся диалоге нажмите «new» для создания рабочего множества.

[](http://sotnyk.com/wp-content/uploads/2011/10/27-xworkingset20.png)

В следующем диалоге выберите «java», выберите папку исходников, которую вы хотите видеть и дайте ей имя. Теперь вы можете легко видеть только те файлы, которые вам нужны в данный момент.

[](http://sotnyk.com/wp-content/uploads/2011/10/28-xworkingset30.png)

#### 8.5. Синхронизация отображения эксплорера пакетов с редактируемым кодом

Вид «package explorer» позволяет отображать связанные файлы из текущего редактора. Например, если вы работаете над файлом foo.java и смение в редакторе на bar.java, то отображение в эксплорере пакетов изменится.

Чтобы активировать данную возможность, нажмите “Link with Editor”.

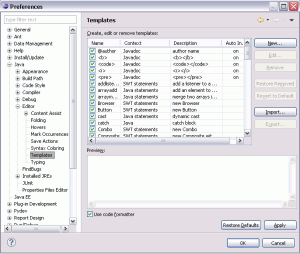
[http://sotnyk.com/wp-content/uploads/2011/10/29-xlinkwitheditor10-300x50.png](http://sotnyk.com/wp-content/uploads/2011/10/29-xlinkwitheditor10.png)

#### 8.6. Шаблоны кода

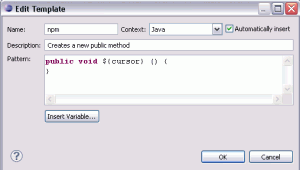
Для часто используемых кусков кода, частей документа, можно создать шаблоны, которые будут активироваться через функцию автодополнения (Ctrl + Space).

Например, представим себе, что вы часто создаете методы типа “public void name(){}”. Вы можете определить шаблон, который создаст тело метода.

Для создания такого шаблона, выберите пункт меню Window->Preferences and Open Java -> Editor -> Templates

[](http://sotnyk.com/wp-content/uploads/2011/10/30-xtemplates10.png)

Нажмите «New». Создайте следующий шаблон. Макрос «${cursor}» указывает, что курсор должен быть расположен в данной позиции после применения шаблона.

[](http://sotnyk.com/wp-content/uploads/2011/10/31-xtemplates30.png)

Данный пример помечен ключевым словом “npm”.

Теперь каждый раз, когда мы будем вводить в редакторе Java данное ключевое слово и нажимать Ctrl+Space, оно будет заменяться текстом соответствующего шаблона.

[http://sotnyk.com/wp-content/uploads/2011/10/32-.png](http://sotnyk.com/wp-content/uploads/2011/10/32-.png)

### 9. Дальнейшие шаги

Для того, чтобы больше узнать, как отлаживать Java-программы в Eclipse, вы можете изучить руководство [Eclipse Debugging](http://www.vogella.de/articles/EclipseDebugging/article.html)  
Чтобы более подробно изучить интернет-разработку на Java, вы можете использовать руководство [Servlet and JSP development](http://www.vogella.de/articles/EclipseWTP/article.html). Если вы хотите разработать клиентское Java-приложение, используя Eclipse, вам может помочь руководство [Eclipse RCP](http://www.vogella.de/articles/RichClientPlatform/article.html). Также вы можете расширить функциональность Eclipse, используя руководство[Eclipse Plugins](http://www.vogella.de/articles/EclipsePlugIn/article.html).

Удачи вам в дальнейшем изучении Java!

### 10. Спасибо вам

В оригинальной статье здесь находилось предложение поддержать автора материально. [Вы можете это сделать, перейдя на оригинальную статью](http://www.vogella.de/articles/Eclipse/article.html#thankyou).

### 11. Вопросы и обсуждения

Прежде, чем послать вопрос, пожалуйста, просмотрите [vogella FAQ](http://www.vogella.de/faq.html). Если вы имеете вопросы, либо нашли ошибки в данной статье, пожалуйста, используйте[www.vogella.de Google Group](http://groups.google.com/group/vogella). Я создал небольшой перечень [how to create good questions](http://www.vogella.de/blog/2010/03/09/asking-community-questions/), который также может помочь вам правильно сформулировать вопрос.

### 12. Ссылки и литература

#### 12.1. Исходные коды

[Source Code of Examples](http://www.vogella.de/code.html)

12.2. Ресурсы Eclipse  
[Eclipse.org Homepage](http://www.eclipse.org/)

12.3. vogella Resources  
[Eclipse RCP Training](http://www.vogella.de/training/eclipsercp.html) (Немецкий) Тренинг по Eclipse RCP с Ларсом Фогелем  
[Android Tutorial](http://www.vogella.de/articles/Android/article.html) Введение в программирование на Андроид  
[GWT Tutorial](http://www.vogella.de/articles/GWT/article.html) Программа на Java и компиляция в JavaScript и HTML  
[Eclipse RCP Tutorial](http://www.vogella.de/articles/EclipseRCP/article.html) Создание «родных» приложений на Java  
[JUnit Tutorial](http://www.vogella.de/articles/JUnit/article.html) Тестирование ваших приложений  
[Git Tutorial](http://www.vogella.de/articles/Git/article.html) Поместите все, что у вас есть в распределенную систему контроля версий Git

This entry was posted in [Android](http://sotnyk.com/category/it/android/), [IT](http://sotnyk.com/category/it/), [Переводы](http://sotnyk.com/category/perevodyi/) and tagged [Android](http://sotnyk.com/tag/android/), [Eclipse](http://sotnyk.com/tag/eclipse/), [Java](http://sotnyk.com/tag/java/), [Ларс Фогель](http://sotnyk.com/tag/lars-fogel/), [перевод](http://sotnyk.com/tag/perevod/). Bookmark the [permalink](http://sotnyk.com/2011/10/09/rukovodstvo-po-eclipse-ide/" \o "Permalink to Руководство по Eclipse IDE).

Pingback: [*Урок 3. Начало работы в Eclipse IDE | Java для начинающих | Изучаем java программирование*](http://study-java.ru/uroki-java/urok-3-znakomstvo-s-eclipse/)