



[Home](#) [Книги и статьи](#) [О себе](#) [рлюм кзгзз](#)

← Когда технологии помогают

Чем бы себя занять... →

## Руководство по Eclipse IDE

Posted on [October 9, 2011](#) by [Serge](#)



Данный пост является переводом с английского [руководства от Ларса Фогеля](#). 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](#) 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

### Search

 Search

### Categories

- IT
  - Android
  - C#
  - Тесты
- Личное
- Остальное
- Переводы
- Советы

### Calendar

#### October 2011

M	T	W	T	F	S	S
					1	2
3	4	5	6	7	8	9
10	11	12	13	14	15	16
17	18	19	20	21	22	23
24	25	26	27	28	29	30
31						

« Sep

Nov »

### Recent Comments

- Serge on [Speaking speedometer – SSPEED](#)
- Serge on [Немного об экологии](#)
- Diana on [Немного об экологии](#)
- Ollroz on [Руководство по Eclipse IDE](#)
- ma-Gavet on [Speaking speedometer – SSPEED](#)

### Tags

[Александр Алексеевич](#)  
[Анатолий Александрович](#)  
[Карпаты Ларс Фогель](#) [Лигун Питер](#)  
[Норвиг Себастиан Трун](#) [Случаи из практики Шумейко](#)  
[Яндекс.Диск](#) [генетический алгоритм](#) [желтые воды](#)  
[конференция](#) [лицензии](#)  
[лохотроны](#) [лямбды](#)  
[патенты](#) [перевод](#)  
[рецензия](#) [стенфорд](#)  
[тест производительности](#)  
[трансфер-фактор](#) [шучу](#)  
[конечно](#) [.net framework](#)  
[ai-class](#) [Andrew Ng](#)  
[Android](#) [android](#)

- 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](#) или же как платформа для разработки приложений [Eclipse RCP](#) (Rich Client Platform). Здесь мы рассмотрим только первый вариант использования Eclipse – как интегрированной среды разработки для Java.

## 2. Начало работы

### 2.1. Установка

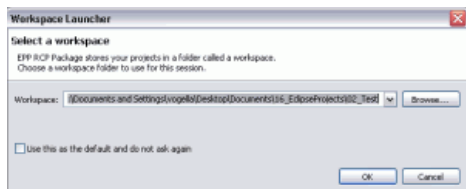
Eclipse требует установленную requires [Java](#) Runtime. Я рекомендую использовать Java 6 (также известную как Java 1.6).

Загрузите “Eclipse IDE for Java Developers” с сайта [Eclipse 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.

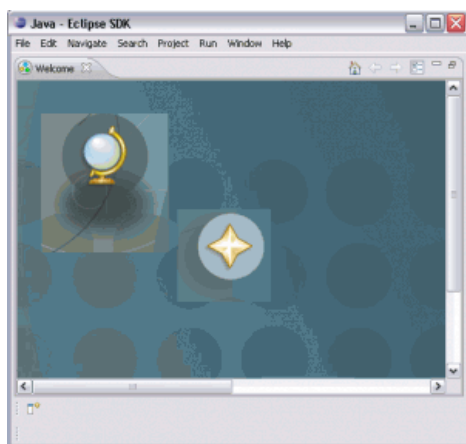
### 2.2. Запуск Eclipse

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.



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



### 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

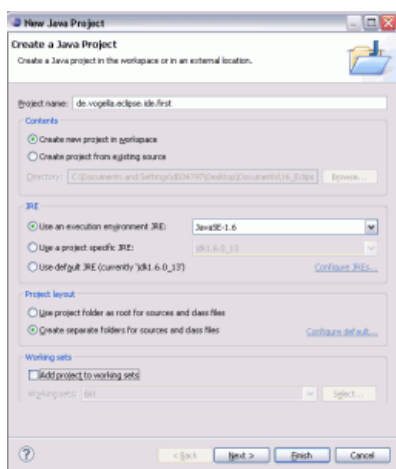
Вид как правило используется для навигации по иерархической информации или для открытия редактора. Изменения в виде, непосредственно применяются к привязанным к нему данным. Редакторы используются для модификации элементов. Редакторы могут иметь возможности автозавершения кода, отката и повтора, другие. 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”.

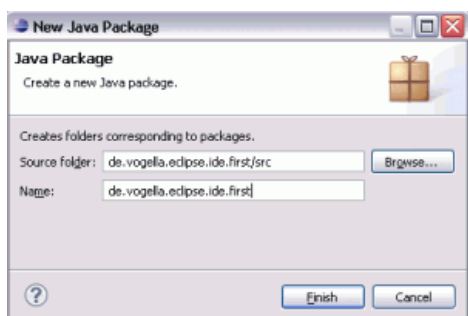


Нажмите finish для создания проекта. Новый проект создан и будет показан как папка. Откройте папку “de.vogella.eclipse.ide.first”

### 4.2. Создание пакета

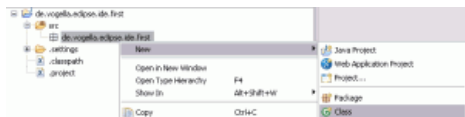
Теперь создайте пакет. Хорошей практикой будет использование для пакета того же самого имени, что и для проекта. Соответственно, создаем пакет с именем “de.vogella.eclipse.ide.first”.

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

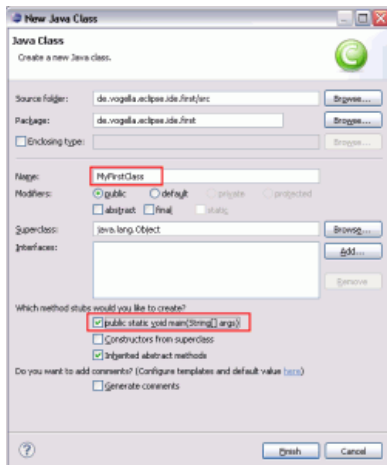


#### 4.3. Создание Java класса

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



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

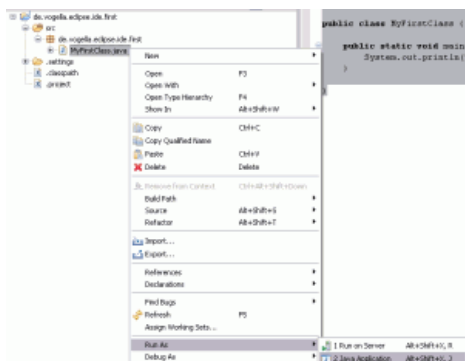


В открывшемся редакторе введем следующий код.

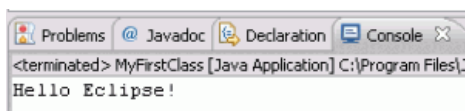
```
1 package de.vogella.eclipse.ide.first;
2
3 public class MyFirstClass {
4
5     public static void main(String[] args) {
6         System.out.println("Hello Eclipse!");
7     }
8
9 }
```

#### 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



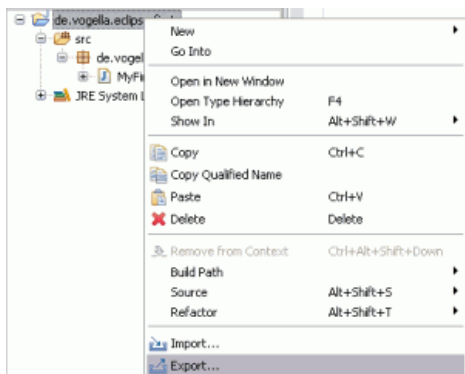
Все готово! Сейчас вы можете увидеть вывод в консоли.



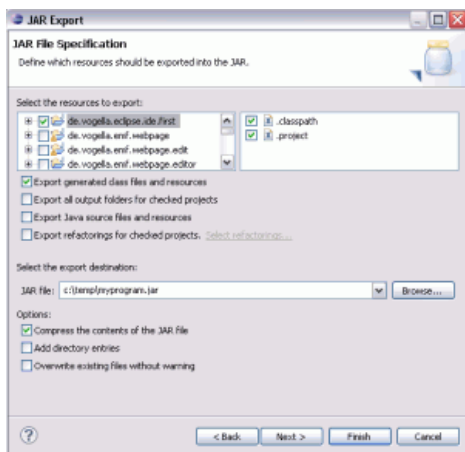
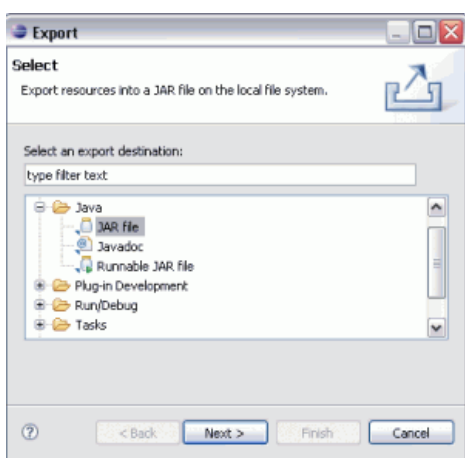
#### 4.5. Запуск Java-программы извне Eclipse (создаем jar-файл)

Для того, чтобы запустить вашу Java-программу без Eclipse, необходимо

экспортировать её как jar-файл. Выберите проект, кликните его правой кнопкой и выберите “Export”.



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”.



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](#) и [Java JAR Files](#).

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

```
cd /tmp/java && java -classpath myprogram.jar de.vogella.eclipse.ide.first.MyFirstClass
```

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

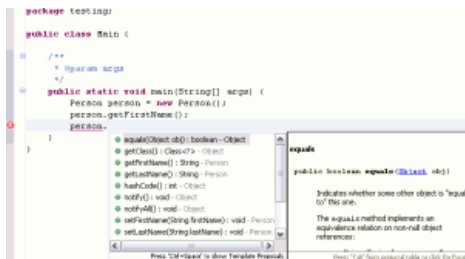
## 5. Content Assists и Quick Fix

*Чтобы получить список наиболее важных клавиатурных сокращений Eclipse, пожалуйста, обратитесь к руководству [Eclipse Shortcuts](#)*

### 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».

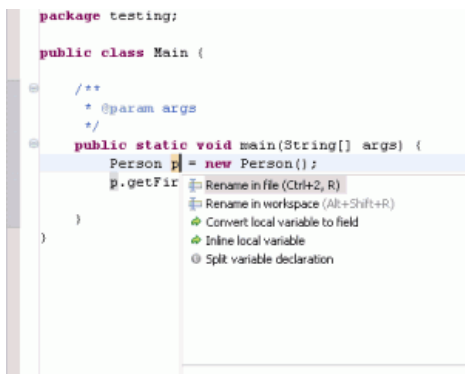


### 5.2. Quick Fix

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.



## 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](#)*

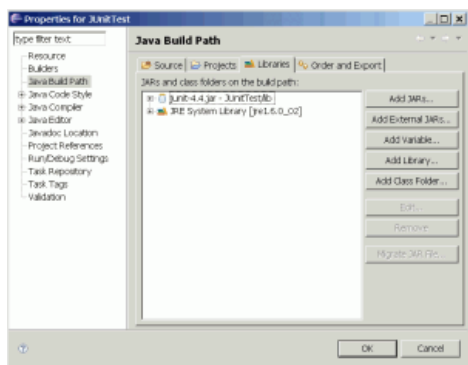
Создайте новый Java-проект “de.vogella.eclipse.ide.jar”. Создайте новую папку с именем “lib” (или используйте уже существующую папку) при помощи клика правой кнопкой на проекте и выбрав New -> Folder



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».



## 6.2. Показ исходников jar-библиотеки

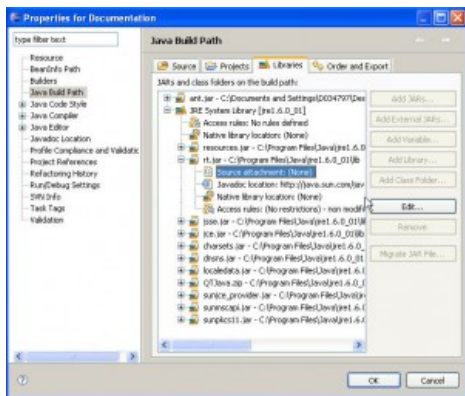
Чтобы просмотреть исходные коды типов, содержащихся в библиотеке, можно присоединить архив исходников или папку с исходниками к библиотеке. После этого редактор будет показывать исходный код вместо декомпилированного кода. Настройки присоединенных исходников позволяют настраивать уровень вхождения в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





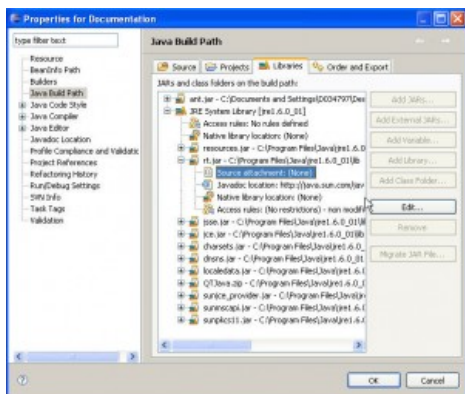
Для этого в поле «Location path» введите путь к архиву, либо к папке, содержащей исходные файлы.

### 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.



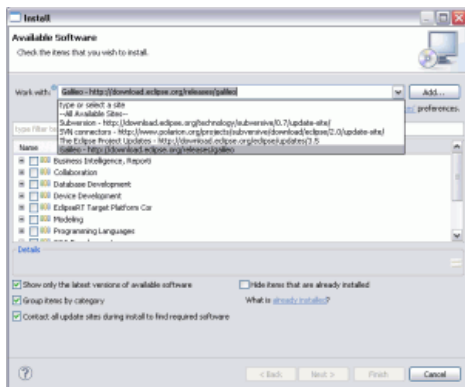
## 7. Обновление и установка плагинов

### 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 установить новые.

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

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



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](#).*

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 установить 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.

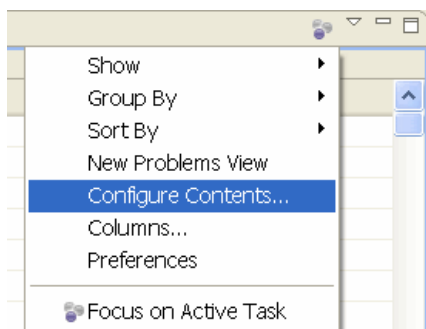
## 8. Дополнительные подсказки

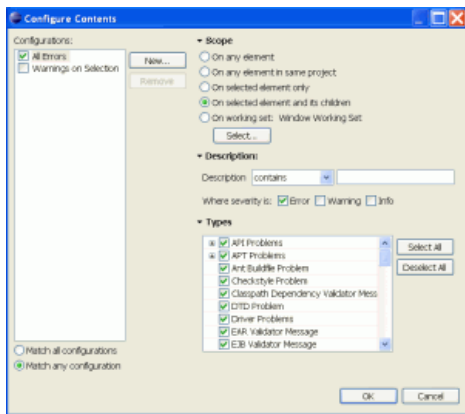
### 8.1. Вид «Problems»

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



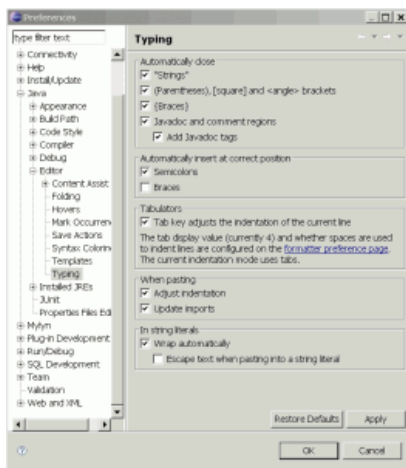
Вы можете настроить этот вид. Например, для того, чтобы настроить отображение проблем только выделенного проекта, выберите “Configure Contents”.



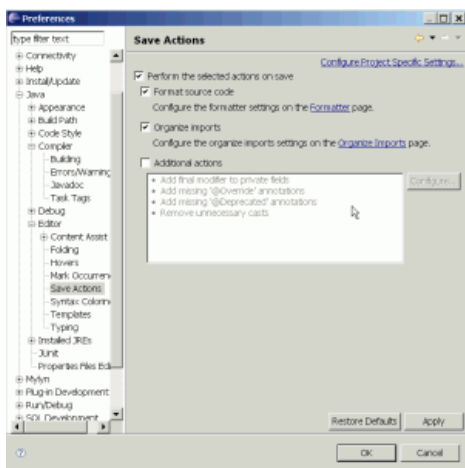


## 8.2. Важные настройки

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



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



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.

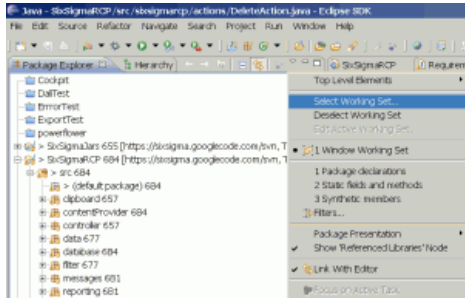
## 8.3. Управление заданиями

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

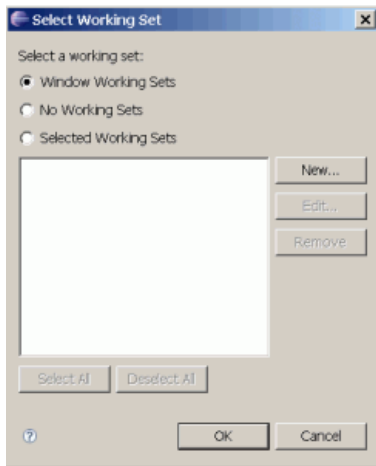
Для расширенных задач вы можете использовать руководство [Eclipse Mylyn Tutorial](#).

## 8.4. Рабочие множества

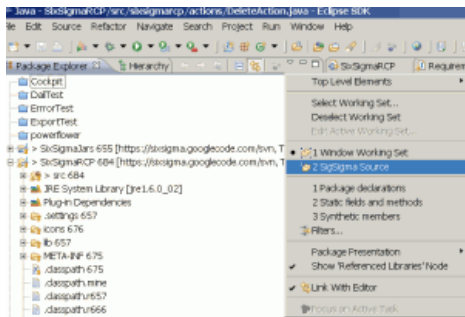
Обычной проблемой в Eclipse является то, что ваши данные в вашем рабочем пространстве растут и постепенно оно становится плохо структурированным. Для дополнительной организации используемых проектов и данных, вы можете использовать рабочие множества. Чтобы настроить ваше рабочее множество, выберите команду 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.



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



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.



## 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 смените в редакторе на bar.java, то отображение в эксплорере пакетов изменится.

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

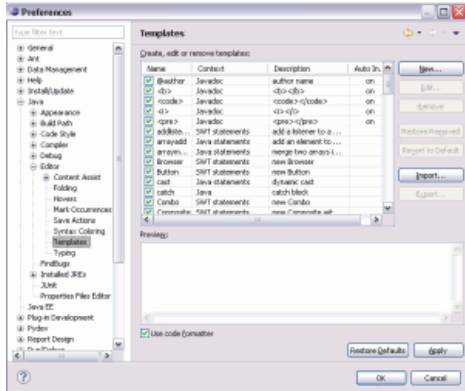


## 8.6. Шаблоны кода

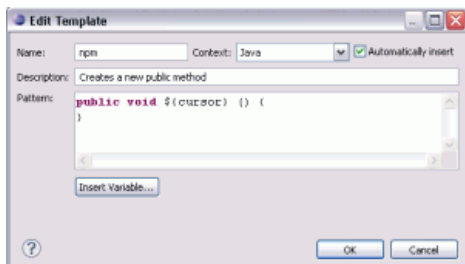
Для часто используемых кусков кода, частей документа, можно создать шаблоны, которые будут активироваться через функцию автодополнения (Ctrl + Space).

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

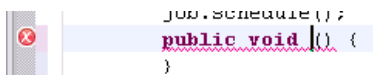


Нажмите «New». Создайте следующий шаблон. Макрос «`${cursor}`» указывает, что курсор должен быть расположен в данной позиции после применения шаблона.



Данный пример помечен ключевым словом “nprn”.

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



## 9. Дальнейшие шаги

Для того, чтобы больше узнать, как отлаживать Java-программы в Eclipse, вы можете изучить руководство [Eclipse Debugging](#)

Чтобы более подробно изучить интернет-разработку на Java, вы можете использовать руководство [Servlet and JSP development](#). Если вы хотите разработать клиентское Java-приложение, используя Eclipse, вам может помочь руководство [Eclipse RCP](#). Также вы можете расширить функциональность Eclipse, используя руководство [Eclipse Plugins](#).

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.](#)

## 11. Вопросы и обсуждения

Прежде, чем послать вопрос, пожалуйста, просмотрите [vogella FAQ](#). Если вы имеете вопросы, либо нашли ошибки в данной статье, пожалуйста, используйте [www.vogella.de Google Group](#). Я создал небольшой перечень [how to create good questions](#), который также может помочь вам правильно сформулировать вопрос.

## 12. Ссылки и литература

### 12.1. Исходные коды

[Source Code of Examples](#)

### 12.2. Ресурсы Eclipse

[Eclipse.org Homepage](#)

### 12.3. vogella Resources

[Eclipse RCP Training](#) (Немецкий) Тренинг по Eclipse RCP с Ларсом Фогелем

[Android Tutorial](#) Введение в программирование на Андроид

[GWT Tutorial](#) Программа на Java и компиляция в JavaScript и HTML

[Eclipse RCP Tutorial](#) Создание «родных» приложений на Java

[JUnit Tutorial](#) Тестирование ваших приложений

[Git Tutorial](#) Поместите все, что у вас есть в распределенную систему контроля версий Git

This entry was posted in [Android](#), [IT](#), [Переводы](#) and tagged [Android](#), [Eclipse](#), [Java](#), [Ларс Фогель](#), [перевод](#). Bookmark the [permalink](#).

← Когда технологии помогают

Чем бы себя занять... →

## 18 Responses to *Руководство по Eclipse IDE*



**Сергей** says:

November 1, 2011 at 11:10 pm

Прежде, спасибо добрый человек за труд и тем что поделился! Я пенсионер, решил с нуля освоить программирование на языке java, так как это мультиплатформенная среда и есть бесплатная программа Eclipse IDE для написания и компиляции. Ужас! Сейчас пишу это, а себе говорю “ну куда ты, ведь ты вообще ноль по этой теме”. Не тех.образования ни опыта. Правда скомпилировал как все начинающие “Hello World” и миниатюрную вычислялку “мой счет”. А вот этот перевод надеюсь дальше продвинет меня. Еще раз спасибо.

[Reply](#)



**Александр** says:

February 27, 2012 at 2:16 pm

Сделали бы версию zip для скачивания людям, что в сети раз в две недели, а дома (вне дествия связи) читают, что скачать успели. Есть и такие.

[Reply](#)



**baduzzzer** says:

May 21, 2012 at 9:39 pm

mmm, те кто не имеет постоянного доступа в интернет можно просто нажать на файл-> сохранить страницу как... . Сам всю школу только так и читал большинство ресурсов, сохранял страницы в школе, а дома уже просматривал и изучал их.

[Reply](#)

---



**Evgeny Knoll** says:

June 10, 2012 at 10:39 am

Сергей, огромное спасибо за Ваш труд! Очень полезная статья!

[Reply](#)

---



**Lex** says:

June 15, 2012 at 12:28 am

Огромное тебе, человеческое спасибо. Просто неоценимо полезная и интересная статья.

[Reply](#)

---



**Noob Max** says:

July 19, 2012 at 3:34 pm

Очень полезная статейка, пасиб за труд.

[Reply](#)

---



**Николай** says:

September 2, 2012 at 11:45 am

Долго пришлось искать руководство по Eclipse на русском языке!

[Reply](#)

---



**Vlada** says:

September 20, 2012 at 5:47 pm

Спасибо, наконец-то удалось во всем разобраться.

[Reply](#)

---



**Dasha** says:

September 23, 2012 at 5:49 pm

Спасибо за статью)))Наконец-то разобралась что к чему)

[Reply](#)

---



**Andrey** says:

July 23, 2013 at 11:53 am

Здравствуйте, классная статья, но не получилась одна вещь. Когда в командной строке прописал java -classpath ... У меня выскочил error: Could



not find or load main class ....

Что делать?

[Reply](#)



**Fill** says:

August 8, 2013 at 2:54 pm

Имею маленькую программку \*.APK. Хотелось бы узнать простейший путь изменения нескольких строчек её исходного кода.  
Убедительная просьба, не пишите “почитай здесь <http://www...>” С радостью бы никого не напрягал и прочитал бы и разобрался бы сам, но, видимо Господь обделил меня возможностью понять хотя бы азы программирования, а нуждаюсь я сильно в данной программе.  
Установил Eclipse, Android SDK, все загрузил/установил, но что делать дальше, хоть убейте, не могу додуматься (облазил уйму форумов).  
Если у кого-нибудь будет желание/возможность, пожалуйста, объясните всё по порядку и очень подробно, от А до Я, как для обезьяны, буду очень признателен.  
Спасибо

[Reply](#)



**daniil** says:

December 7, 2013 at 5:38 pm

я чуть изменил программу и у меня в место  
`package de.vogella.eclipse.ide.first;`

```
public class MyFirstClass {
```

```
    public static void main(String[] args) {  
        System.out.println("Hello Eclipse!");  
    }
```

```
}
```

было

```
package de.vogella.eclipse.ide.first;
```

```
public class MyFirstClass{  
    public static void main (String args[]){  
        int i = 999;  
        int r = 9999;  
        int k = 99;  
        i = count (i);  
        r = count (r);  
        k = count (k);  
        System.out.println(i);  
        System.out.println(r);  
        System.out.println(k);  
    }
```

```
    private static int count(int x){  
        x = x + 1;  
        return x;  
    }  
}
```

и программа выводила мне числа:1000;10000;100.

P.S.я просто похвастался 😊

[Reply](#)

---

Pingback: [Урок 3. Начало работы в Eclipse IDE | Java для начинающих | Изучаем java программирование](#)

---

Pingback: [android gamedev #2. Создание игры для андроид. Что внутри игры. Структура проекта | Стартап - как стиль жизни](#)



**tip** says:

December 11, 2014 at 5:20 am

вот автор перевёл статью и даёт ссылки на сайты на английском языке. чем он думал?

[Reply](#)



**Serge** says:

December 25, 2014 at 4:52 pm

Переводить остальные англоязычные сайты? 😊

[Reply](#)



**Владимир** says:

May 5, 2015 at 6:21 pm

Здравствуйте ! Создал проект по примеру , что у Вас на странице <http://sotnyk.com/2011/10/09/rukovodstvo-po-eclipse-ide/> public class

```
MyFirstClass {  
public static void main(String[] args) {  
System.out.println("Hello Eclipse!");  
}  
}
```

Подскажите , пожалуйста , как при следующем запуске программной среды Java Eclipse открыть этот проект ? Через какую закладку это делается.

Пробовал через File не получается. А по существу мне желательно иметь возможность открыть любой исходник , необходимо для учебы. Пробовал сделать это разными способами не получается. Читал об этом на сайтах , но ничего не помогло ! В Вашей статье программная среда очень похожа на мою .

[Reply](#)



**Ollroz** says:

July 2, 2015 at 9:07 am

Большое спасибо за Ваш труд! Такие статьи нужны как воздух для людей не занимающихся программированием только в одной среде, а «прыгающим» с одного языка на другой

[Reply](#)

---

## Leave a Reply

Your email address will not be published. Required fields are marked \*

Name \*

Email \*

Website

Comment

Post Comment

Spam Protection by [WP-SpamFree](#)

---

#### Meta

- [Log in](#)
- [Entries RSS](#)
- [Comments RSS](#)
- [WordPress.org](#)

**Сергей Сотник**

*Proudly powered by [WordPress](#).*