

CONTENTS INCLUDE:

- Getting Eclipse
- Workbench 101
- Development with Eclipse
- Keyboard Shortcuts
- Plug-ins
- Community Web Sites
- Hot Tips and more...

Getting Started with Eclipse

By Ed Burnette & Adam Houghton

WHAT IS ECLIPSE?

Eclipse is the leading Integrated Development Environment (IDE) for Java, with a rich ecosystem of plug-ins and an open source framework that supports other languages and projects. You'll find this reference card useful for getting started with Eclipse and exploring the breadth of its features.

We rundown the Eclipse distributions and configuration options, then guide you through Views, Editors, and Perspectives in Workbench 101. We list the top shortcuts and toolbar actions for everyday development. And, we provide a guide to the best places for finding plug-ins and getting involved with the Eclipse community.

We focus on the Windows and Mac OS X versions, but Eclipse runs on any modern operating system. Each Eclipse release is tested and validated on different versions of Windows, Linux, OS X, Solaris, and AIX.



OS Friendly

Upgrade to Vista? Eclipse 3.3 runs great on 32-bit versions of Microsoft's latest operating system and uses native WPF components. Eclipse 3.4 adds support for 64-bit Windows XP and Vista.

Mac user? Eclipse for OS X is a Universal Binary, so it natively supports both Intel and PowerPC Macs.

Eclipse is the most well known of several dozen open source projects hosted at eclipse.org (<http://www.eclipse.org>). Since 2001, the Eclipse SDK has been downloaded over 50 million times.

Most people think of Eclipse as a Java IDE but it's also one of the most popular tools for developing programs in Python, PHP, Ruby, C/C++, and other languages. You can even use it for non-programming tasks such as document creation and order entry. It achieves this flexibility through its modular plug-in architecture (more on that later).



Clean Install

Never install a new version of Eclipse on top of an older version. Rename the old one first to move it out of the way, and let the new version be unpacked in a clean directory.

GETTING ECLIPSE

Go to the eclipse.org download site—<http://download.eclipse.org/eclipse/downloads>—and choose the package that's right for you:

Package	Major Features
Eclipse IDE for Java Developers	Java IDE with incremental compilation, cross-referencing, code-assist, and Mylyn task management.
Eclipse IDE for Java EE Developers	Adds JEE validation, app server support, graphical HTML/JSP/JSF editing, and database tools.
Eclipse IDE for C/C++ Developers	C/C++ IDE with syntax highlighting and code completion, launcher, debugger, and makefile generator.
Eclipse for RCP/Plug-In Developers	Java IDE plus the Plug-In Development Environment for creating Eclipse plug-ins and applications.
Eclipse Classic	The original Java IDE and Rich Client Platform.

What version should I get?

At any given time up to five different build types are available. To see these, select **All versions** from the download page or visit <http://download.eclipse.org/eclipse/downloads>

Version	Frequency	Stability	Audience
Releases	Yearly	Best	Everyone
Maintenance Builds	Quarterly	Best	Everyone
Stable/Milestone Builds	6 Weeks	Good	Users interested in the latest features
Integration Builds	1 Week	Fair	Contributors to Eclipse
Nightly Builds	1 Day	Poor	Contributors to Eclipse

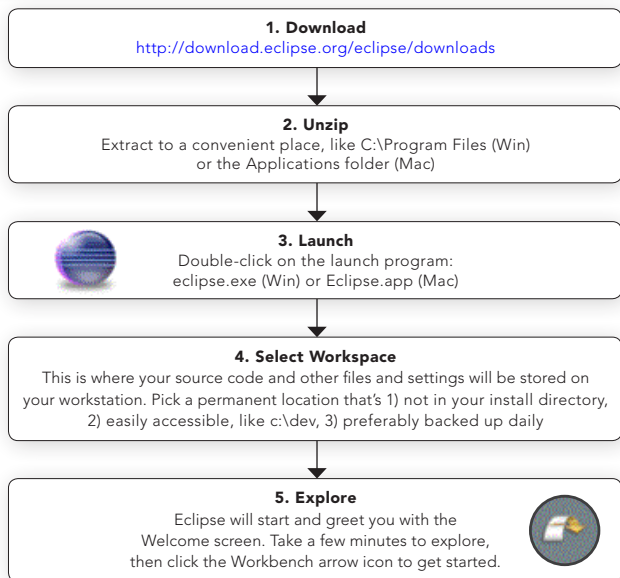


Get More Refcardz (They're free!)

- Authoritative content
- Designed for developers
- Written by top experts
- Latest tools & technologies
- Hot tips & examples
- Bonus content online
- New issue every 1-2 weeks

Subscribe Now for FREE!
Refcardz.com

GETTING STARTED



Messed Up Workspace?



If your Eclipse installation has startup errors or a corrupted configuration, it might be time to get a fresh start. Start Eclipse with the `-clean` option, and all cached framework and runtime data will be cleared out. This often helps fix plug-in issues and improve general stability.

THIRD-PARTY DISTRIBUTORS

There are many third-party distributions of Eclipse available. Some are free and just repackage the open source Eclipse code. Others are commercial, building on top of Eclipse by adding unique features and support. The most popular ones are listed here.

Distributions	Resource	Price
Yoxos on demand: Ajax based interface lets you pick and choose exactly what functionality you need.	http://yoxos.com/ondemand	Free
Easy Eclipse: Prepackaged releases of Eclipse targeted for specific types of developers.	http://www.easyeclipse.org	Free
Aptana Studio: A popular IDE for HTML/CSS/JavaScript development, with plug-ins for Adobe AIR, Ruby on Rails (RadRails), and PHP development. Built on Eclipse.	http://www.apтана.com	Free
PHPEclipse: An IDE for PHP development built on Eclipse.	http://www.phpeclipse.net	Free
PyDEV: An Eclipse-based environment for developing in Python and Jython.	http://pydev.sourceforge.net	Free
MyEclipseIDE: Inexpensive end-to-end Java and J2EE development environment.	http://www.myeclipseide.com	\$
BEA Workshop: End-to-end browser-to-database solution built on Eclipse.	http://workshopstudio.bea.com	\$\$\$
IBM RAD: Helps developers to quickly create Web, SOA, Java, J2EE and portal applications.	http://www-306.ibm.com/software/awdtools/developer/application	\$\$\$\$\$



Up the Memory

The default memory size of Eclipse is set in the `eclipse.ini` file. If you get Out of Memory errors or sluggish response, you may have to increase the defaults. In particular, if you get an error about "PermGen" memory (permanent generation), add this line at the end and restart Eclipse: `-XX:MaxPermSize=256m`

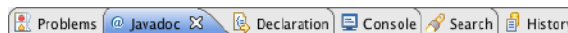
Use the lowest memory settings that work and perform well for your mix of projects.

On Windows and Linux, `eclipse.ini` resides in the Eclipse installation directory. On Mac OS X, it's located inside the application bundle: `Eclipse.app/Contents/MacOS/eclipse.ini`

WORKBENCH 101

Eclipse's main window is called the Workbench window. Its menus and toolbars should be familiar with users of any desktop applications, but some other elements like views, editors, and perspectives may not be.

Views



A view is a window that lets you examine something, such as a list of files in your projects. Eclipse comes with dozens of different views. Some open automatically when you use certain perspectives, but you can use the **Window > Show View** menu to open them at any time. Here's a partial list of views included in the Eclipse SDK:

View Name	Description
Breakpoints	Shows a list of all the breakpoints you have set in your projects
Console	Displays the output of your program.
Debug	Lists all programs that were launched by Eclipse. Use it to pause program execution, view tracebacks, and locate the cause of deadlocks.
Declaration	Shows the source code where the selected object is declared.
Display	Shows expression results in an unstructured format.
Error Log	View internal errors and stack dumps when reporting Eclipse problems.
Expressions	Shows a list of expressions and their values in the debugger.
Hierarchy	Displays the class and interface relationships for the selected object.
History	Shows the CVS change history of the file.
Javadoc	Shows the description (from comments) of the selected object.
JUnit	Shows the progress and results of JUnit tests.
Navigator	Shows all projects in the workspace as they exist on disk.
Outline	Displays the structure of the currently open file.
Package Explorer	Shows all your projects, Java packages, and files.
Problems	Shows compiler errors and warnings in your code.
Search	Shows the results of any search operation.
Tasks	Lists all the markers placed in your source code.
Variables	Shows all the parameters and local variables in scope while debugging.

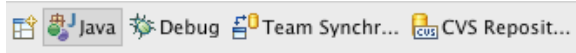
Multiple views can be stacked together in the same rectangular area. The title bar will show a tab for each view, but only one of them can be active at a time. If the window is too narrow to show all the tabs, a special menu will appear to give you access to the hidden ones.

Editors



Editors are very similar to views, but they are constrained to a single rectangular area in the Workbench window. Editors may be text-based (like the Java editor) or graphical (such as a diagram editor). You can save or revert an editor's contents. An asterisk in the editor's title bar indicates it has unsaved data.

Perspectives



A perspective is a set of views, editors, and toolbars, along with their arrangement within the Workbench window. Perspectives are not "modes". As you perform a task, you may rearrange windows, new views, and so on. Your arrangement is saved under the current perspective. The next time you have to perform the same kind of task, simply switch to that perspective and Eclipse will put everything back the way you left it.

For the most part you will be using the built-in Java or Debug perspectives, but you can also select **Window > Save Perspective As...** to save your current arrangement under a new name.

Toolbar

Five useful icons on the Eclipse toolbar:

	Debug As: debugs the last application you've launched.
	Run As: runs the last application you've launched.
	Toggle Mark Occurrences: highlighting an element will highlight all other occurrences of that element in the editor window.
	Last Edit Location: jumps to the file and location of the last edit you've made.
	Link with Editor: the Package Explorer will continually update to reflect the active file in the editor window.

Rearranging Views and Editors

To move a view or editor, click and drag its title bar. You can drag views outside of the Workbench window, or collapse them to an icon on the edge of the window (as a fast view). As you move your cursor it will change shape to indicate actions you can take or places you can drop things on.



Side by Side

By dragging editors, you can show two files side by side. You can also edit two portions of the same file by using the **Window > New Editor** command.

DEVELOPMENT WITH ECLIPSE

Eclipse can be used to create anything from a Java program to the next bestselling novel. Regardless of what you're creating, a few basic concepts remain the same...

Managing Projects

An Eclipse project is a directory that contains any source files, metadata, resources, and derived files. Typically projects are subdirectories of your workspace directory, but Eclipse allows them to be anywhere on the file system. You can either create a project from scratch or import an existing project from the file system, from CVS, or from an existing Ant build file.



Building

Where possible, Eclipse uses incremental compilation to build derived objects in the background. For example, when you save a Java file in Eclipse, there is no need to run a Build—it's already done. In Ant or Make based projects, this build is run after the Save and the results are shown in the Console view. You can turn automatic building on or off with an option: **Window > Preferences > General > Workspace > Build automatically**. If automatic builds are disabled then you can use **Project > Build All (Ctrl+B)** to start a build.



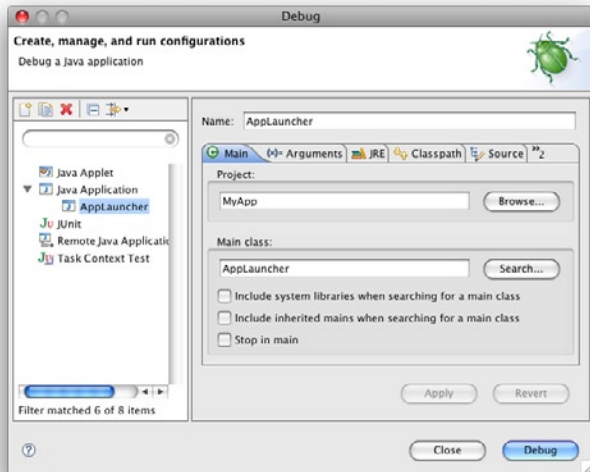
Blame Game

Want to track down who broke the build? Open the Annotate View with **Team > Show Annotation**, then click on a line of code to see who last changed it.

DEVELOPMENT WITH ECLIPSE, *continued*

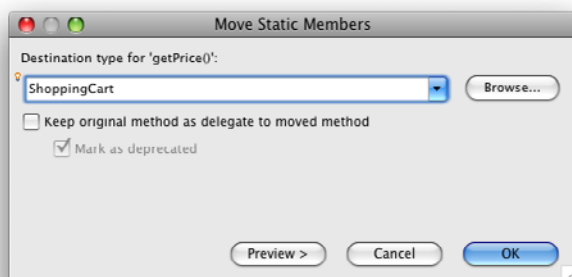
Running and Debugging

In Eclipse there is no overall "Run" command. Instead, you use one or more launch configurations, recipes describing how to run or debug your project. To create or modify a launch configuration, select **Run > Debug...** or **Run > Run...** depending on whether you want to run the program under the debugger or not. The Debug and Run icons on the toolbar provide a quicker way to do the same thing. Or, press **Ctrl+F11** to re-run the last program.



Refactoring

Eclipse has a powerful set of refactoring tools to help reshape your code. Rename variables, move classes between packages, and extract methods without fear of breaking the build. Highlight an element and press **Alt+Shift+T** to see the list of available refactorings.



Synchronizing

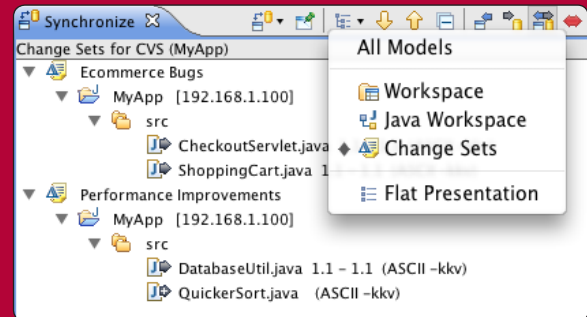
CVS synchronization is built into Eclipse. Right click on a resource and select **Team >...** to see options for synchronizing, committing, and updating. The Team Synchronizing Perspective provides helpful tools for keeping track of pushes and managing merges. The CVS Repository Exploring perspective lets you browse branches and check out projects from existing CVS repositories.

Not using CVS? There are Eclipse plug-ins for every type of version control system, including Subversion, Visual SourceSafe, Git, and Mercurial. Check out the Plug-ins section for the best places to find and download plug-ins.



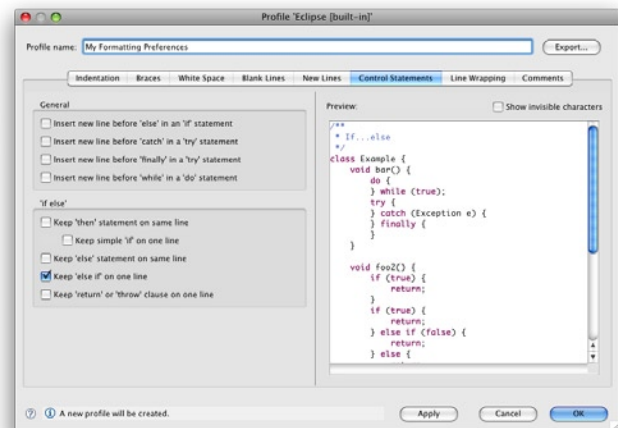
Juggling Code

Change Sets help you keep track of multiple commits in your playpen. Enable the Change Sets view, then select a group of files and right-click **Add To > New Change Set**.



Formatting

Control the minutiae of code formatting with Eclipse's plethora of Formatter options. Spend hours tweaking your settings in **Window > Preferences > Java > Code Style > Formatter**, then format all of your code with **Ctrl+Shift+F**. Save your formatting profile by selecting "Export" in the profile properties, then share it with your team members.



Automatic Clean Up

Forgetful about formatting? Set up Eclipse to automatically format source code and organize imports on every save. Select **Window > Preferences > Java Editor > Save Actions** to enable these actions. This dialog also lets you configure actions like removing unnecessary casts or adding missing annotations.

KEYBOARD SHORTCUTS

Here are our Top 50 favorite keyboard shortcuts defined by Eclipse. You can replace any of these (or even switch to Emacs bindings) by selecting **Window > Preferences > General > Keys**.

Debugging Shortcuts

Windows	OS X	Shortcut
F5	F5	Step Into
F6	F6	Step Over
F7	F7	Step Return
F8	F8	Resume
F11	F11	Debug Last Launched
Ctrl+F11	Cmd+F11	Run Last Launched
Ctrl+R	Cmd+R	Run to Line
Ctrl+Shift+B	Cmd+Shift+B	Toggle Breakpoint on Line

Navigation Shortcuts

Windows	OS X	Shortcut
F3	F3	Format Source
Ctrl+Shift+G	Cmd+Shift+G	Find Reference
Ctrl+Shift+R	Cmd+Shift+R	Open Resource
Ctrl+Shift+T	Cmd+Shift+T	Open Type
Ctrl+H	Cmd+H	Open Search Dialog
Ctrl+O	Cmd+O	Quick Outline
Ctrl+T	Cmd+T	Quick Hierarchy
Ctrl+L	Cmd+L	Go to Line
Alt+Left	Cmd+[Back
Alt+Right	Cmd+]	Forward
Ctrl+Q	Cmd+Q	Last Edit
Ctrl+.	Cmd+.	Next Error
Ctrl+,	Cmd+,	Previous Error

Coding Shortcuts

Windows	OS X	Shortcut
Ctrl+1	Cmd+1	Quick Fix
Ctrl+Space	Cmd+Space	Content Assist
F2	F2	Show Tooltip Description
Shift+F2	Shift+F2	Show External Javadoc
Ctrl+Shift+O	Cmd+Shift+O	Organize Imports
Ctrl+Shift+F	Cmd+Shift+F	Format Source
Ctrl+I	Cmd+I	Correct Indentation
Ctrl+/ Ctrl+F	Cmd+/ Cmd+F	Toggle Comment Find and Replace
Ctrl+D	Cmd+D	Delete Line
Ctrl+Alt+Up	Cmd+Option+Up	Copy Lines Up
Alt+Up	Option+Up	Move Lines Up
Ctrl+Alt+Down	Cmd+Option+Down	Copy Lines Down
Alt+Down	Option+Down	Move Lines Down

Java Refactoring Shortcuts

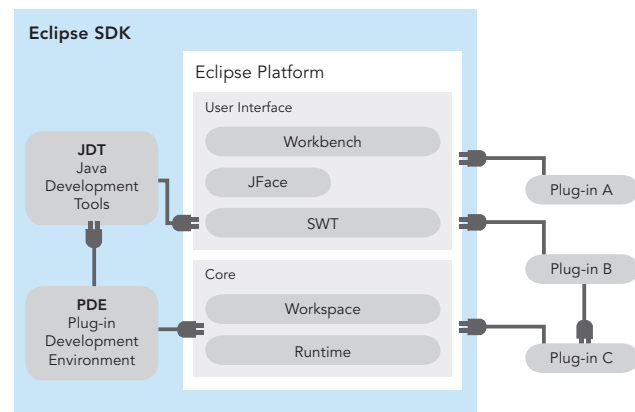
Windows	OS X	Shortcut
Alt+Shift+R	Cmd+Option+R	Rename Element
Alt+Shift+L	Cmd+Option+L	Extract to Local Variable
Alt+Shift+M	Cmd+Option+M	Extract to Method
Ctrl+Shift+Y	Cmd+Option+Y	Change Method Signature
Ctrl+Shift+Z	Cmd+Option+Z	Undo Refactoring

Editor Window Shortcuts

Windows	OS X	Shortcut
Ctrl+E	Cmd+E	Open Editor Window
Ctrl+M	Cmd+M	Maximize Editor Window
Ctrl+W	Cmd+W	Close Editor Window
Ctrl+Page Up	Cmd+Page Up	Switch to Tab Left
Ctrl+Page Down	Cmd+Page Down	Switch to Tab Right
Ctrl+F8	Cmd+F8	Change Perspective
Ctrl+Shift+S	Cmd+Shift+S	Save All
Ctrl+Shift+W	Cmd+Shift+W	Close All
F5	F5	Refresh

PLUG-INS

Aside from a small “bootstrapping” core runtime, Eclipse is made up entirely of plug-ins. Small collections of plug-ins are logically grouped into “features” that are installed and uninstalled as a unit. This figure shows how everything fits together:



The Eclipse SDK is pre-packaged with over 100 plug-ins including plug-ins for basic text editing, Java development, online help, and plug-in development. Literally thousands of other plug-ins have been written to add support for other languages, reporting, charting, database manipulation, J2EE, UML, GUI editing, RSS reading, and so forth.

There are several places on the web where you can find and install plug-ins. The main ones are:

Plug-in Resource	Description
Eclipse Downloads Home http://download.eclipse.org/eclipse/downloads	Select “By Project” or “By Topic” to access the plug-ins produced by all projects at eclipse.org . Most are downloaded as zip files that need to be manually unpacked like the Eclipse SDK.
Eclipse Plugin Central http://eclipseplugincentral.com	This site offers a plug-in directory, reviews, ratings, news, forums, and listings for products and services.
Yoxos Subscription Service http://www.yoxos.com	For a small yearly fee you can access the Yoxos Install Manager, which provides a convenient automatic discovery and update service for a consistent and tested set of plug-ins.
Source Forge http://sf.net	A large and growing number of Eclipse plug-ins are being developed in this open source nexus.
Pulse 2.0 Service http://www.poweredbypulse.com	Pulse 2.0 is a free service for discovering, downloading, and managing Eclipse plug-ins. Pulse allows sharing of profiles among multiple computers—great for bringing work home or showing tips to team members.

COMMUNITY WEB SITES

By installing Eclipse and reading this guide, you've already taken the first steps in joining the Eclipse community. To help you go further, web sites, articles, and other resources are available to assist you, as are thousands of Eclipse enthusiasts and professionals from around the world.

Community Web Sites	Description
Eclipse Zone http://eclipsezone.com	An online community for Eclipse users that includes exclusive articles, podcasts, and forums.
Eclipse Home Page http://www.eclipse.org	The Eclipse Foundation's home page.
Eclipse Newsgroups http://www.eclipse.org/newsgroups	Get peer support from thousands of other users and developers.

Community Web Sites	Description
Planet Eclipse http://planet.eclipse.org	Planet Eclipse is a window into the world, work, and lives of Eclipse users and contributors.
Eclipsepedia http://wiki.eclipse.org	The official eclipse.org wiki.
IBM AlphaWorks http://alphaworks.ibm.com/eclipse	Part of IBM's emerging technologies web site, this is dedicated to Eclipse and WebSphere-related projects and plug-ins.
IBM developerWorks http://www.ibm.com/developerworks/opensource	developerWorks hosts a variety of tutorials, articles, and related information on Eclipse and other open source projects.
Apache http://www.apache.org	Apache software is used throughout Eclipse, and the two projects collaborate in many areas.
O'Reilly Open Source http://opensource.oreilly.com	The O'Reilly Resource Center provides a broad range of references and links to publications about open source.

ABOUT THE AUTHORS



Ed Burnette

Ed Burnette has been hooked on computers ever since he laid eyes on a TRS-80 in the local Radio Shack. Since graduating from NC State University he has programmed everything from compilers and video games to web services and load testers. Ed is a Principal Systems Developer at the SAS Advanced Computing Lab in Cary, NC. In his copious spare time, Ed enjoys learning and writing about Open Source, Java, software development, and technology in general.

Publications: *Eclipse IDE Pocket Guide* (O'Reilly), *Eclipse in Action* (Manning)

Blog: Dev Connection blog at ZDNet: <http://blogs.zdnet.com/Burnette>



Adam Houghton

Adam Houghton's high tech career began at age 12, running a multi-line bulletin board system from the family living room. He later enjoyed riding the dot-com wave as a server-side Java developer. After moving to Durham, North Carolina, he settled down into the stable world of business software and is currently a Senior Software Developer in the SAS Advanced Computing Lab.

Articles: *iPhone Web Development with Eclipse* (IBM developerWorks), *Hands-On Google Web Toolkit* (Dr. Dobbs Journal)

Blog: www.adamhoughton.com, personal weblog

RECOMMENDED BOOK



Eclipse in Action provides a thorough guide to using Eclipse features and plugins effectively in the context of real-world Java development. Realistic examples demonstrate

how to use Eclipse effectively to build, test and debug applications using the tools provided by Eclipse and other third-party open source plugins.

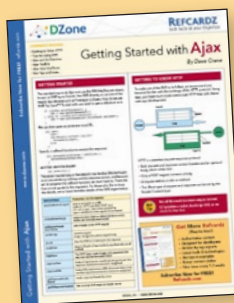
BUY NOW

books.dzone.com/books/eclipse-in-action

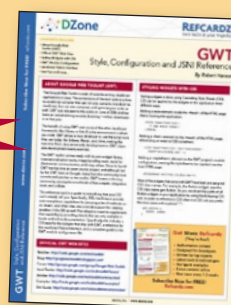
Want More? **Download Now.** Subscribe at refcardz.com

Upcoming Refcardz:

- Spring Configuration
- Windows PowerShell
- Dependency Injection in EJB3
- Flexible Rails: Flex 3 on Rails 2
- RSS and Atom
- jQuery Selectors
- Design Patterns
- MS Silverlight 2.0
- NetBeans IDE 6 Java Editor
- Groovy



Getting Started with Ajax



GWT Style, Configuration and JSNI Reference



The **DZone Network** is a group of free online services that aim to satisfy the information needs of software developers and architects. From news, blogs, tutorials, source code and more, DZone offers everything technology professionals need to succeed.

To quote *PC magazine*, "DZone is a developer's dream."

DZone, Inc.
1251 NW Maynard
Cary, NC 27513
888.678.0399
919.678.0300

Refcardz Feedback Welcome
refcardz@dzone.com

Sponsorship Opportunities
sales@dzone.com

ISBN-13: 978-1-934238-04-2
ISBN-10: 1-934238-04-X



9 781934 238042

50795
\$7.95