


ORACLE®



**ORACLE®**

**What's New in NetBeans IDE 7.2**

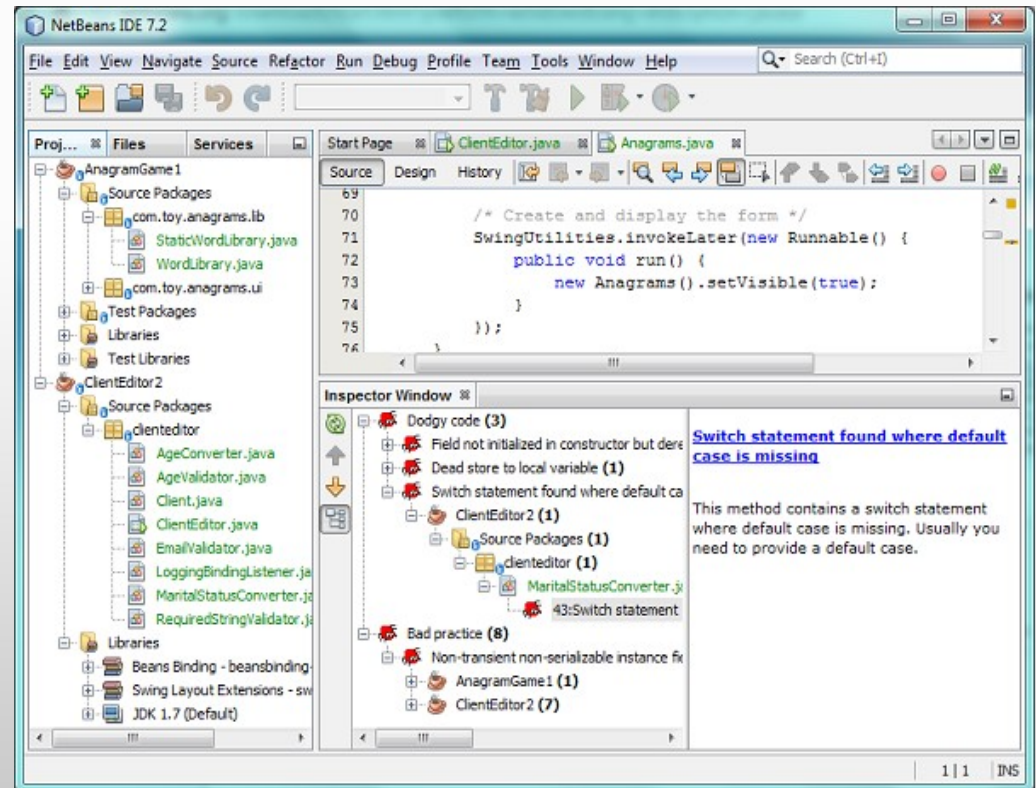


The following is intended to outline our general product direction. It is intended for information purposes only, and may not be incorporated into any contract. It is not a commitment to deliver any material, code, or functionality, and should not be relied upon in making purchasing decisions. The development, release, and timing of any features or functionality described for Oracle's products remains at the sole discretion of Oracle.

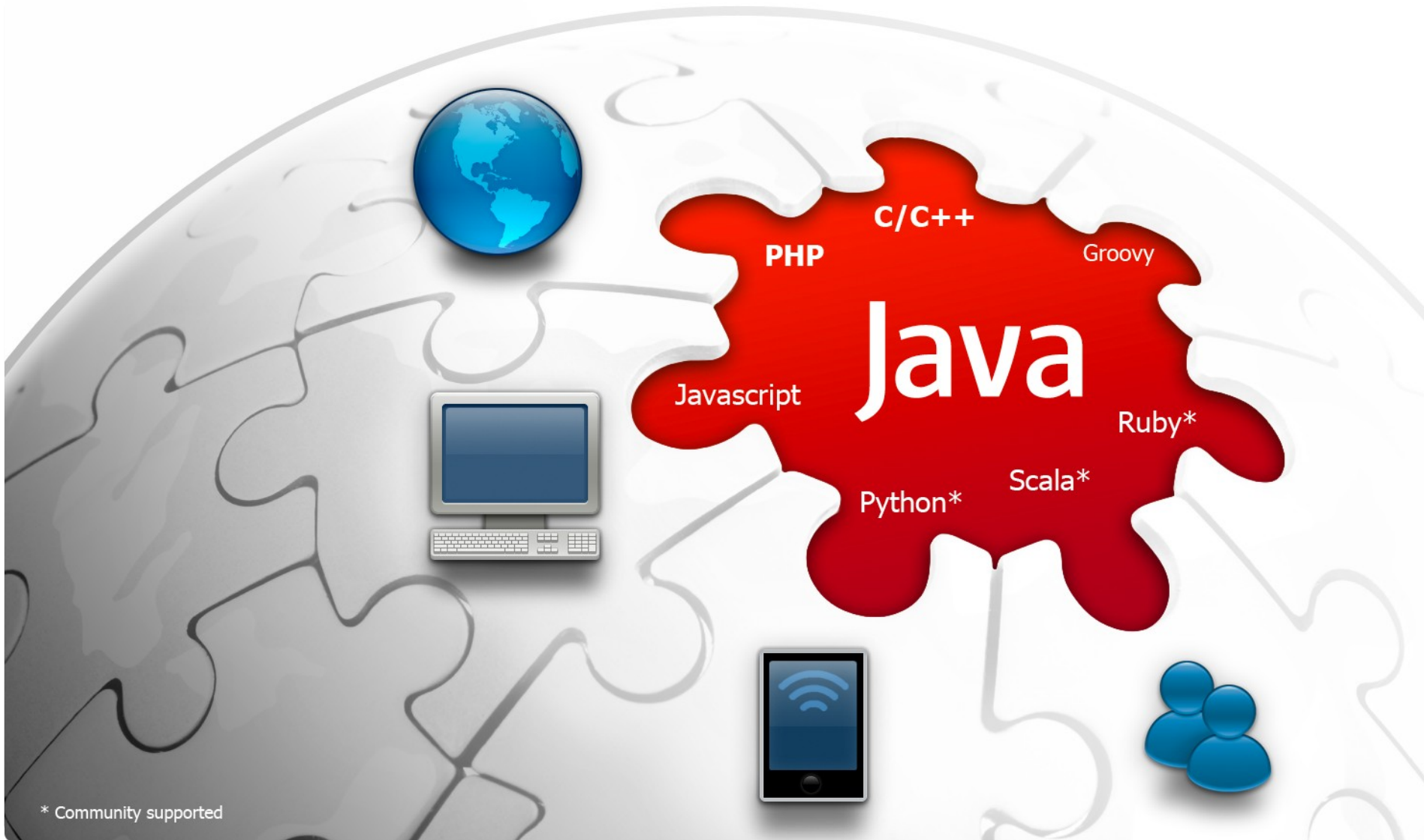
# What is NetBeans?



- A comprehensive, modular IDE
  - Ready to use out of the box
  - Support for latest Java specifications & standards
  - Other languages too (PHP, C/C++, etc)
  - Intuitive workflow
  - Debugger, Profiler, Refactoring, etc.
  - Binaries & ZIPs for Mac, Solaris, Linux, Windows
- Simply download and run on any OS!



# What is NetBeans?



# What is NetBeans?



## A Thriving Open Source Community



Growing list of 3rd party modules (plug-ins)

Over 20 Million Downloads



Active blogging community

Available in many languages



# Why use NetBeans?



## Increase Development Productivity

- Works Out-of-the-box
- Freely Available
- Open Source
- Support for Java Standards & Platforms
- Extensible Java Desktop Framework
- Powerful, Award Winning GUI Builder
- Profiling & Debugging Tools
- Ant & Maven Based Projects
- Customizable Projects





- Improved performance
- Be more productive when coding
- Easily analyze large code bases
- Seamless development with JavaFX Scene Builder
- New features & enhancements to existing features



- What was done in NetBeans IDE 7.2?
  - Significantly improved NetBeans startup time
  - Significantly improved scanning to sync your code with NetBeans
- Result
  - Up to 65% improvement in startup time at first startup
  - Up to 25% improvement in project scanning speed
- Example: SilverPeas.org  
Time to open very large project until ready-to-code:
  - 44 seconds in NetBeans IDE 7.2
  - 2 minutes and 37 seconds in NetBeans IDE 7.1.1

## NetBeans 7.2 beta: Faster and More Helpful

There has been significant excitement about the **beta release** of **NetBeans 7.2**. In this post, I look briefly at what makes this release so exciting (including better performance, providing more hints, and integrating FindBugs).

NetBeans 7.2 beta  
C/C++, and PHP b  
easy as ever and t

## NetBeans IDE 7.2: Smarter & Faster

05.23.2012



Email



Like

8



Tweet

41



+1

24



Share

8

Views: 16531

The NetBeans IDE 7.2 release, [with its Beta having made available today](#), comes with significantly improved performance and many new features in the Java Editor, with a special focus on new static code analysis capabilities. Along with a vastly improved developer experience that helps you write good, bug-free code quickly, it also delivers improvements in other areas, notably in its support for JavaFX, Java EE, Maven, PHP, and Groovy.

# Community Feedback



**Robert Basic** @robertbasic

1h

odd. #netbeans 7.2b feels much snappier/faster than 7.1.x

Expand



**Romain N** @romainneutron

31 May

@netbeans testing the 7.2 beta. Awesome work, no more freeze while "scanning", more responsive interface. Good job guys !



**Skye Book** @sbook

21m

Definitely feeling the speed improvements in **NetBeans** 7.2 today

Expand



**Faissal Boutaounte** @bfaissal

26 May

WOW #netbeans 7.2 open faster than #Firefox ;) . Great JOB

@netbeans team, thank you !

# Be More Productive When Coding

## Improved Java Editor

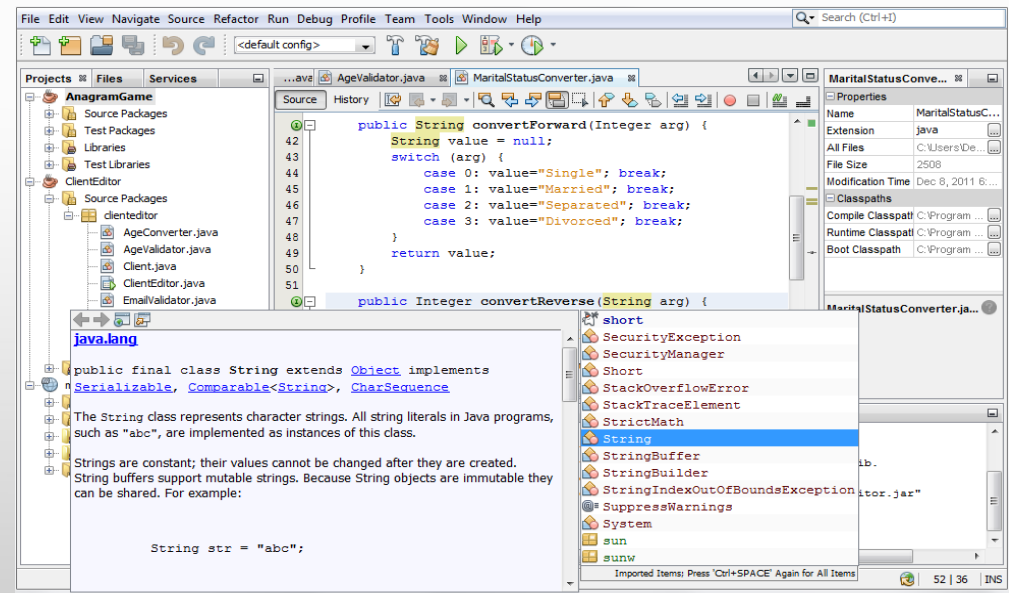


- The main feature of NetBeans IDE

- Developed in close co-operation with Java SE Team in Sun and in Oracle

- First editor to support Java 7 & JavaFX 2.0

- “javac” is our parser  
all language features  
always in sync with Java



# New in the NetBeans Java Editor

## Static Analysis Tools Integration



- FindBugs Integration

- Popular open source static analysis tool for Java

- Identifies bugs in many categories

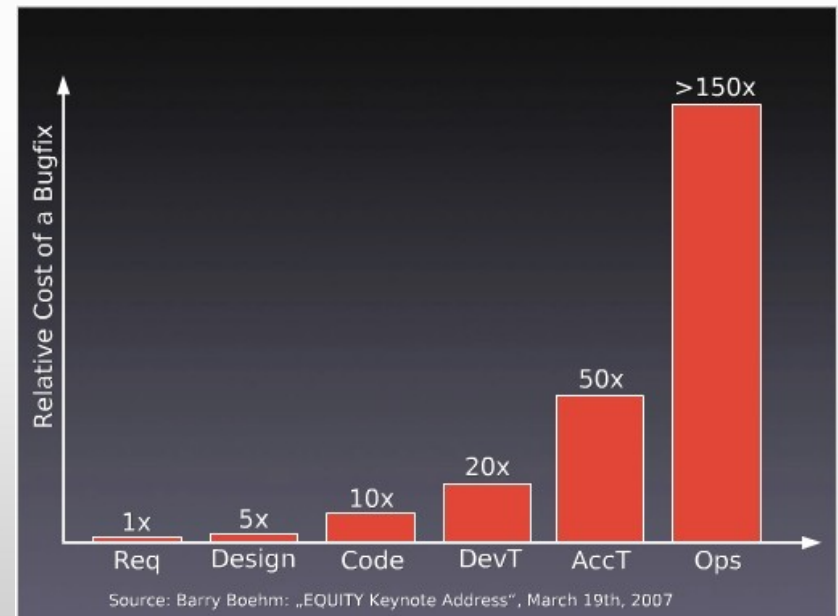
- **Bad Practice** — Eg: Method may fail to close stream
    - **Correctness** — Eg: equals method always returns false
    - **Internationalization** — Eg: Reliance on default encoding
    - **Malicious code vulnerability** — Eg: Expose internal rep by reference to mutable object
    - **Multithreaded correctness** — Eg: method calls `Thread.sleep()` with a lock held
    - **Performance** — Eg: Huge string constants duplicated across multiple class files
    - **Security** — Eg: Hardcoded constant database password
    - **“Dodgy code”** — Eg: Method ignores return value, is this OK?



# Why FindBugs? Cost of Bug Fixing



- The earlier the better
- The larger the software system, the more pronounced is the cost difference across the stages



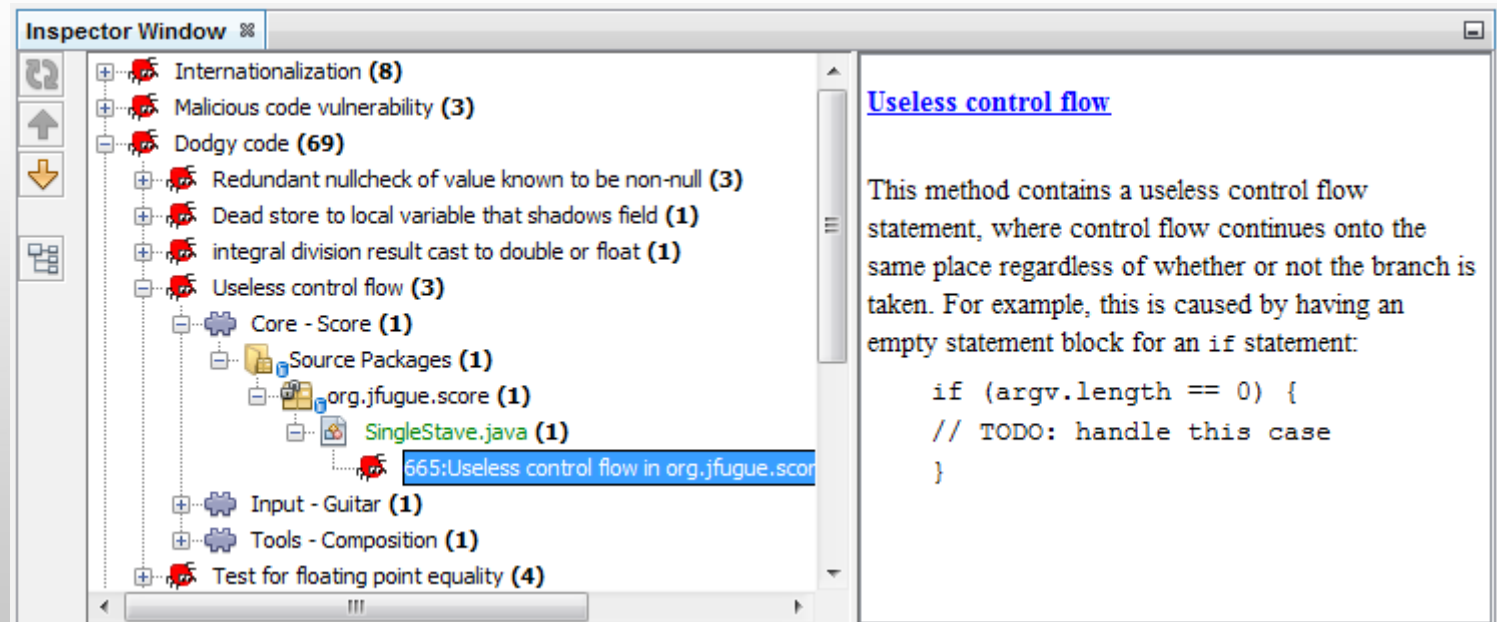
- Programming style
  - Helps in making it easier to spot bugs in code
  - Defensive Programming
- Programming techniques
  - Programed to self monitor or check consistency of internal data
- Development methodologies
- Programming language support
  - Eg. Type systems leading to compile time type checks
  - Decision to not support pointers in Java
- Code analysis
  - Inspect the code for errors beyond what the compiler catches



# FindBugs Integration

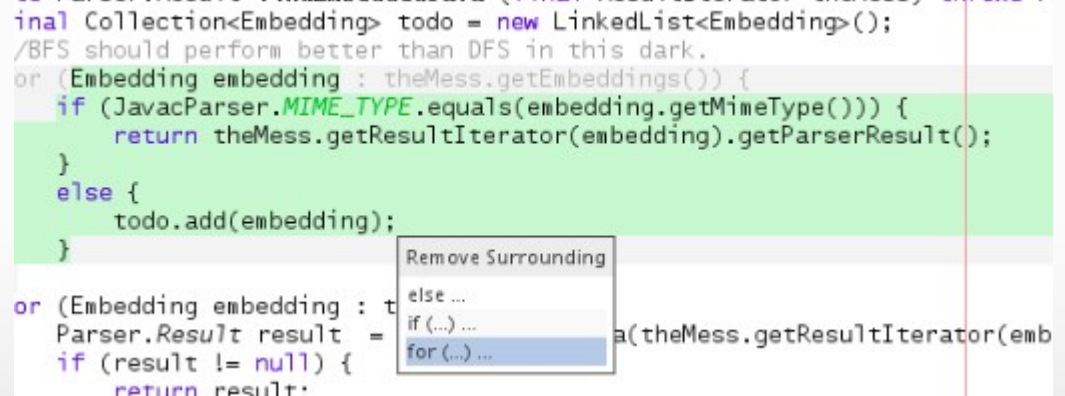


- Set scope of analysis on projects, packages, or files
- Examine results and jump into editor



- Remove Surrounding Code

- Quickly remove enclosing parts of nested statements
- Source > Remove Surrounding Code
- Alt-Backspace



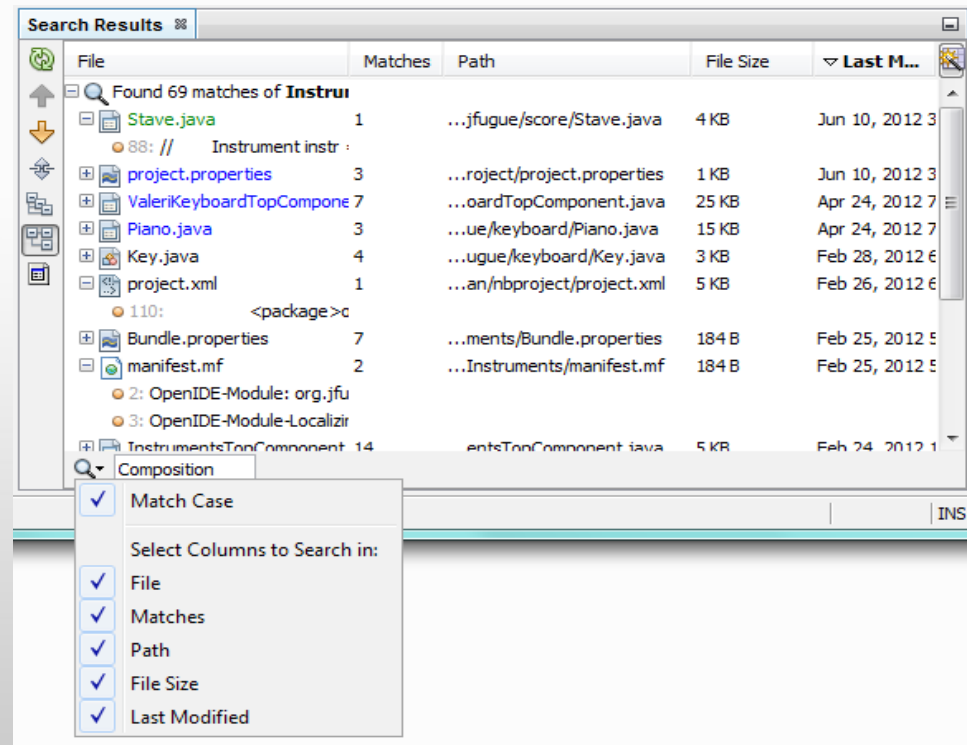
The screenshot shows a Java code snippet in the NetBeans editor. A green selection highlights a block of code within a larger method. A context menu is open over the selected code, with the 'Remove Surrounding' option at the top. Below it, three sub-options are visible: 'else ...', 'if (...) ...', and 'for (...) ...', with the 'for (...) ...' option currently selected. The code in the background includes a linked list, a comment about BFS vs DFS, and a loop processing embeddings.

```
final Collection<Embedding> todo = new LinkedList<Embedding>();  
//BFS should perform better than DFS in this dark.  
for (Embedding embedding : theMess.getEmbeddings()) {  
    if (JavacParser.MIME_TYPE.equals(embedding.getMimeType())) {  
        return theMess.getResultIterator(embedding).getParserResult();  
    }  
    else {  
        todo.add(embedding);  
    }  
}  
  
for (Embedding embedding : t  
Parser.Result result =  
if (result != null) {  
    return result;  
}
```

# New in the NetBeans Java Editor



- Reformat all code across all projects, packages, files
- Better Search Results
- New package view collapses empty packages
- Export projects to ZIP and import from ZIP
- Replace constructor with builder or factory
- Multi-tab editor and change position of tabs
- And more!

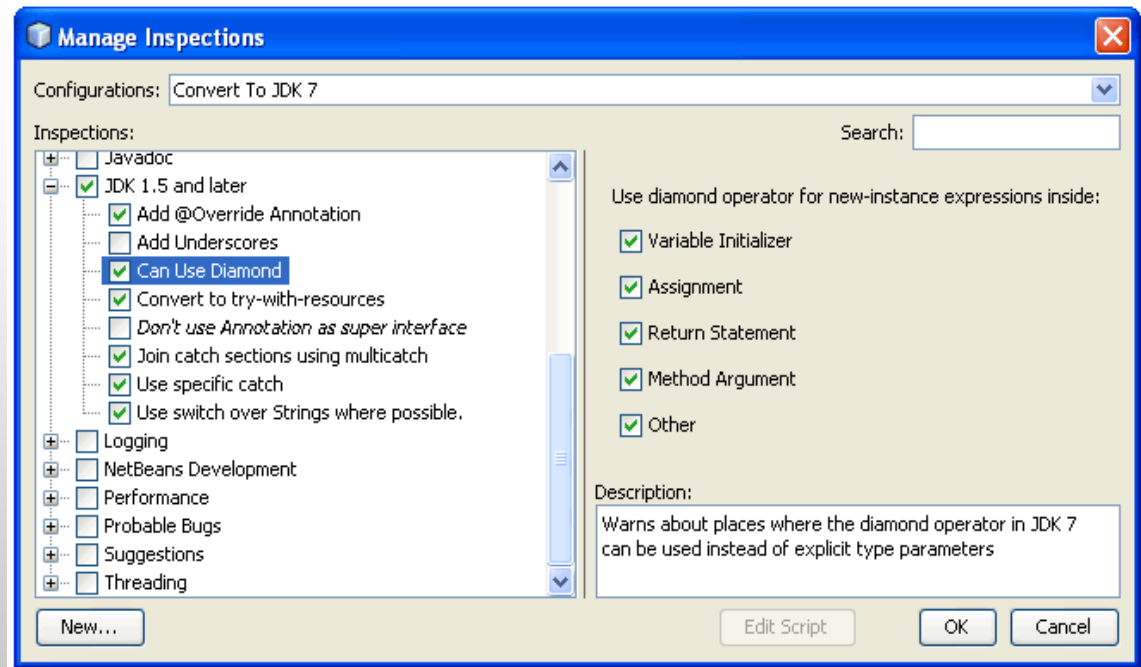


# NetBeans Java Editor

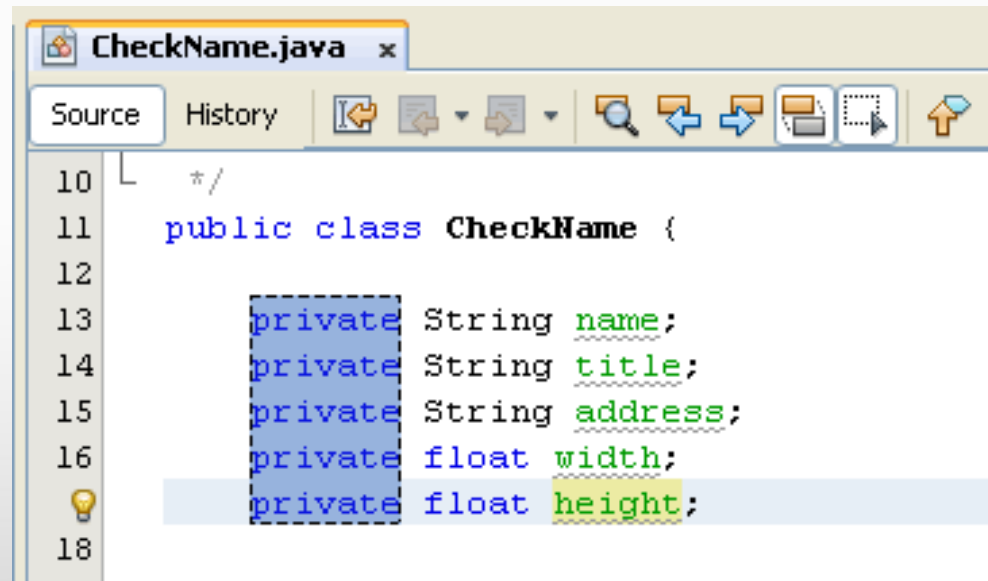
## Other Key Features



- Batch Refactoring
  - Fix problems across packages and projects
  - Upgrade projects completely to JDK 7
  - Customized scopes for refactoring
  - Expression language for defining rules
  - Introduced in 7.1



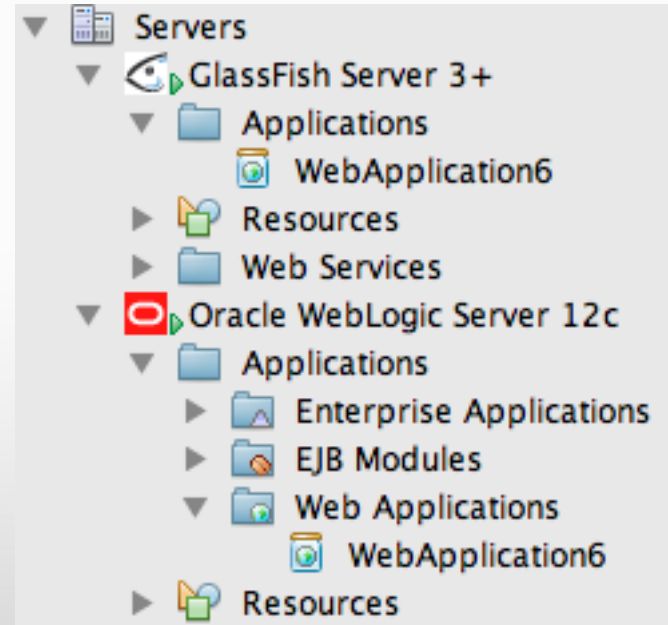
- Block Selection
  - One-click toggle button for selecting code blocks across multiple lines
  - Scenarios:
    - Delete rectangular block
    - In-place editing
    - Cut/Copy/Paste
    - Move up/down
  - Introduced in 7.1



# NetBeans Java EE Support



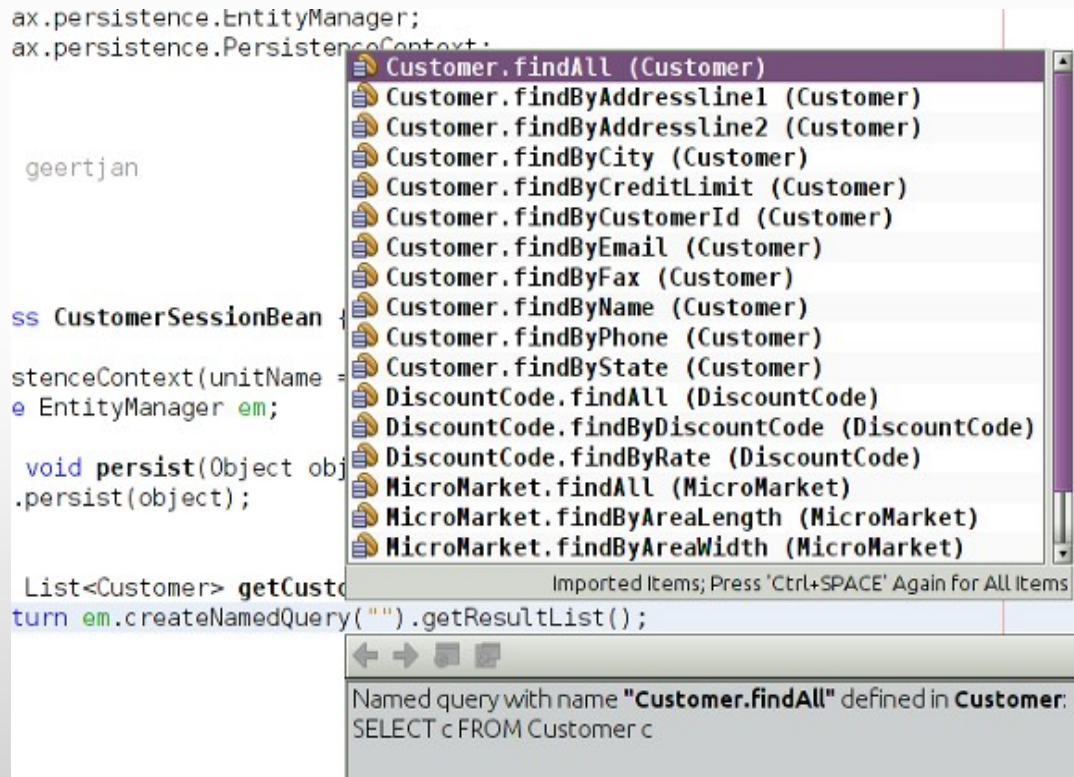
- Developed in co-operation with GlassFish Team and WebLogic Team
- Best-in-breed GlassFish integration, strong support for WebLogic and other servers
- Support for standards, e.g., Java EE 6
- Ready to use out of the box



# New in NetBeans Java EE Support



- JPA code completion in named queries and SQL queries
- PrimeFaces 3.2
- Spring 3.1.1

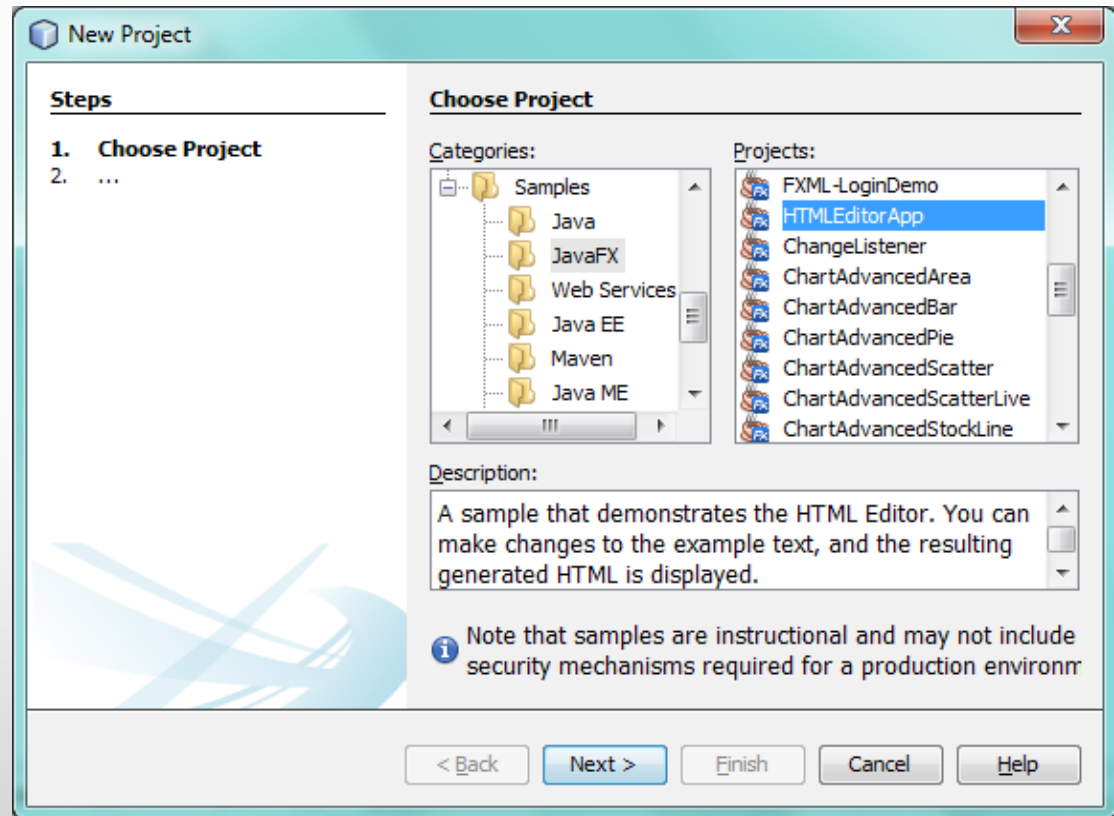




# NetBeans JavaFX Support



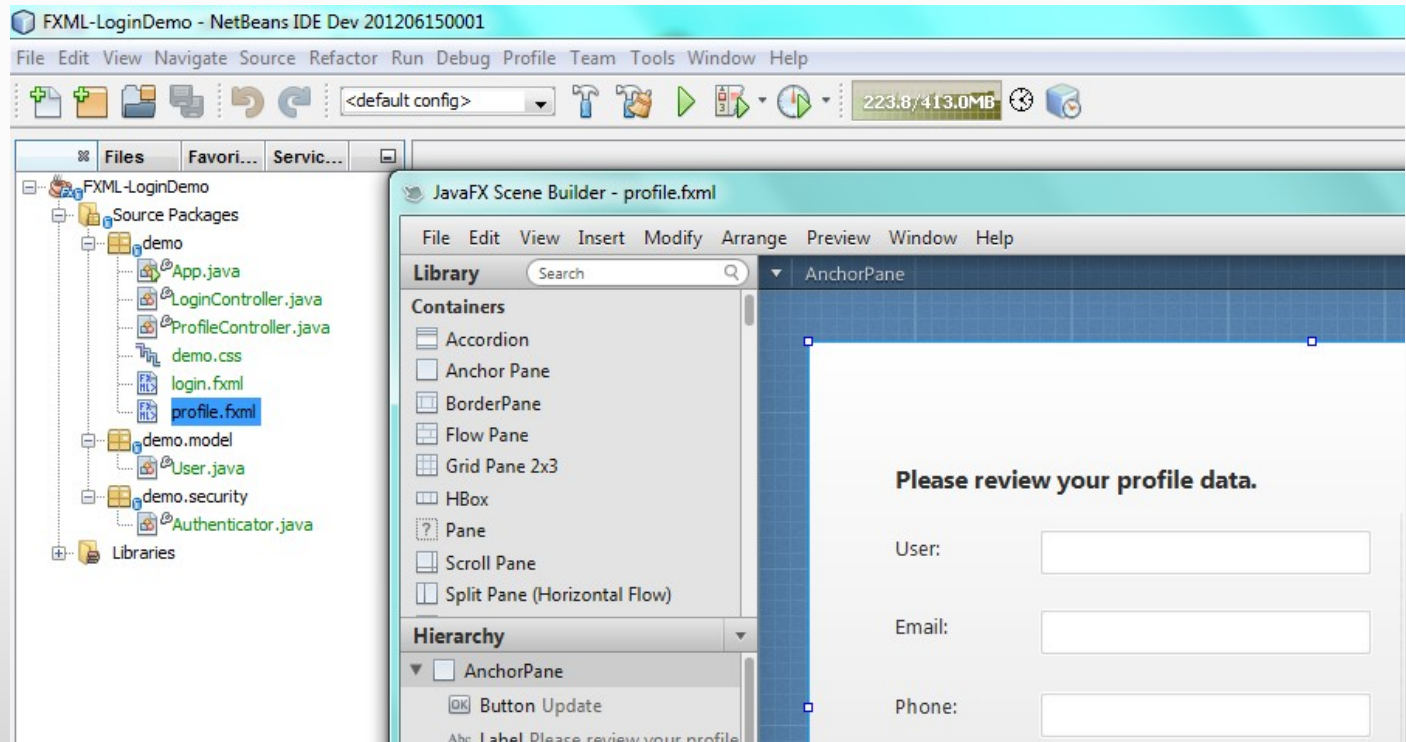
- First IDE to provide JavaFX tools
- Developed in co-operation with JavaFX Team
- Create, edit, compile, deploy
- Library of ready to use samples out of the box



# New in NetBeans JavaFX Support



- Open JavaFX files in Scene Builder
- Project type for JavaFX in Swing
- CSS code completion for JavaFX elements

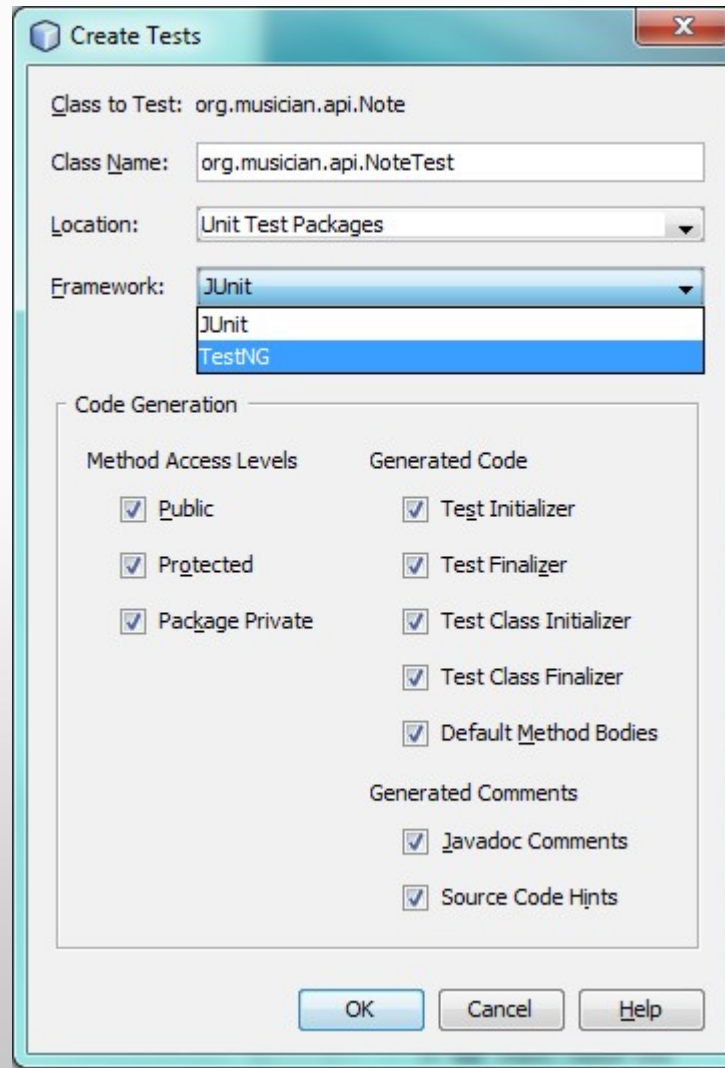


- TestNG
- Maven
- Groovy & Grails
- Debugger & Profiler
- PHP
- C++

# TestNG



- Both TestNG and JUnit supported
- Create new tests, test cases, test suites
- Test results in output window



# Maven



- Maven 3.0.4
- New JaCoCo for code coverage
- Enhanced scanning performance
- No. 1 reason users switch to NetBeans IDE

The screenshot shows the NetBeans IDE interface for a project named 'Maven Scrum Toys'. The left sidebar displays the project structure with folders for 'Web Pages' and 'Source Packages', and a list of Java files including 'AbstractEntity.java', 'PersistentEntity.java', 'Project.java', 'Sprint.java', 'SprintNameUniqu...', 'SprintNameUniqu...', 'Story.java', 'Task.java', and 'TaskStatus.java'. The main window is divided into two panes. The top pane, titled 'Code Coverage Report for Maven Scrum Toys', shows a 'Total Coverage' of 0.35% and a table of file coverage. The bottom pane, titled 'Test Results', shows a green progress bar for 100.00% coverage and a message 'All 5 tests passed. (0.065 s)'. A specific test result for 'jsf2.demo.scrum.model.entities.AbstractEntity' is highlighted.

Filename	Coverage	Total	Not Ex...
jsf2.dem...	0.00 %	2	2
jsf2.dem...	0.00 %	4	4
jsf2.dem...	60.00 %	5	2
jsf2.dem...	0.00 %	9	9

Test Results
com.sun.faces.jsf-scrumtoys:war:0.1
100.00 %
All 5 tests passed. (0.065 s)
jsf2.demo.scrum.model.entities.AbstractEntity

# Groovy & Grails



- Groovy 1.8.6
- Faster Groovy indexing & parsing
- Faster Groovy code completion

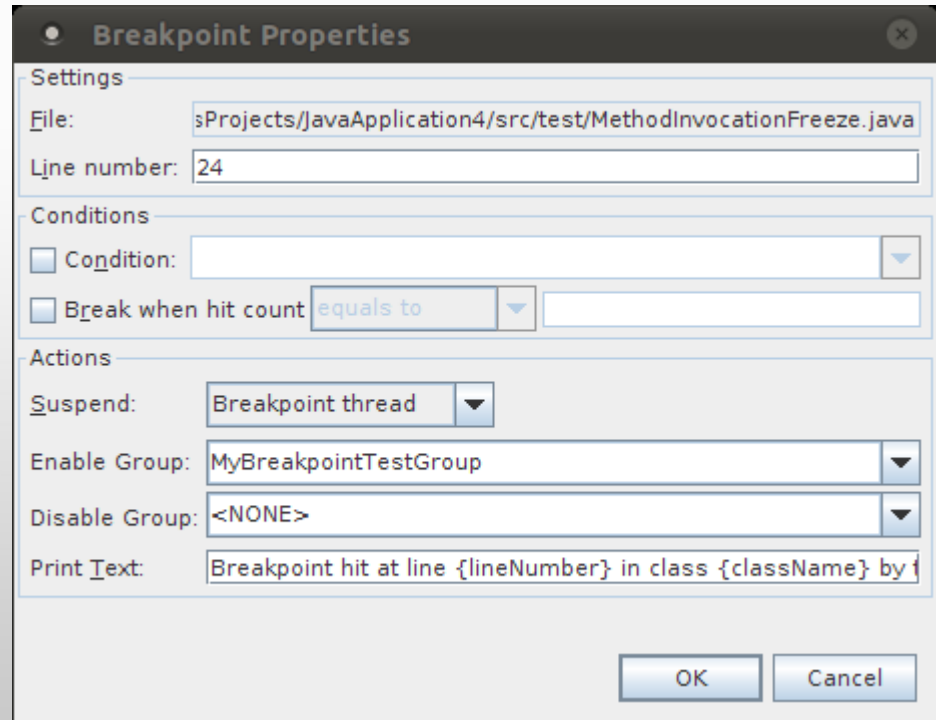
The screenshot shows the NetBeans IDE with a Groovy file named 'Bike.groovy' open. The code defines a 'Bike' class with attributes 'manufacturer', 'model', 'frame', 'serialNo', 'weight', 'status', and 'cost'. It includes a 'setCost' method and a 'toString' method. The 'toString' method uses Groovy's string interpolation to format the output. On the right, the 'Class Hierarchy' window is open, showing the 'manufacturer' class selected, with a list of its superclasses and interfaces, including 'MetaClass', 'Model', 'MalformedURLException', 'Map', 'MapEntry', 'MapWithDefault', 'Math', 'MetaArrayLengthProperty', 'MetaBeanProperty', 'MetaClass', 'MetaClassImpl', 'MetaClassRegistry', 'MetaClassRegistryChangeEvent', 'MetaClassRegistryChangeListener', 'MetaExpandProperty', and 'MetaMethod'.

```
1 package demo
2
3 class Bike {
4
5     String manufacturer
6     String model
7     Integer frame
8     String serialNo
9     Double weight
10    String status
11    BigDecimal cost
12
13    public void setCost(BigDecimal newCost) {
14        cost = newCost.setScale(2, RoundingMode.HALF_UP)
15    }
16
17
18    public String toString() {
19        return ""Bike:
20            manufacturer -- ${manufacturer}
21            model -- ${model}
22            frame -- ${frame}
23            serialNo -- ${serialNo}
24        ""
25    }
```

# Debugger & Profiler

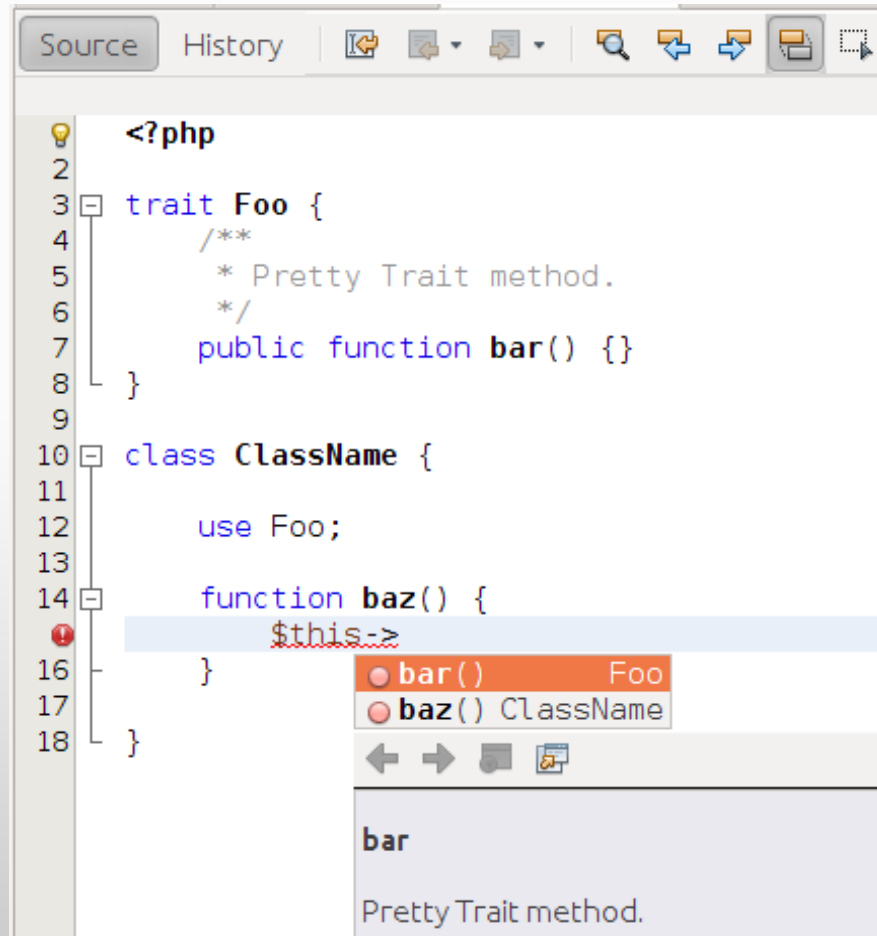


- Debugger
  - Breakpoints enable/disable other breakpoints
  - Source scanning does not block debugger stepping
  - Zoom added to Visual Debugger snapshots
  - Tree view in Variables and Watches
  - Code completion in New Breakpoint dialog





- PHP 5.4
- Support for
  - Symfony2
  - Twig
  - Doctrine2
  - ApiGen
  - Neon
- Tools
  - New hints
  - Hudson
  - PHPUnit & Annotations



```
<?php
2
3 trait Foo {
4     /**
5      * Pretty Trait method.
6      */
7     public function bar() {}
8 }
9
10 class ClassName {
11
12     use Foo;
13
14     function baz() {
15         $this->
16     }
17
18 }
```

Completion popup for `$this->`:

- `bar()` Foo
- `baz()` ClassName

Selected item: `bar`

Documentation: Pretty Trait method.

- C++11 Standard
- Faster “Find in Projects”
- Shell Syntax in Run Command
- Remote Toolbar

```
113  #include <algorithm>
114  void foo5() {
115      char s[] = "Hello C++11!";
116      int uppercase = 0;
117
118      std::for_each(s, s + sizeof (s), [&uppercase] (char c) {
119          if (isupper(c))
120              uppercase++;
121      });
122  }
123
124  template<class Lhs, class Rh>
125  auto adding_func(const Lhs &lhs, const Rh &rhs = lhs) -> decltype(lhs+rhs) {
126      return lhs + rhs;
127  }
```

## Next Steps



1. Download Netbeans IDE



2. Create a user account



3. Participate in NetBeans community



4. Blog about NetBeans

### Resources:

- NetBeans product site: <http://netbeans.org>
- NetBeans wiki: <http://wiki.netbeans.org>
- NetBeans blog aggregation: <http://planetnetbeans.org>

Thank You!

**SOFTWARE. HARDWARE. COMPLETE.**

ORACLE®