JavaCro'18

Take your productivity to the next level using Intellij IDEA and powerful utility tools

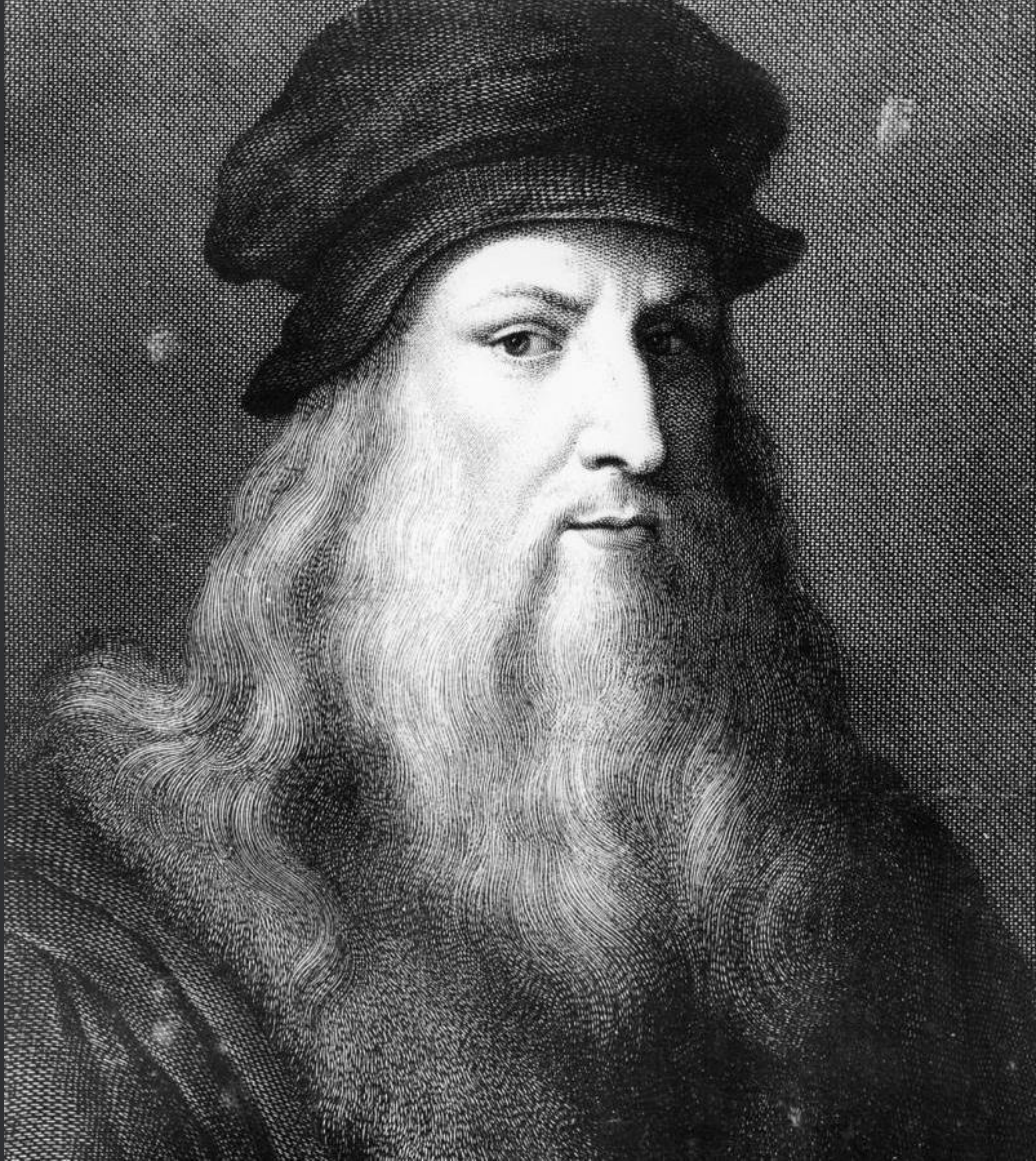
Red Island, 8.5.2018.

ecx.io
an IBM Company

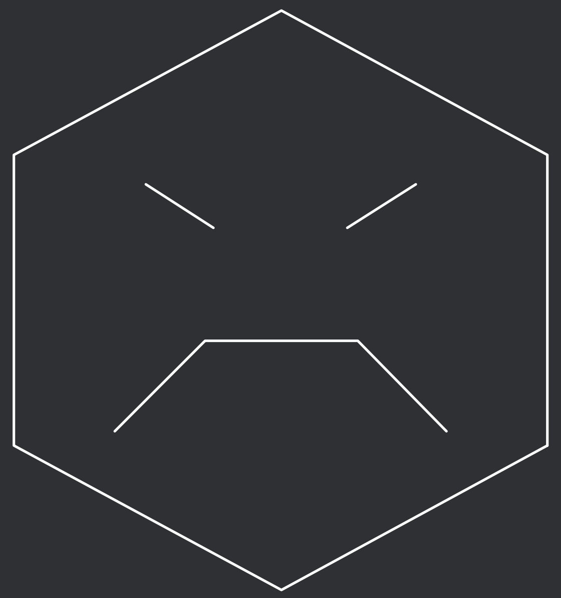
Why utility tools?

Skill is important as well as tools!

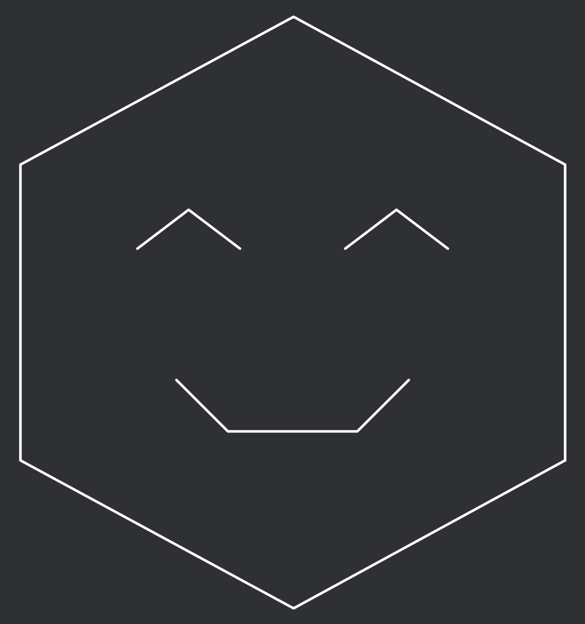
Da Vinci with a mop and a bucket of mud may be a better painter than you, but he would never beat Da Vinci with quality tools.



Complexity...



Complexity is the enemy of execution!



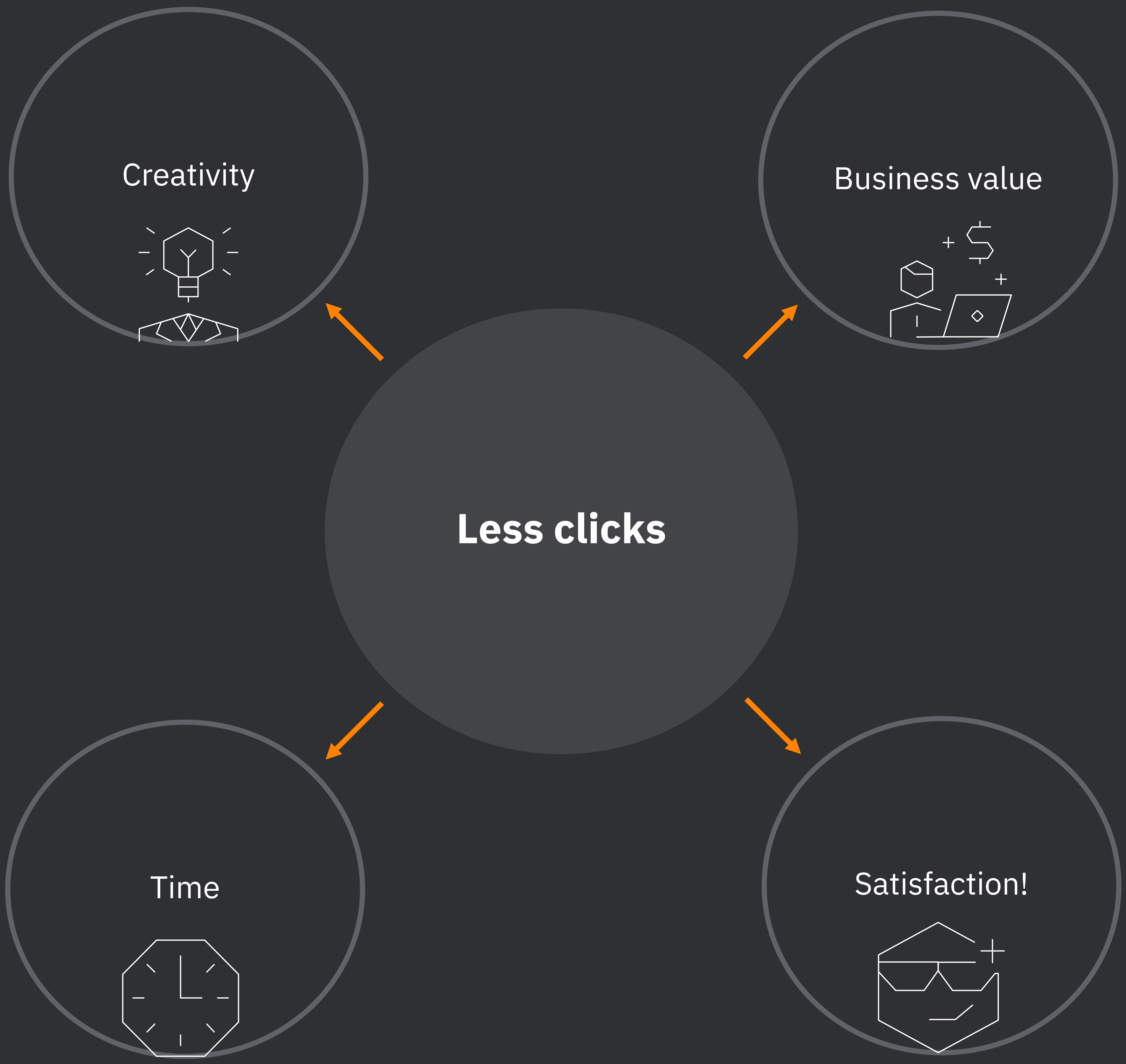
Simplify, simplify, simplify!
Automate, automate, automate!

Keyboard vs Mouse

Aim

1 or 0

key presses


Less clicks
Creativity
Business value
Time
Satisfaction!

Productivity tools

Java IDE

Notes & Sharing

Navigation & Search

Clipboard manager

Global hotkeys

Java IDE

Why IntelliJ IDEA?

Benefits of Intellij IDEA

Staying in the flow

- by using only keyboard

Powerful actions

- actions are searchable

Code generation and refactoring

- a lot of options with undo feature

Debugging

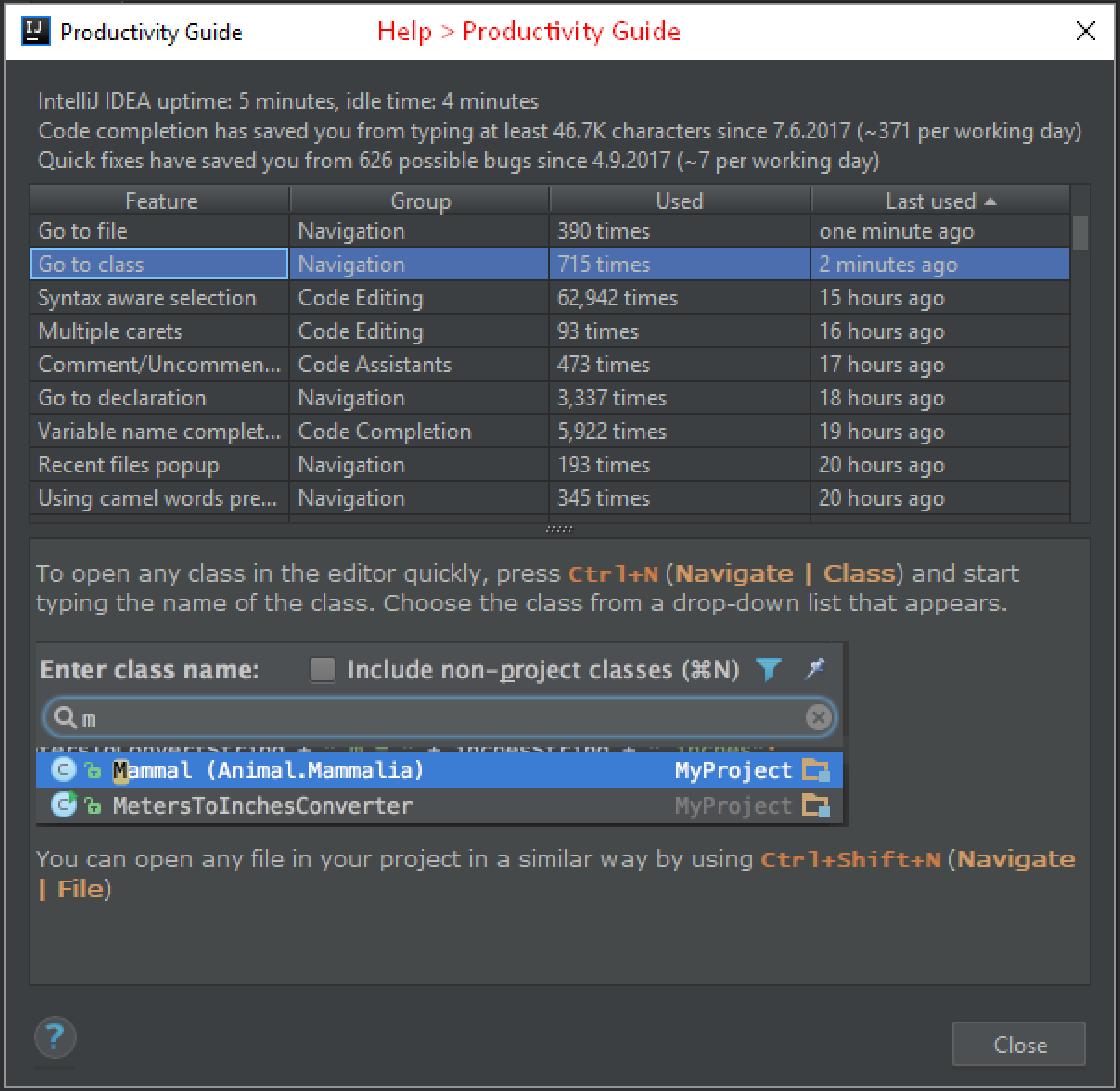
- changing values on the fly

Plugins and built-in tools

- Git, Maven, Terminal, Local History etc.

Powerful Actions

How to measure and increase productivity?



Productivity Guide

Help > Productivity Guide

IntelliJ IDEA uptime: 5 minutes, idle time: 4 minutes
Code completion has saved you from typing at least 46.7K characters since 7.6.2017 (~371 per working day)
Quick fixes have saved you from 626 possible bugs since 4.9.2017 (~7 per working day)

Feature	Group	Used	Last used ▲
Go to file	Navigation	390 times	one minute ago
Go to class	Navigation	715 times	2 minutes ago
Syntax aware selection	Code Editing	62,942 times	15 hours ago
Multiple carets	Code Editing	93 times	16 hours ago
Comment/Uncommen...	Code Assistants	473 times	17 hours ago
Go to declaration	Navigation	3,337 times	18 hours ago
Variable name complet...	Code Completion	5,922 times	19 hours ago
Recent files popup	Navigation	193 times	20 hours ago
Using camel words pre...	Navigation	345 times	20 hours ago

To open any class in the editor quickly, press **Ctrl+N** (**Navigate | Class**) and start typing the name of the class. Choose the class from a drop-down list that appears.

Enter class name: ☐ Include non-project classes (⌘N)

```
m
Mammal (Animal.Mammalia)                MyProject
MetersToInchesConverter                 MyProject
```

You can open any file in your project in a similar way by using **Ctrl+Shift+N** (**Navigate | File**)

Close

The most powerful shortcuts

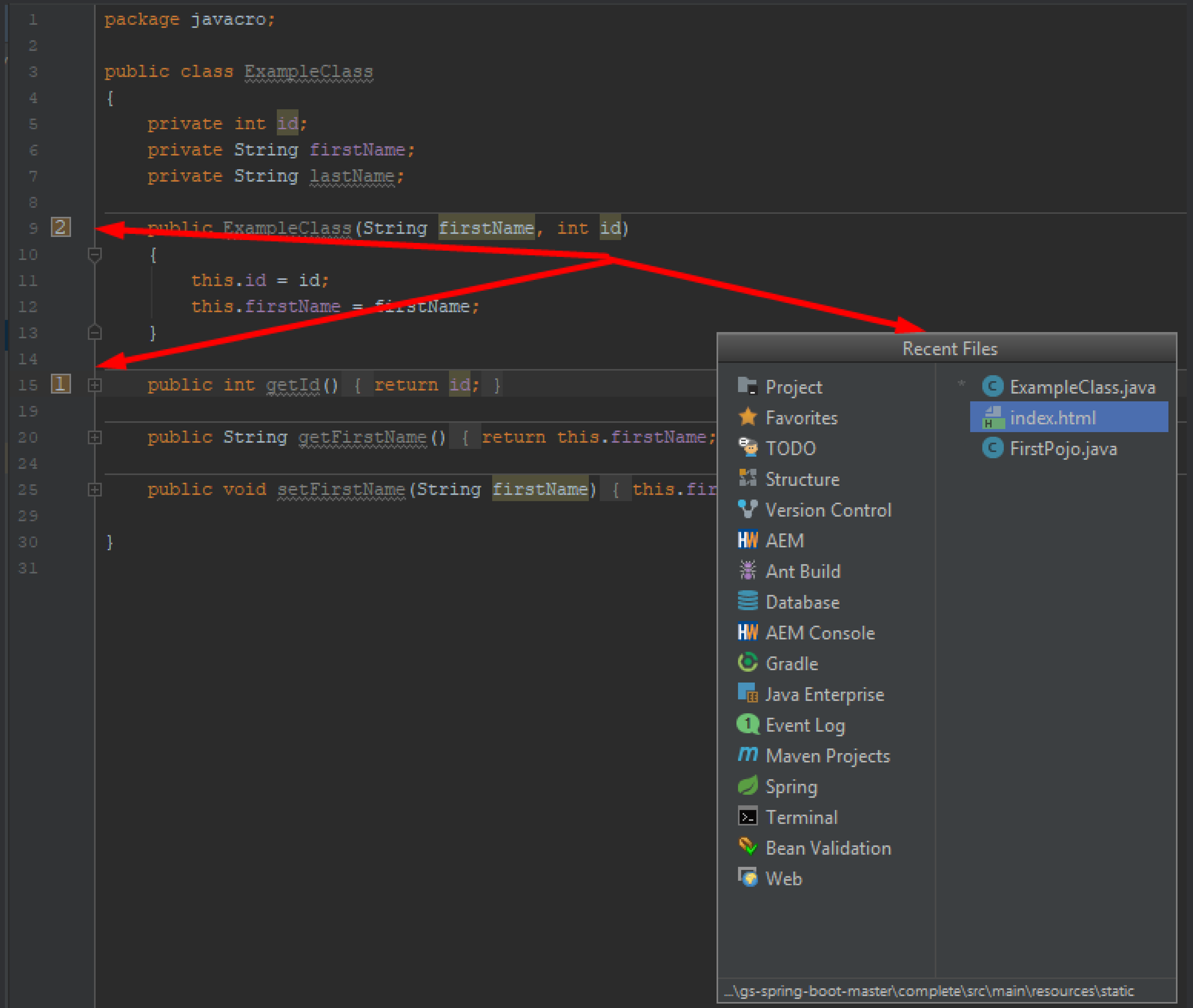
- Find Action
  - *Ctrl + Shift + A*
- Search Everywhere
  - *Shift + Shift* (press it quickly)
- Show intention actions
  - *Alt + Enter*

Powerful Actions

Navigation

Use only ONE TAB!

- History
  - *Ctrl + Alt + left or right arrow key*
- Recent files
  - *Ctrl + E*
- Bookmarks
  - Set/remove a bookmark
    - *Ctrl + Shift + number*
  - Go to a bookmark
    - *Ctrl + number*



```
package javacro;

public class ExampleClass
{
    private int id;
    private String firstName;
    private String lastName;

    public ExampleClass(String firstName, int id)
    {
        this.id = id;
        this.firstName = firstName;
    }

    public int getId() { return id; }

    public String getFirstName() { return this.firstName;

    public void setFirstName(String firstName) { this.fir

}
```

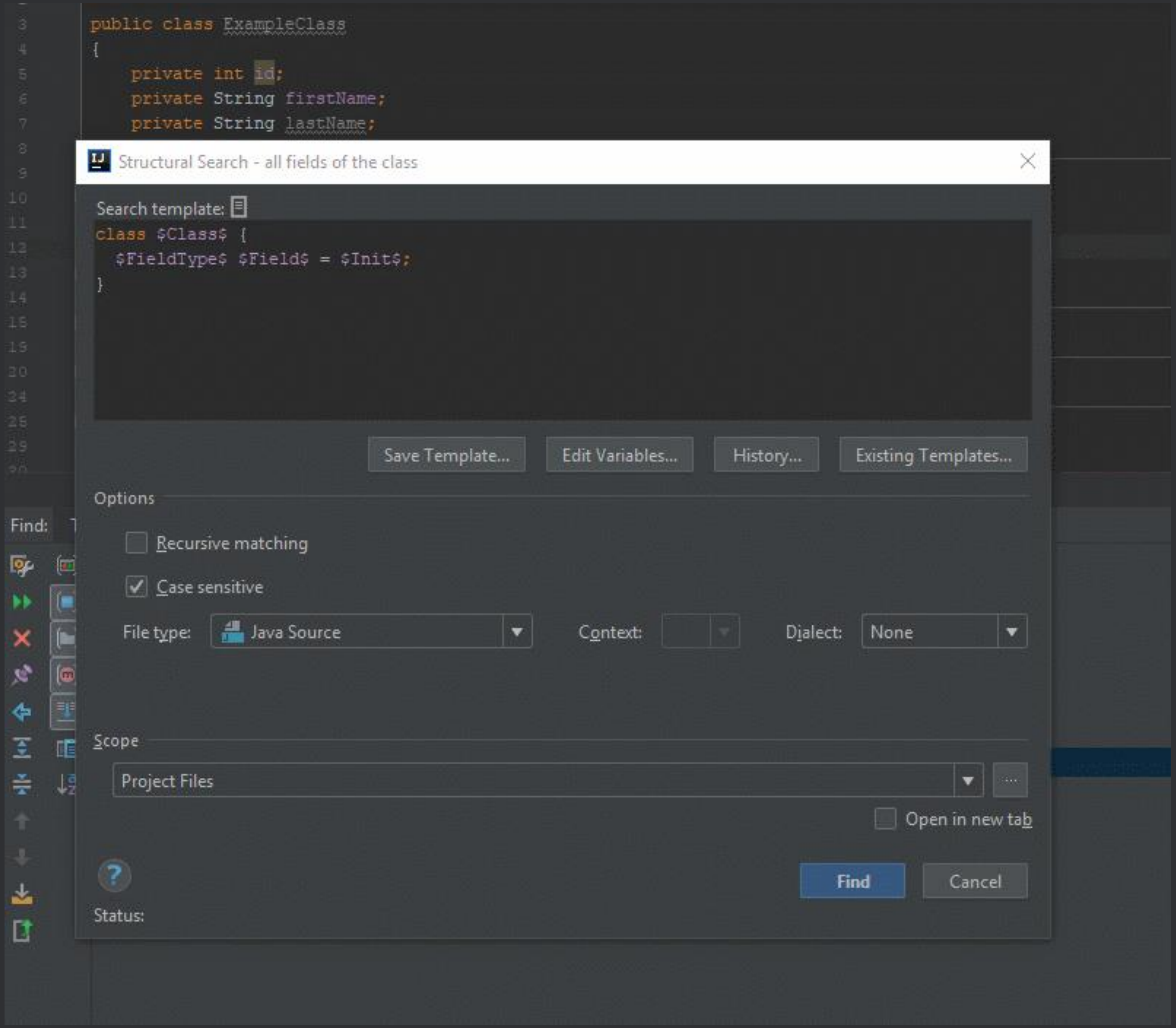
Navigation

Use **one step** solutions instead of manual navigation!

Efficient file navigation

- Navigate to Class
  - *Ctrl + N*
- Navigate to file
  - *Ctrl + Shift + N*
- Navigate to symbols
  - *Ctrl + Shift + Alt + N*
- Find File
  - *Ctrl + Shift + F*
- Structural search (advanced)

Structural search and replace!



```
public class ExampleClass
{
    private int id;
    private String firstName;
    private String lastName;
```

Structural Search - all fields of the class

Search template:

```
class $Class$ {
  $FieldType$ $Field$ = $Init$;
}
```

Options

- [ ] Recursive matching
- [x] Case sensitive

File type: Java Source   Context: Dialect: None

Scope

Project Files

- [ ] Open in new tab

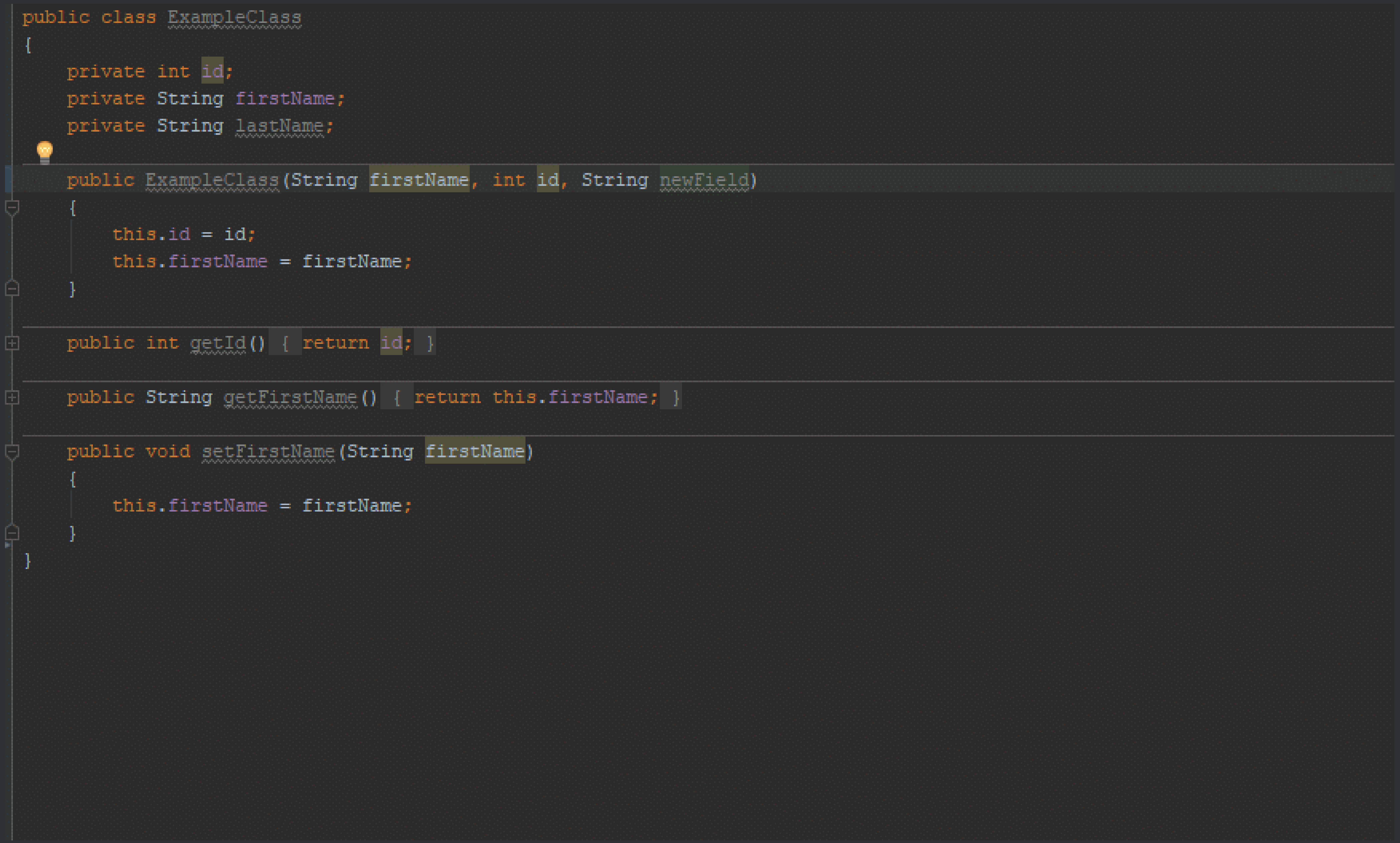
Status:

Powerful Actions
Navigation
Code Generation

Code Generation

Generate code on almost each line you type - easy way!

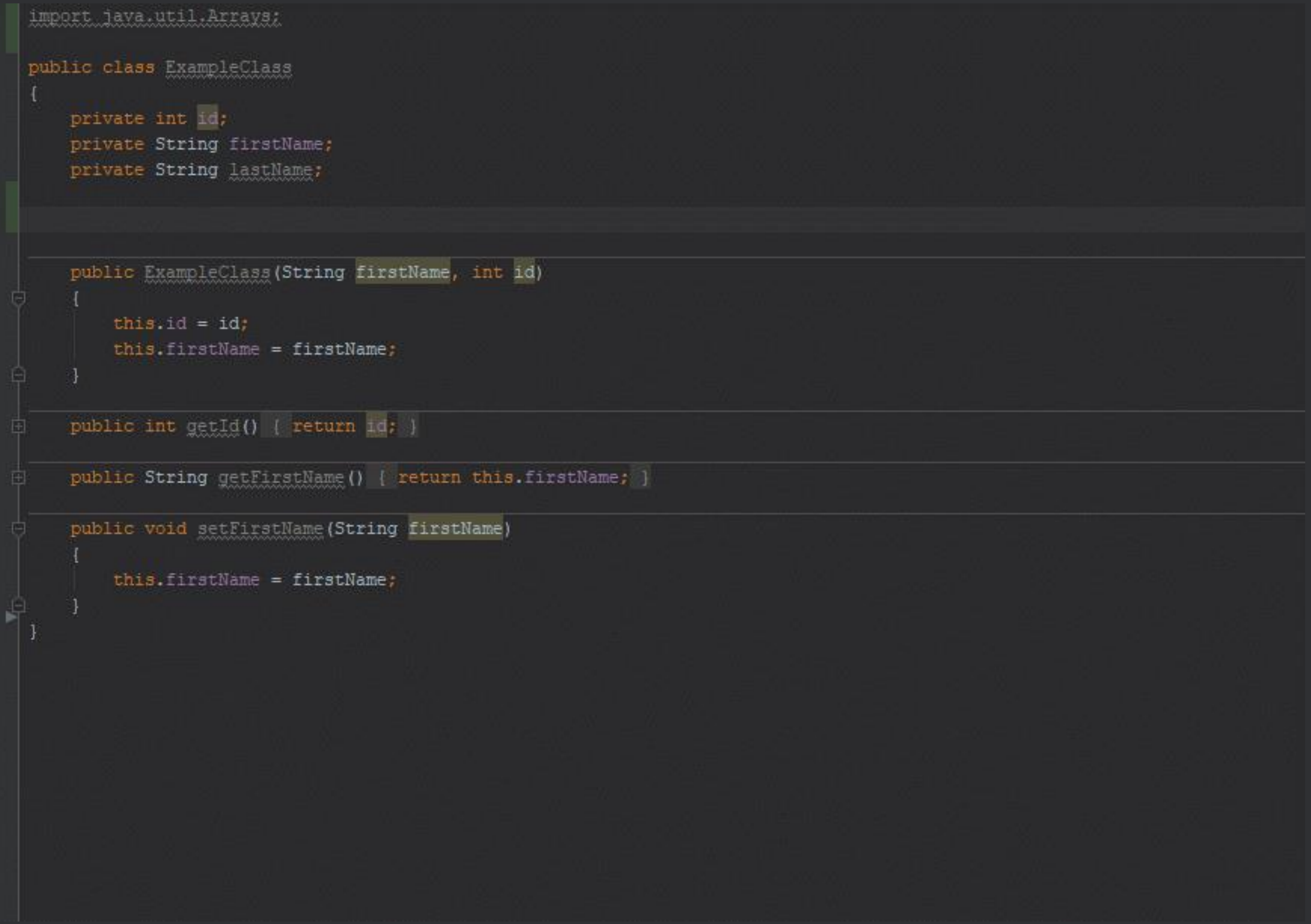
Generate a field from constructor parameter *Alt* + *Enter*



```
public class ExampleClass
{
    private int id;
    private String firstName;
    private String lastName;

    public ExampleClass(String firstName, int id, String newField)
    {
        this.id = id;
        this.firstName = firstName;
    }

    public int getId() { return id; }

    public String getFirstName() { return this.firstName; }

    public void setFirstName(String firstName)
    {
        this.firstName = firstName;
    }
}
```

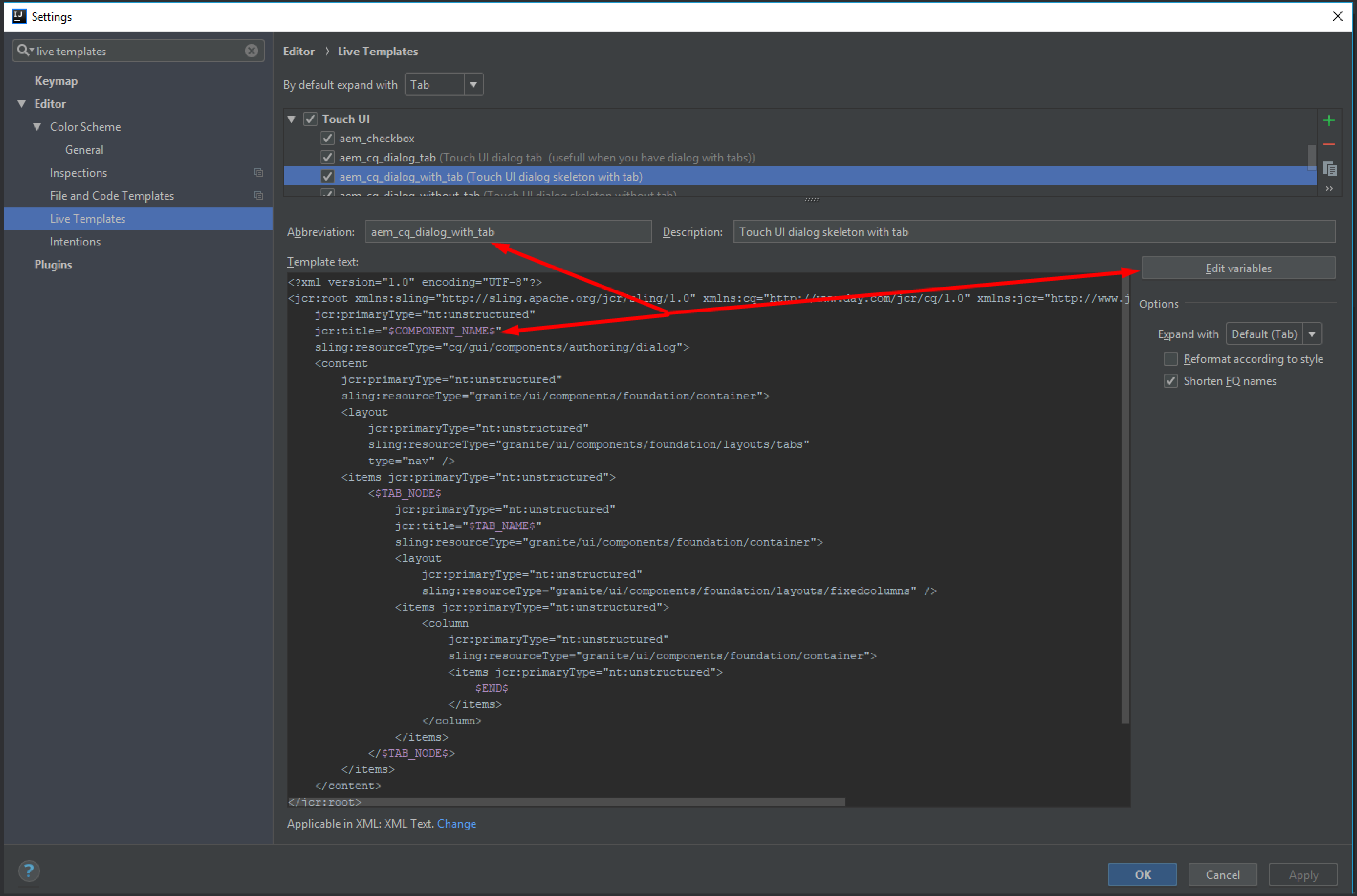
Live Templates

- OOTB templates:
  - *psvm*
  - *sout*
  - *iter* (foreach)
    - *itar* (arrays)
    - *ritar* (reverse, arrays)



```
import java.util.Arrays;

public class ExampleClass
{
    private int id;
    private String firstName;
    private String lastName;



    public ExampleClass(String firstName, int id)
    {
        this.id = id;
        this.firstName = firstName;
    }

    public int getId() { return id; }

    public String getFirstName() { return this.firstName; }

    public void setFirstName(String firstName)
    {
        this.firstName = firstName;
    }
}
```

Live Templates Configuration



Settings

live templates

- Keymap
- Editor
  - Color Scheme
    - General
  - Inspections
  - File and Code Templates
  - Live Templates
  - Intentions
- Plugins

Editor › Live Templates

By default expand with: Tab

- ✓ Touch UI
  - ✓ aem_checkbox
  - ✓ aem_cq_dialog_tab (Touch UI dialog tab (usefull when you have dialog with tabs))
  - ✓ aem_cq_dialog_with_tab (Touch UI dialog skeleton with tab)
  - ✓ aem_cq_dialog_without_tab (Touch UI dialog skeleton without tab)

Abbreviation: aem_cq_dialog_with_tab

Description: Touch UI dialog skeleton with tab

Template text:

```
<?xml version="1.0" encoding="UTF-8"?>
<jcr:root xmlns:sling="http://sling.apache.org/jcr/sling/1.0" xmlns:cq="http://www.day.com/jcr/cq/1.0" xmlns:jcr="http://www.j
    jcr:primaryType="nt:unstructured"
    jcr:title="$COMPONENT_NAME$"
    sling:resourceType="cq/gui/components/authoring/dialog">
    <content
        jcr:primaryType="nt:unstructured"
        sling:resourceType="granite/ui/components/foundation/container">
        <layout
            jcr:primaryType="nt:unstructured"
            sling:resourceType="granite/ui/components/foundation/layouts/tabs"
            type="nav" />
        <items jcr:primaryType="nt:unstructured">
            <$TAB_NODE$
                jcr:primaryType="nt:unstructured"
                jcr:title="$TAB_NAME$"
                sling:resourceType="granite/ui/components/foundation/container">
                <layout
                    jcr:primaryType="nt:unstructured"
                    sling:resourceType="granite/ui/components/foundation/layouts/fixedcolumns" />
                <items jcr:primaryType="nt:unstructured">
                    <column
                        jcr:primaryType="nt:unstructured"
                        sling:resourceType="granite/ui/components/foundation/container">
                        <items jcr:primaryType="nt:unstructured">
                            $END$
                        </items>
                    </column>
                </items>
            </$TAB_NODE$>
        </items>
    </content>
</jcr:root>
```

Edit variables

Options

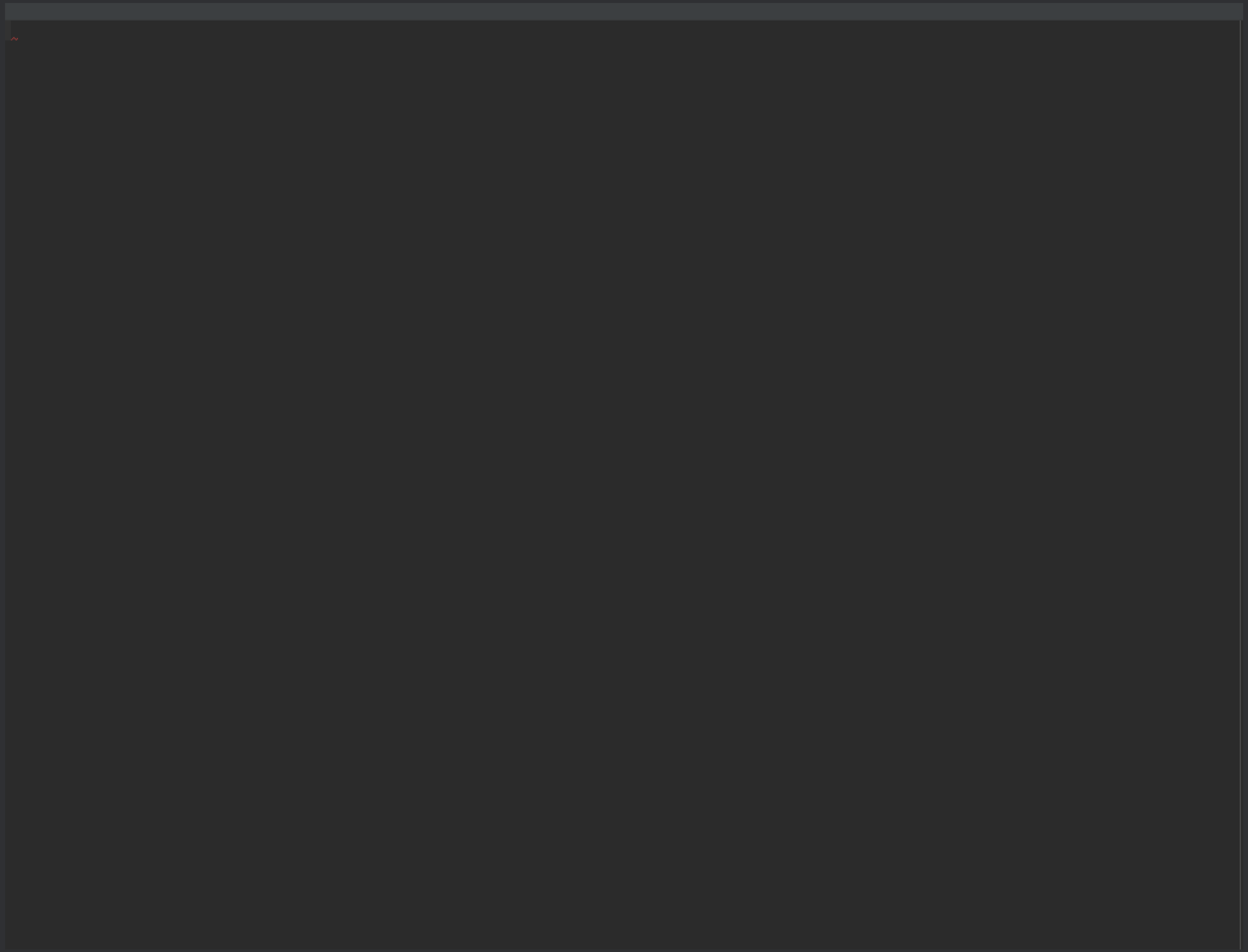
Expand with: Default (Tab)

- [ ] Reformat according to style
- [✓] Shorten FQ names

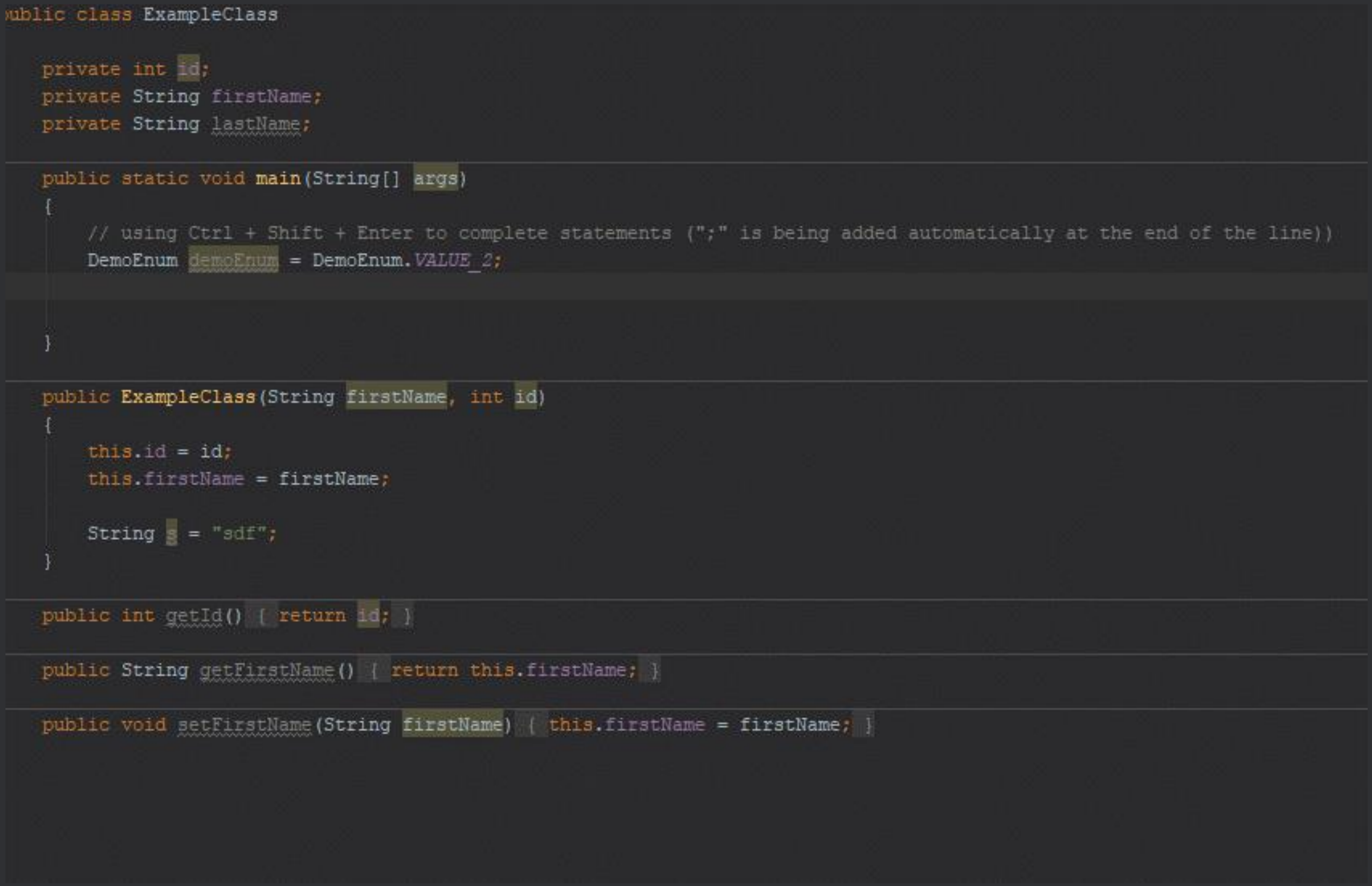
Applicable in XML: XML Text. Change

OK Cancel Apply

Advanced Live Templates: AEM framework example



Complete Current Statement: *Ctrl* + *Shift* + *Enter*



```
ublic class ExampleClass

   private int id;
   private String firstName;
   private String lastName;

   public static void main(String[] args)
   {
       // using Ctrl + Shift + Enter to complete statements (";" is being added automatically at the end of the line))
       DemoEnum demoEnum = DemoEnum.VALUE_2;


   }

   public ExampleClass(String firstName, int id)
   {
       this.id = id;
       this.firstName = firstName;

       String s = "sdf";
   }

   public int getId() { return id; }

   public String getFirstName() { return this.firstName; }

   public void setFirstName(String firstName) { this.firstName = firstName; }
```

Powerful Features

Navigation

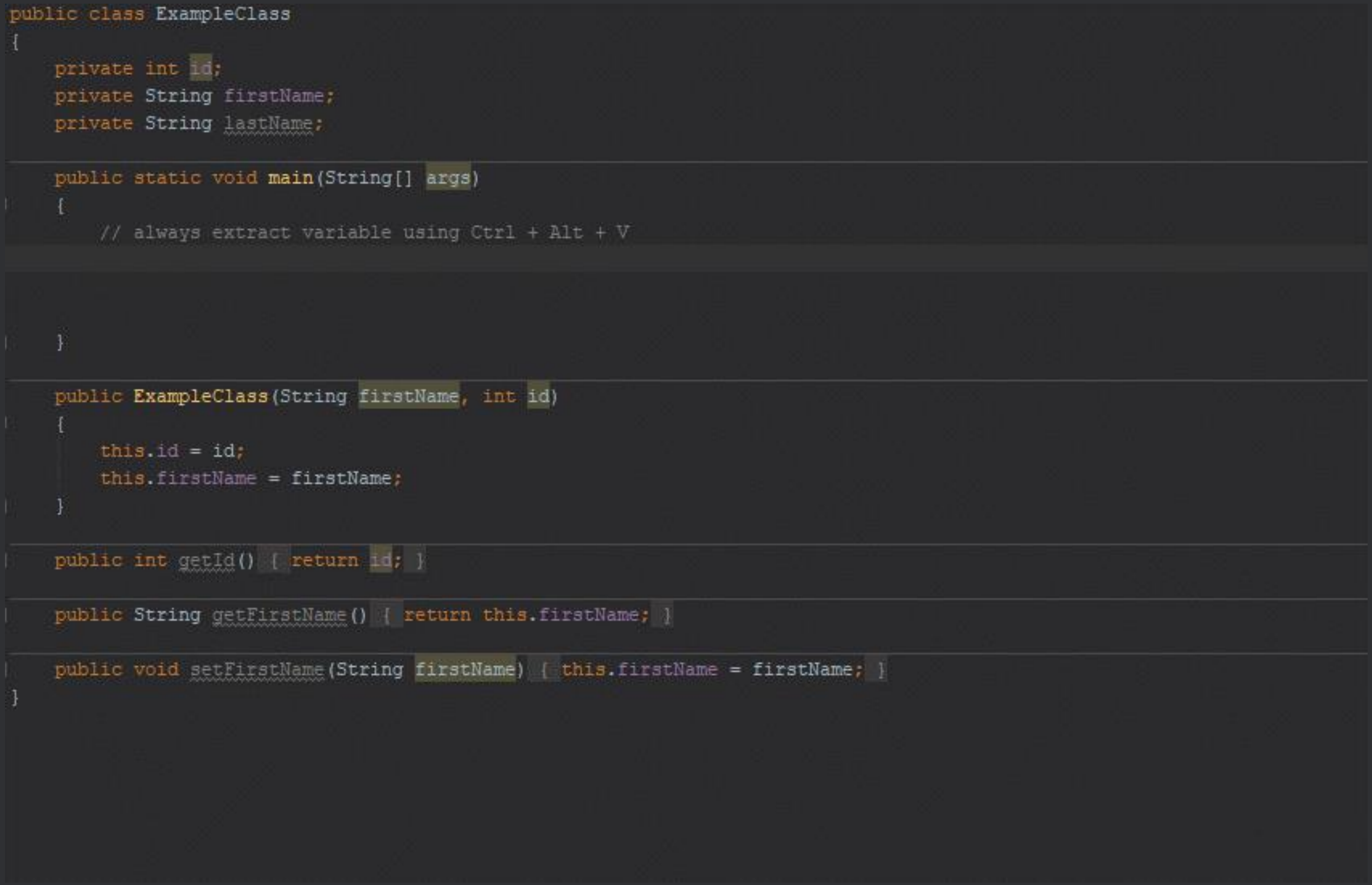
Code Generation

Refactoring

Refactoring

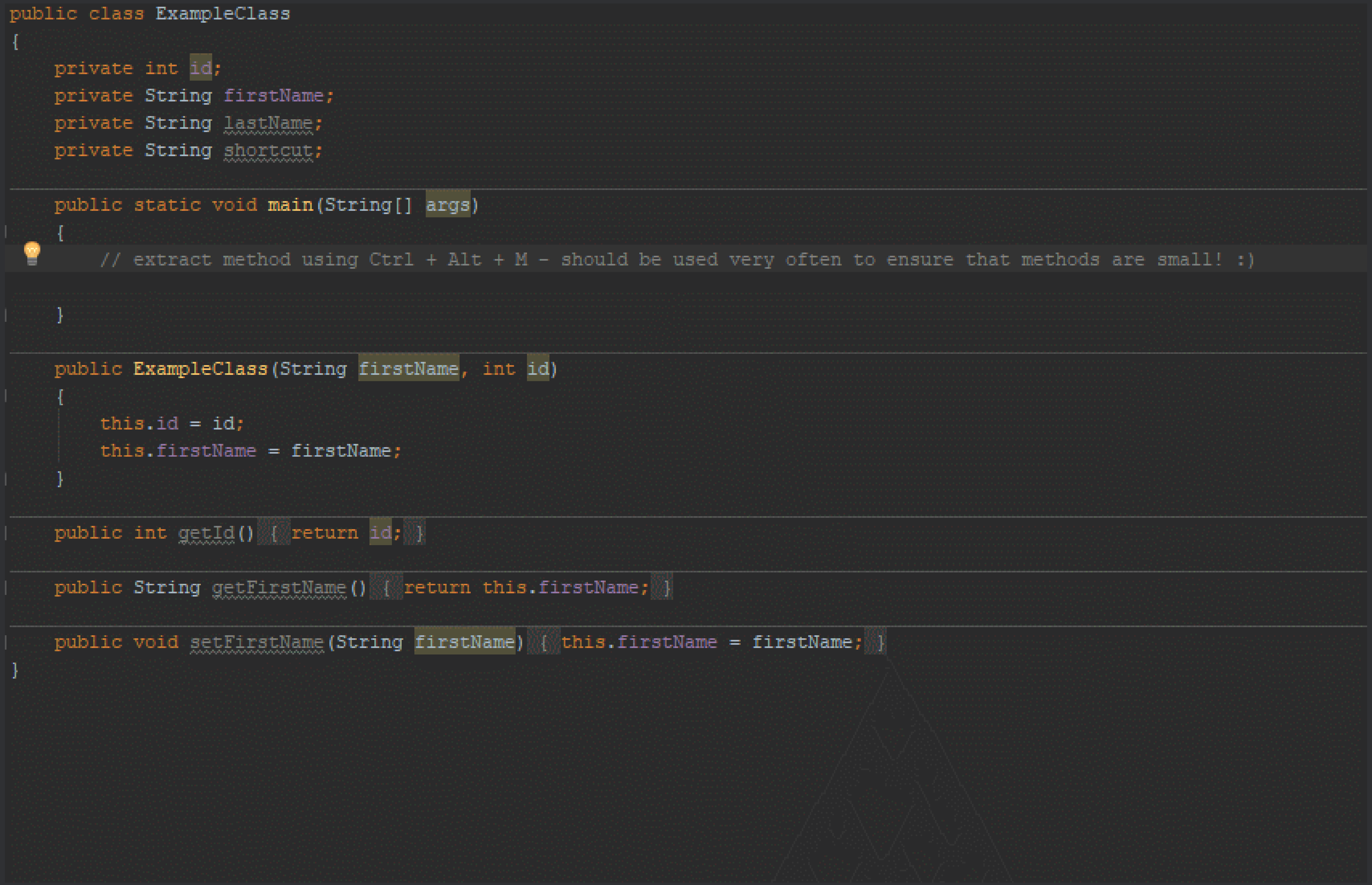
Basic refactoring while coding ensures much higher code quality!

Extract Variable: *Ctrl* + *Alt* + *V*



```
public class ExampleClass
{
    private int id;
    private String firstName;
    private String lastName;

    public static void main(String[] args)
    {
        // always extract variable using Ctrl + Alt + V



    }

    public ExampleClass(String firstName, int id)
    {
        this.id = id;
        this.firstName = firstName;
    }

    public int getId() { return id; }

    public String getFirstName() { return this.firstName; }

    public void setFirstName(String firstName) { this.firstName = firstName; }
}
```

Extract Method: *Ctrl* + *Alt* + *M*



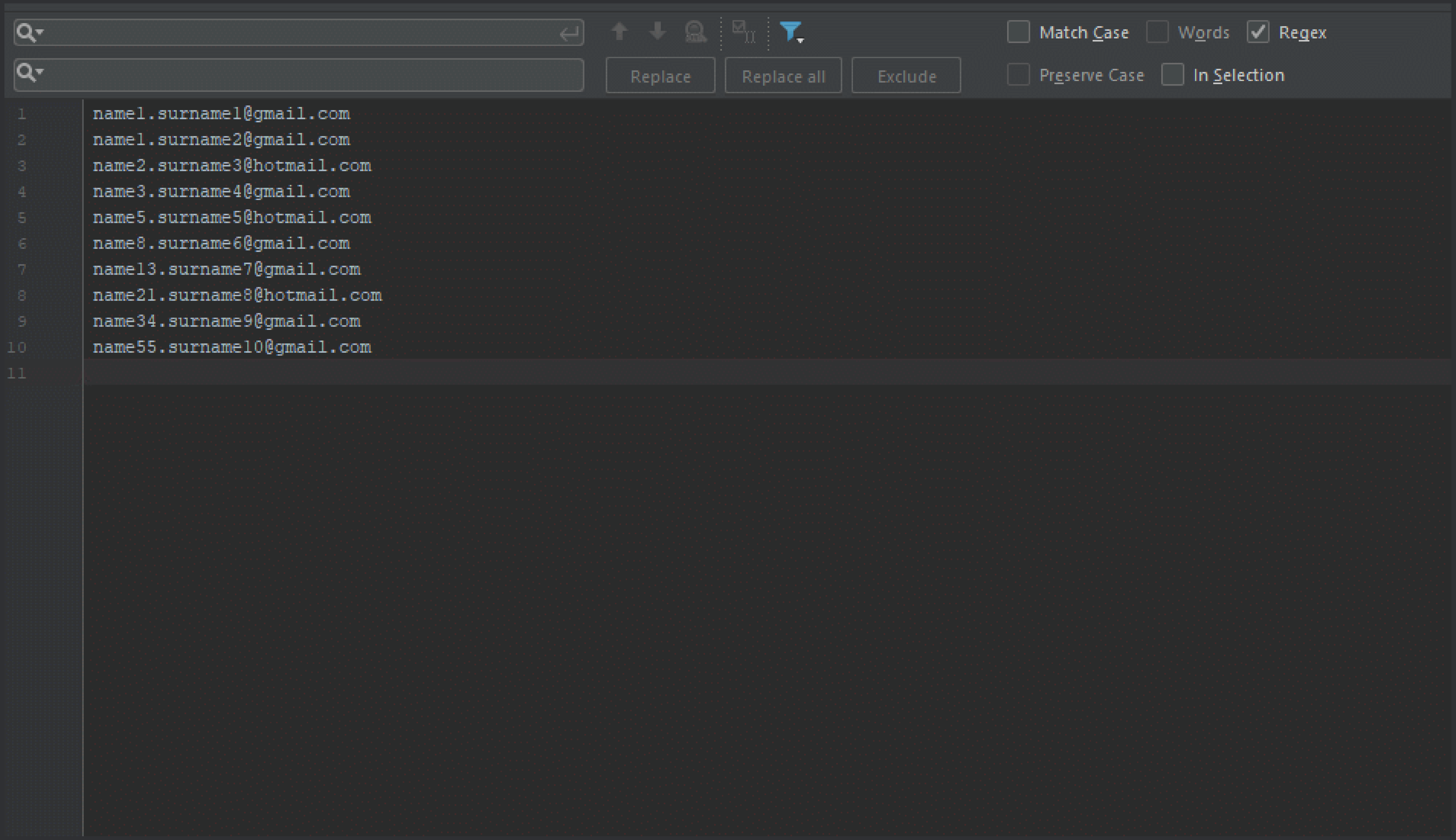
```java
public class ExampleClass
{
    private int id;
    private String firstName;
    private String lastName;
    private String shortcut;

    public static void main(String[] args)
    {
        // extract method using Ctrl + Alt + M - should be used very often to ensure that methods are small! :)

    }

    public ExampleClass(String firstName, int id)
    {
        this.id = id;
        this.firstName = firstName;
    }

    public int getId() { return id; }

    public String getFirstName() { return this.firstName; }

    public void setFirstName(String firstName) { this.firstName = firstName; }
}
```

What is your first reaction when you hear word “Regex”?

COMMON

SEIZE THE REGEX POWER!

Simple Regex Usage with Preview

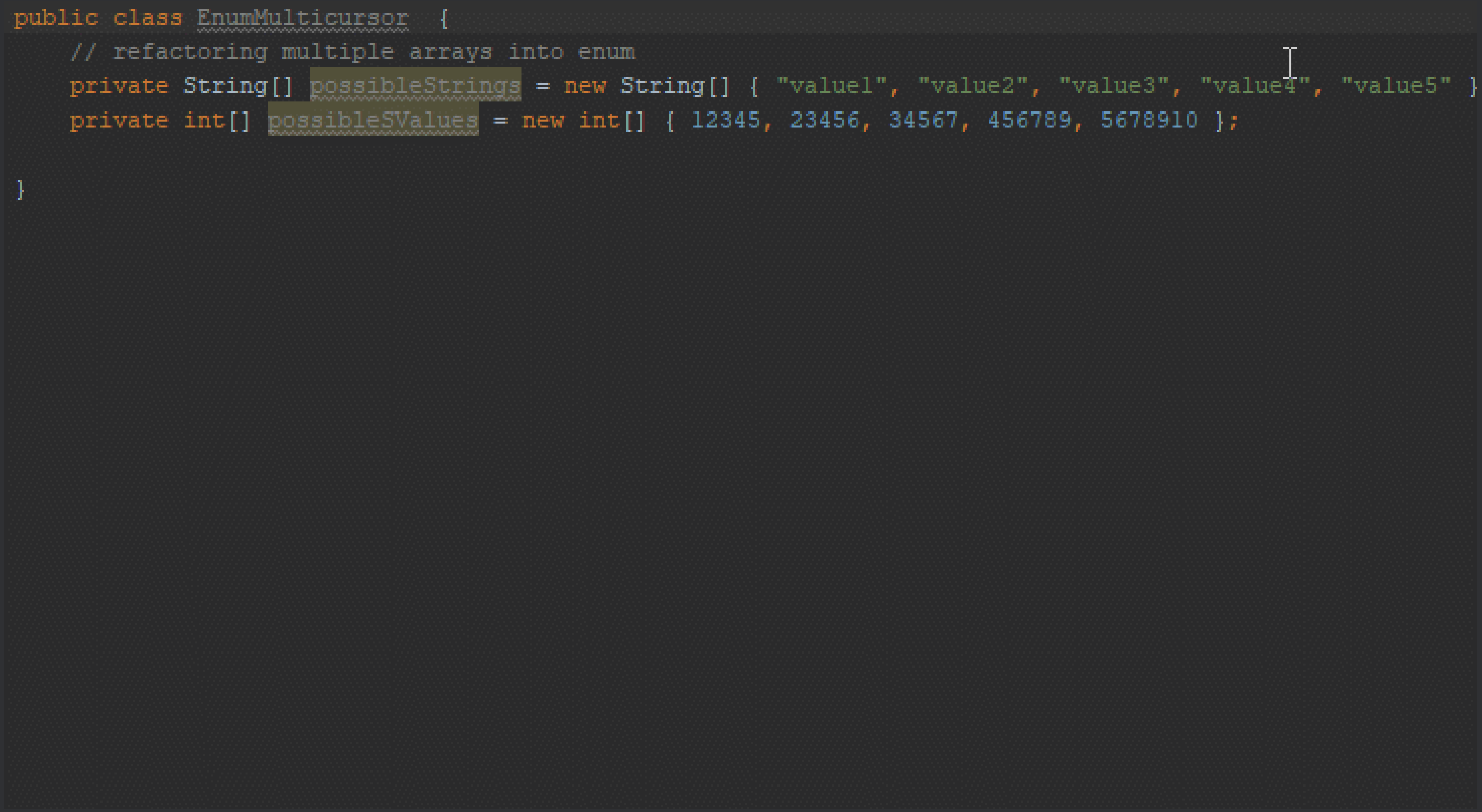


Match Case [ ] Words [ ] Regex [✓]
Preserve Case [ ] In Selection [ ]

```
name1.surname1@gmail.com
name1.surname2@gmail.com
name2.surname3@hotmail.com
name3.surname4@gmail.com
name5.surname5@hotmail.com
name8.surname6@gmail.com
name13.surname7@gmail.com
name21.surname8@hotmail.com
name34.surname9@gmail.com
name55.surname10@gmail.com
```

HIDDEN IN PLAIN SIGHT - MULTICURSORS

Advanced Feature: Multicursors *Alt* + *Shift* + *Left click*



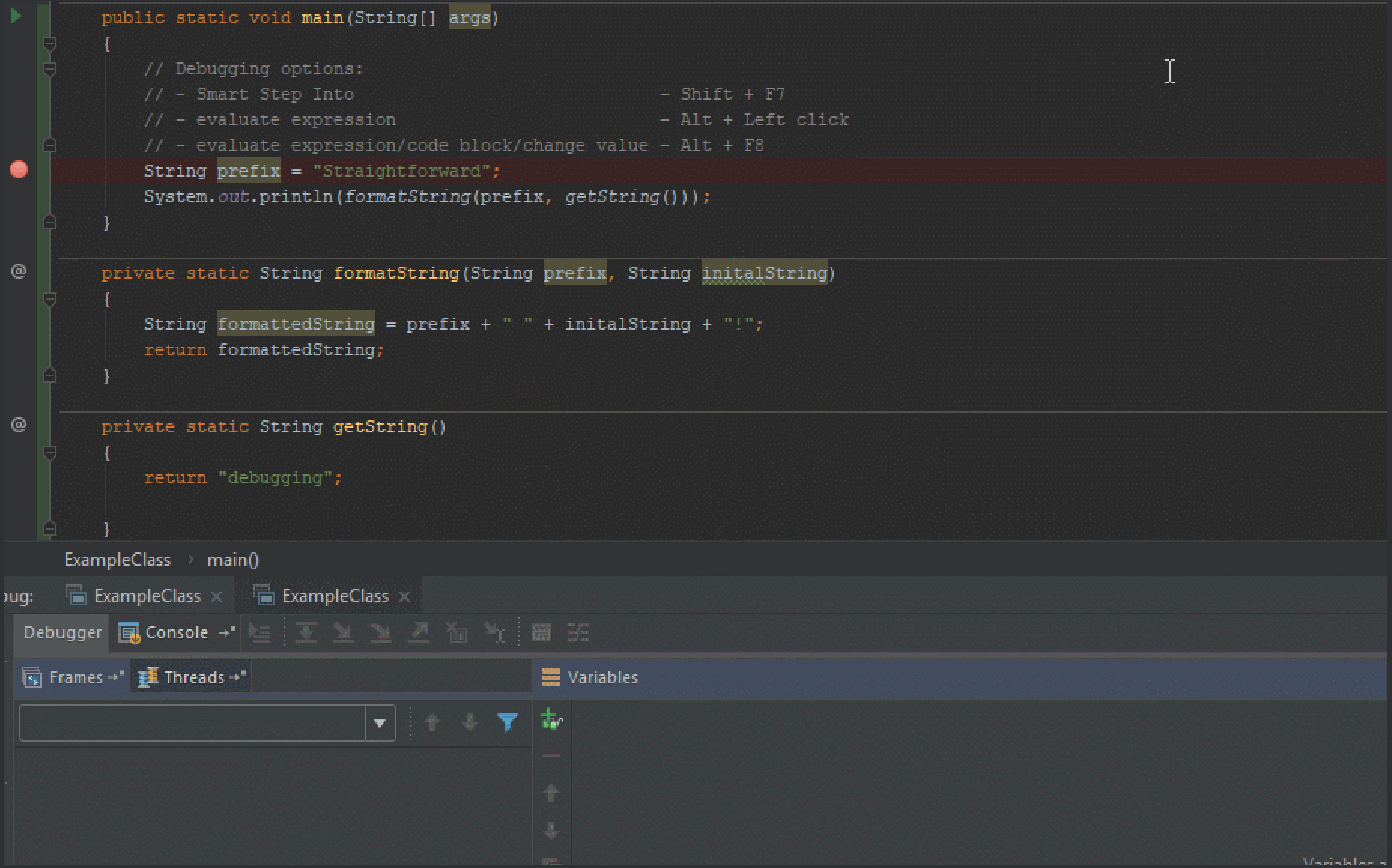
```
public class EnumMulticursor  {
    // refactoring multiple arrays into enum
    private String[] possibleStrings = new String[] { "value1", "value2", "value3", "value4", "value5" }
    private int[] possibleSValues = new int[] { 12345, 23456, 34567, 456789, 5678910 };

}
```

Powerful Features
Navigation
Code Generation
Refactoring
Debugging

Debugging

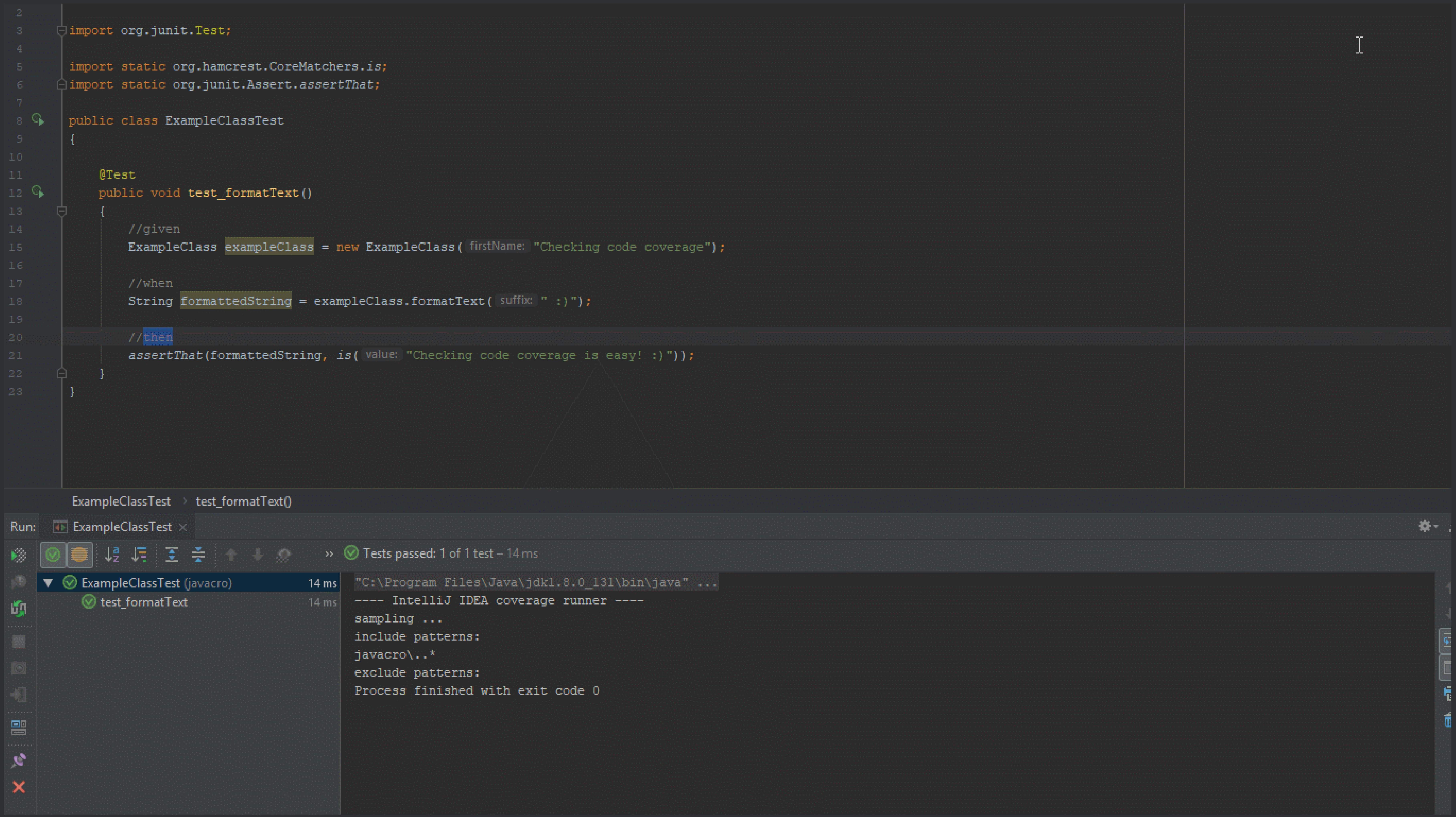
- Variable’s value is shown inside code editor!
- Remote debugging
- Built-in code coverage tool

Debugging



```
public static void main(String[] args)
{
    // Debugging options:
    // - Smart Step Into                              - Shift + F7
    // - evaluate expression                          - Alt + Left click
    // - evaluate expression/code block/change value - Alt + F8
    String prefix = "Straightforward";
    System.out.println(formatString(prefix, getString()));
}

private static String formatString(String prefix, String initalString)
{
    String formattedString = prefix + " " + initalString + "!";
    return formattedString;
}

private static String getString()
{
    return "debugging";

}
```

Debugging: Code Coverage



```java
import org.junit.Test;

import static org.hamcrest.CoreMatchers.is;
import static org.junit.Assert.assertThat;

public class ExampleClassTest
{

    @Test
    public void test_formatText()
    {
        //given
        ExampleClass exampleClass = new ExampleClass( firstName: "Checking code coverage");

        //when
        String formattedString = exampleClass.formatText( suffix: " :)");

        //then
        assertThat(formattedString, is( value: "Checking code coverage is easy! :)"));
    }
}
```

ExampleClassTest › test_formatText()

Run: ExampleClassTest

Tests passed: 1 of 1 test – 14 ms

ExampleClassTest (javacro)	14 ms
test_formatText	14 ms

```
"C:\Program Files\Java\jdk1.8.0_131\bin\java" ...
---- IntelliJ IDEA coverage runner ----
sampling ...
include patterns:
javacro\..*
exclude patterns:
Process finished with exit code 0
```

Powerful Features

Navigation

Code Generation

Refactoring

Debugging

Plugins & Built-in Features

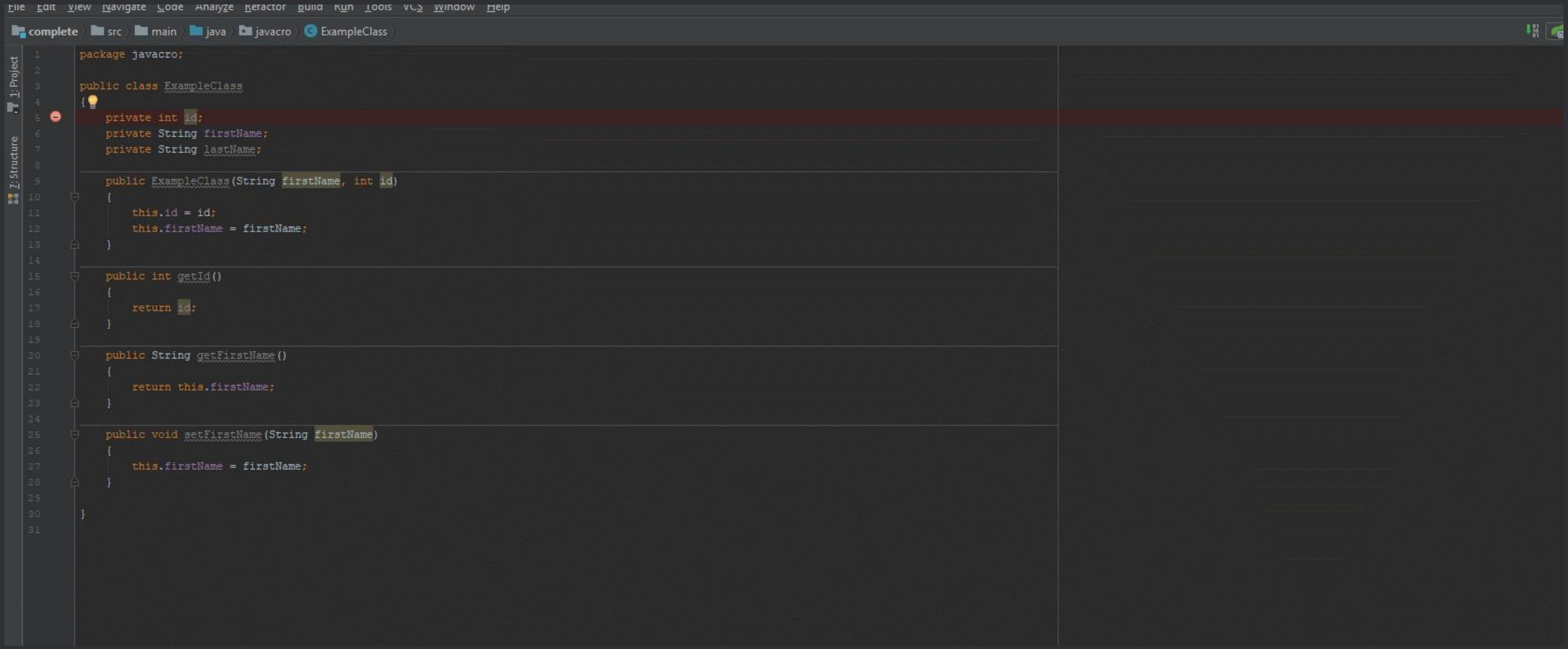
Plugins

- StringManipulation
- .gitignore plugin
- AceJump (alternative exists)
- ...

Built-in Features

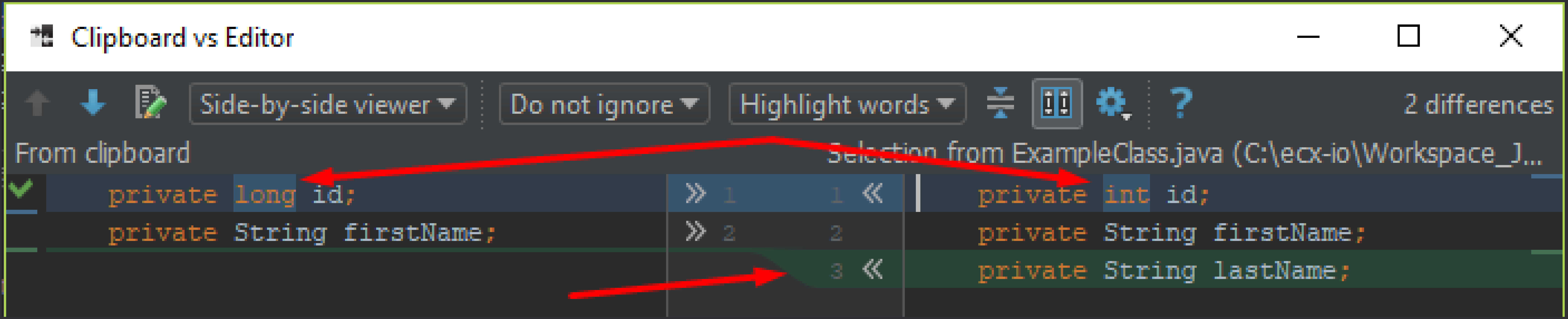
- Compare with Clipboard
- Local History
- Git Integration
- Language Injection
- Terminal

Compare with Clipboard


File Edit View Navigate Code Analyze Refactor Build Run Tools VCS Window Help
complete src main java javacro ExampleClass
1: Project
7: Structure

```java
package javacro;

public class ExampleClass
{
    private int id;
    private String firstName;
    private String lastName;

    public ExampleClass(String firstName, int id)
    {
        this.id = id;
        this.firstName = firstName;
    }

    public int getId()
    {
        return id;
    }

    public String getFirstName()
    {
        return this.firstName;
    }

    public void setFirstName(String firstName)
    {
        this.firstName = firstName;
    }

}
```

Compare with Clipboard


Clipboard vs Editor

Side-by-side viewer | Do not ignore | Highlight words | 2 differences

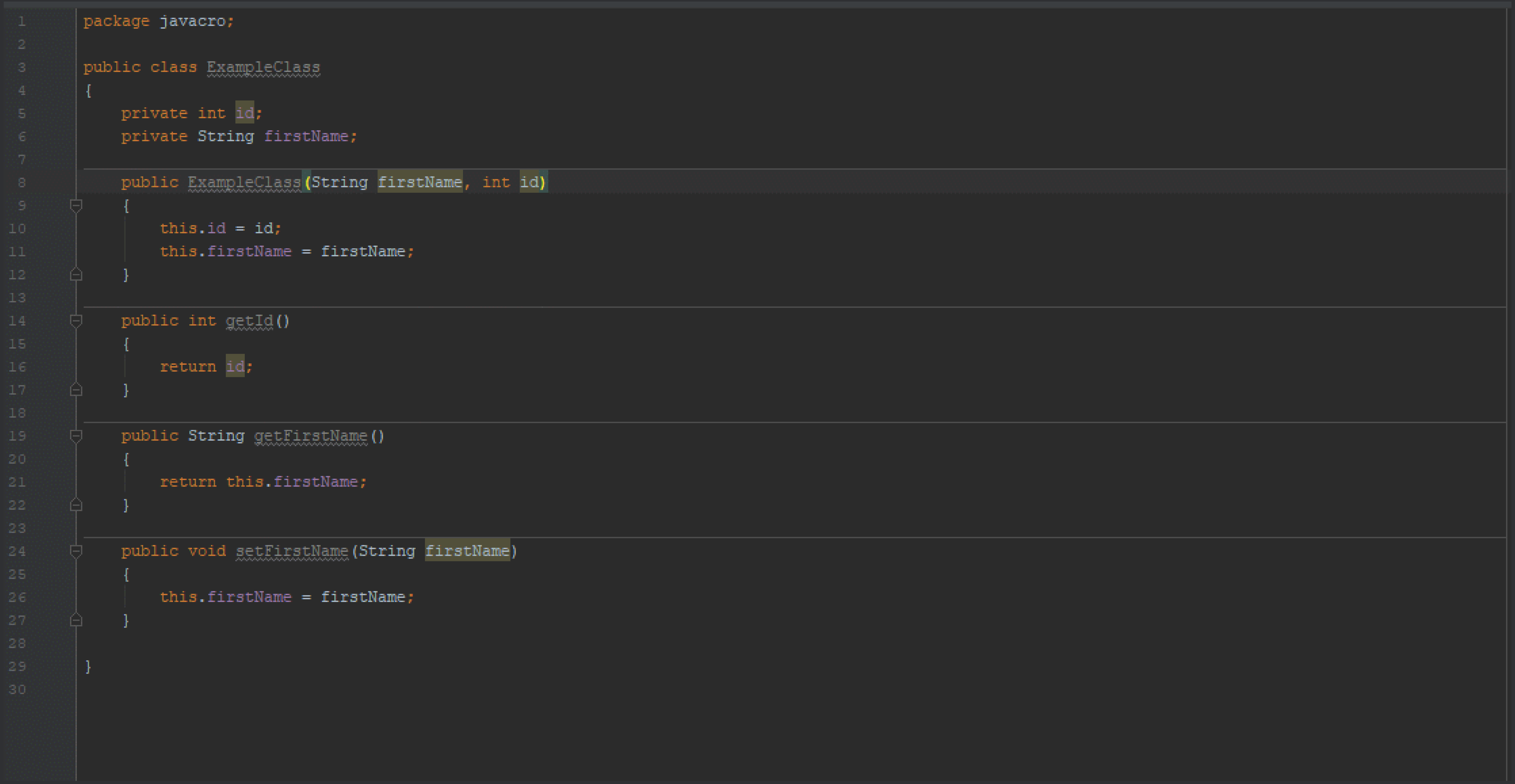
From clipboard

```
private long id;
private String firstName;
```

Selection from ExampleClass.java (C:\ecx-io\Workspace_J...

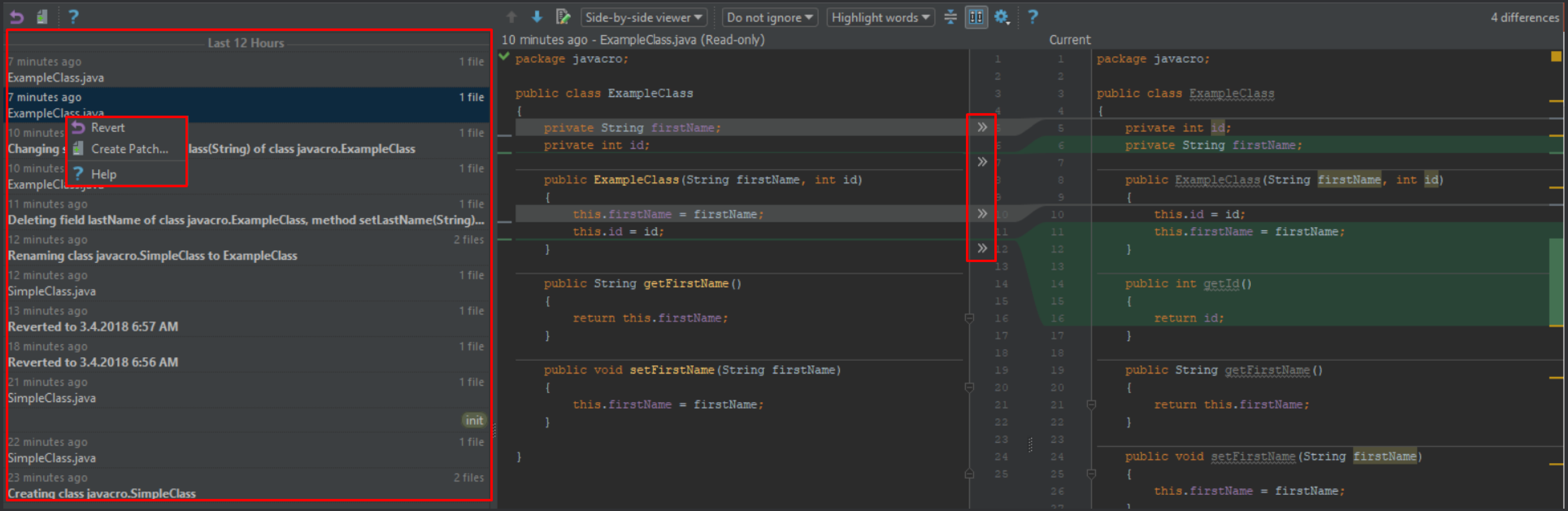
```
private int id;
private String firstName;
private String lastName;
```

Local History: Opening



```
package javacro;

public class ExampleClass
{
    private int id;
    private String firstName;

    public ExampleClass(String firstName, int id)
    {
        this.id = id;
        this.firstName = firstName;
    }

    public int getId()
    {
        return id;
    }

    public String getFirstName()
    {
        return this.firstName;
    }

    public void setFirstName(String firstName)
    {
        this.firstName = firstName;
    }

}
```

Local History: Differences



Last 12 Hours

7 minutes ago ExampleClass.java	1 file
7 minutes ago ExampleClass.java	1 file
10 minutes Changing s … lass(String) of class javacro.ExampleClass	1 file
10 minutes ExampleCl…	1 file
11 minutes ago Deleting field lastName of class javacro.ExampleClass, method setLastName(String)...	1 file
12 minutes ago Renaming class javacro.SimpleClass to ExampleClass	2 files
12 minutes ago SimpleClass.java	1 file
13 minutes ago Reverted to 3.4.2018 6:57 AM	1 file
18 minutes ago Reverted to 3.4.2018 6:56 AM	1 file
21 minutes ago SimpleClass.java	1 file
	init
22 minutes ago SimpleClass.java	1 file
23 minutes ago Creating class javacro.SimpleClass	2 files

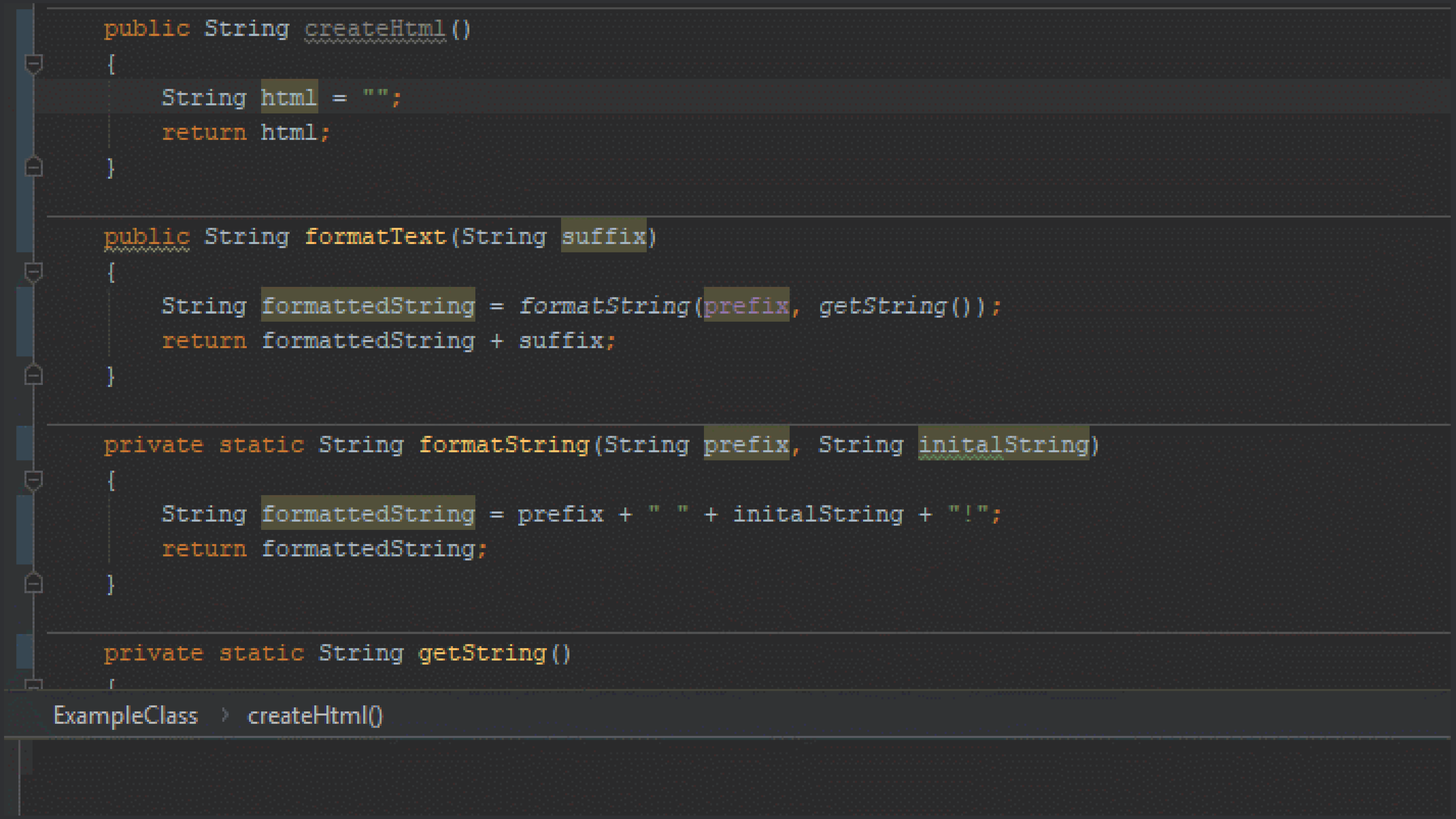
10 minutes ago - ExampleClass.java (Read-only)

```
package javacro;

public class ExampleClass
{
    private String firstName;
    private int id;

    public ExampleClass(String firstName, int id)
    {
        this.firstName = firstName;
        this.id = id;
    }

    public String getFirstName()
    {
        return this.firstName;
    }

    public void setFirstName(String firstName)
    {
        this.firstName = firstName;
    }

}
```

Current

```
package javacro;

public class ExampleClass
{
    private int id;
    private String firstName;

    public ExampleClass(String firstName, int id)
    {
        this.id = id;
        this.firstName = firstName;
    }

    public int getId()
    {
        return id;
    }

    public String getFirstName()
    {
        return this.firstName;
    }

    public void setFirstName(String firstName)
    {
        this.firstName = firstName;
```

Language Injection *Alt* + *Enter*



```
public String createHtml()
{
    String html = "";
    return html;
}

public String formatText(String suffix)
{
    String formattedString = formatString(prefix, getString());
    return formattedString + suffix;
}

private static String formatString(String prefix, String initalString)
{
    String formattedString = prefix + " " + initalString + "!";
    return formattedString;
}

private static String getString()
{
```

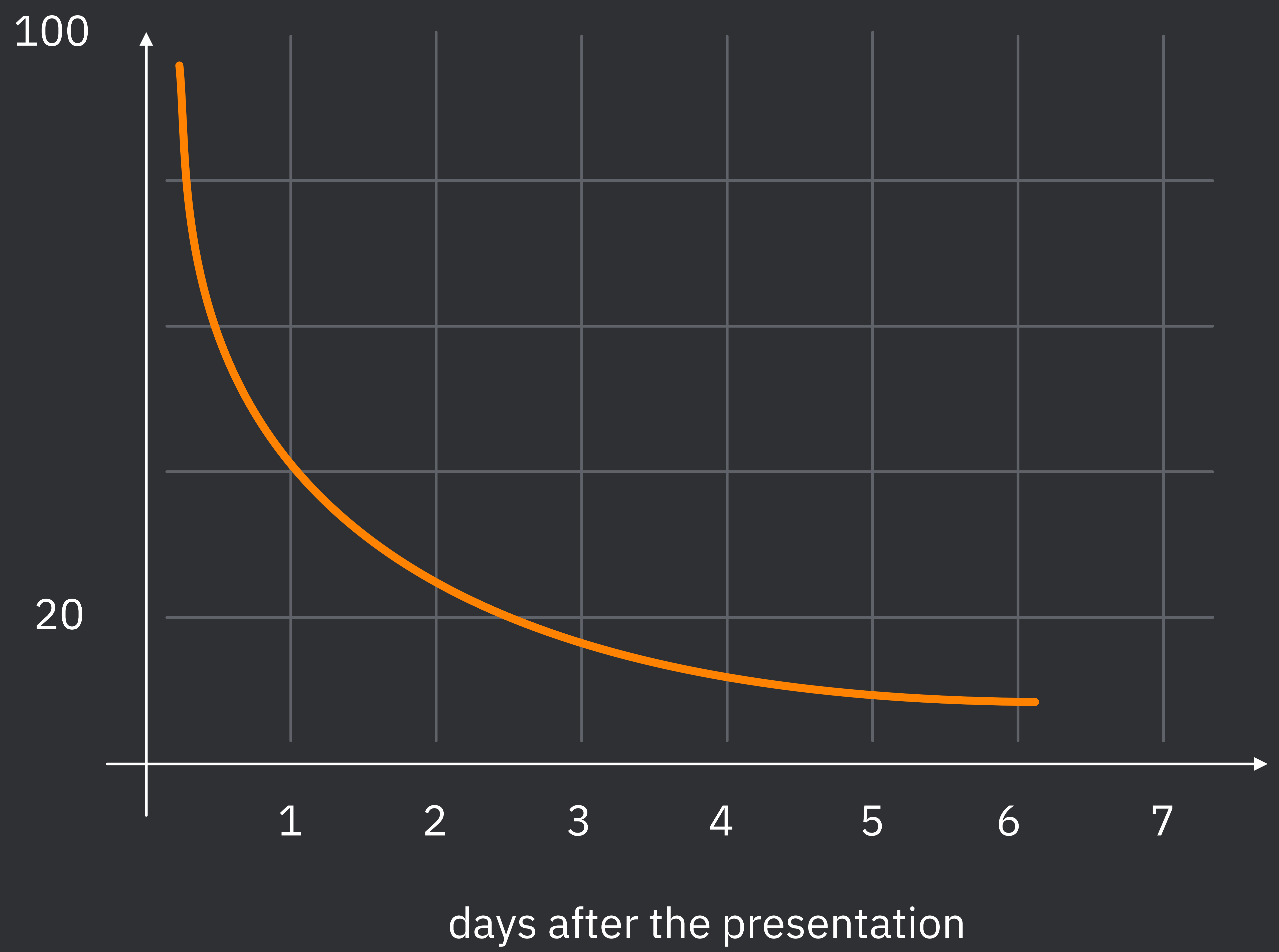
ExampleClass › createHtml()

Notes & Sharing

Notes & Sharing

How often do you take notes?

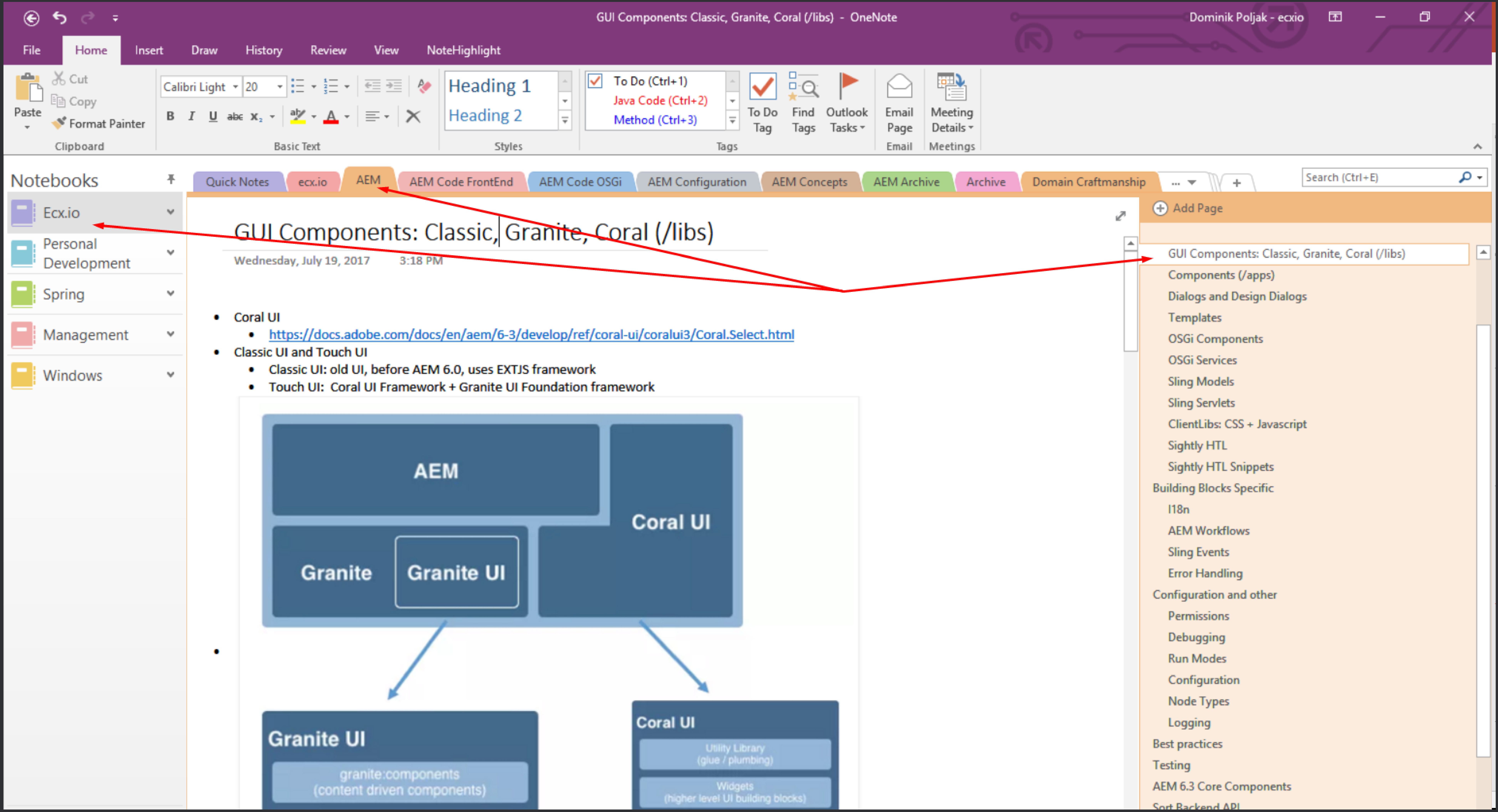
The forgetting Curve.



OneNote Features

- Search (even in screenshots)
- Cloud storage option
- Code highlighting (Highlight plugin)
- Shortcuts for reorganization, *Alt* + *Shift* + *Up* *or* *Down*

OneNote: notebooks, sections, pages, content



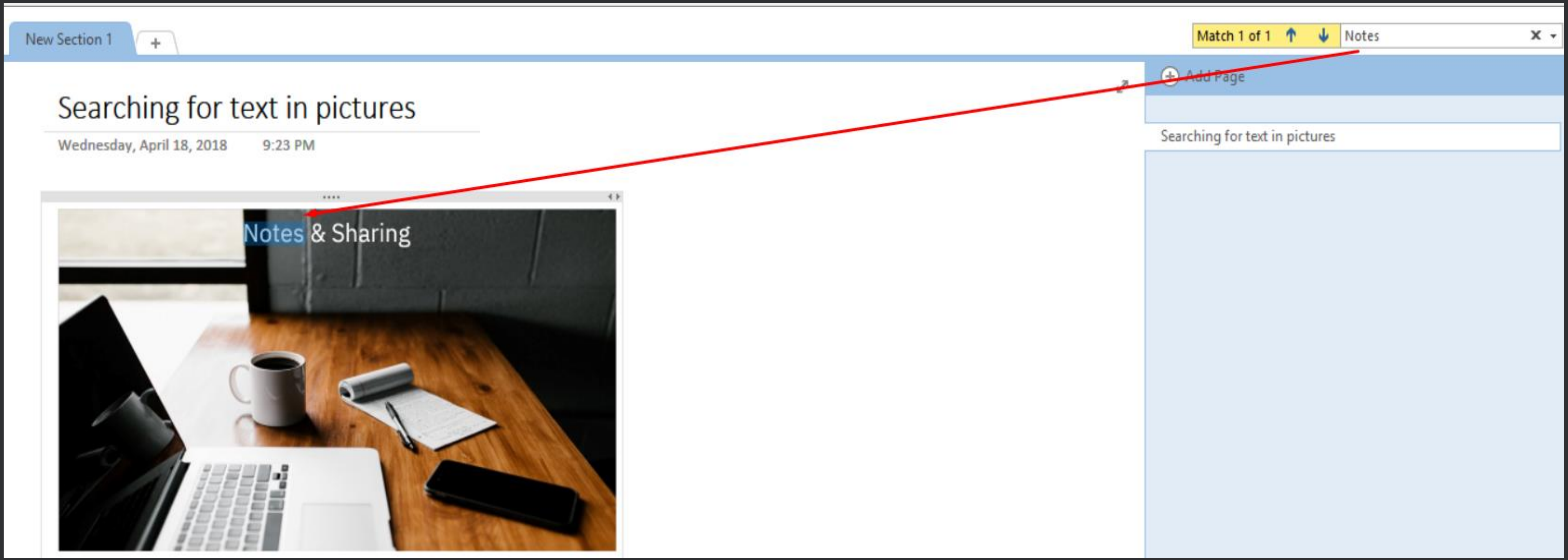
GUI Components: Classic, Granite, Coral (/libs)

Wednesday, July 19, 2017 3:18 PM

- Coral UI
  - https://docs.adobe.com/docs/en/aem/6-3/develop/ref/coral-ui/coralui3/Coral.Select.html
- Classic UI and Touch UI
  - Classic UI: old UI, before AEM 6.0, uses EXTJS framework
  - Touch UI: Coral UI Framework + Granite UI Foundation framework
-

GUI Components: Classic, Granite, Coral (/libs)
Components (/apps)
Dialogs and Design Dialogs
Templates
OSGi Components
OSGi Services
Sling Models
Sling Servlets
ClientLibs: CSS + Javascript
Sightly HTL
Sightly HTL Snippets
Building Blocks Specific
I18n
AEM Workflows
Sling Events
Error Handling
Configuration and other
Permissions
Debugging
Run Modes
Configuration
Node Types
Logging
Best practices
Testing
AEM 6.3 Core Components
Sort Backend API

OneNote: Search



Searching for text in pictures

Wednesday, April 18, 2018 9:23 PM

Sharing

How do you share information?
Text < Picture < Video

Screenshot tool: Lightshot

- Fast: *PrtSc* key
- Sharing: online or *Ctrl* + *C*
- Lightweight
- Simple editing options (arrows, text…)

Video recording tool: OBS Studio

- Free, open source
- No limits on time
- Simple
- Option to capture screen or window
- 1 hour of video with audio ≈ 100 MB

NAVIGATION TO FREQUENTLY USED FILES

Launchy: suggesting *Alt* + *X*

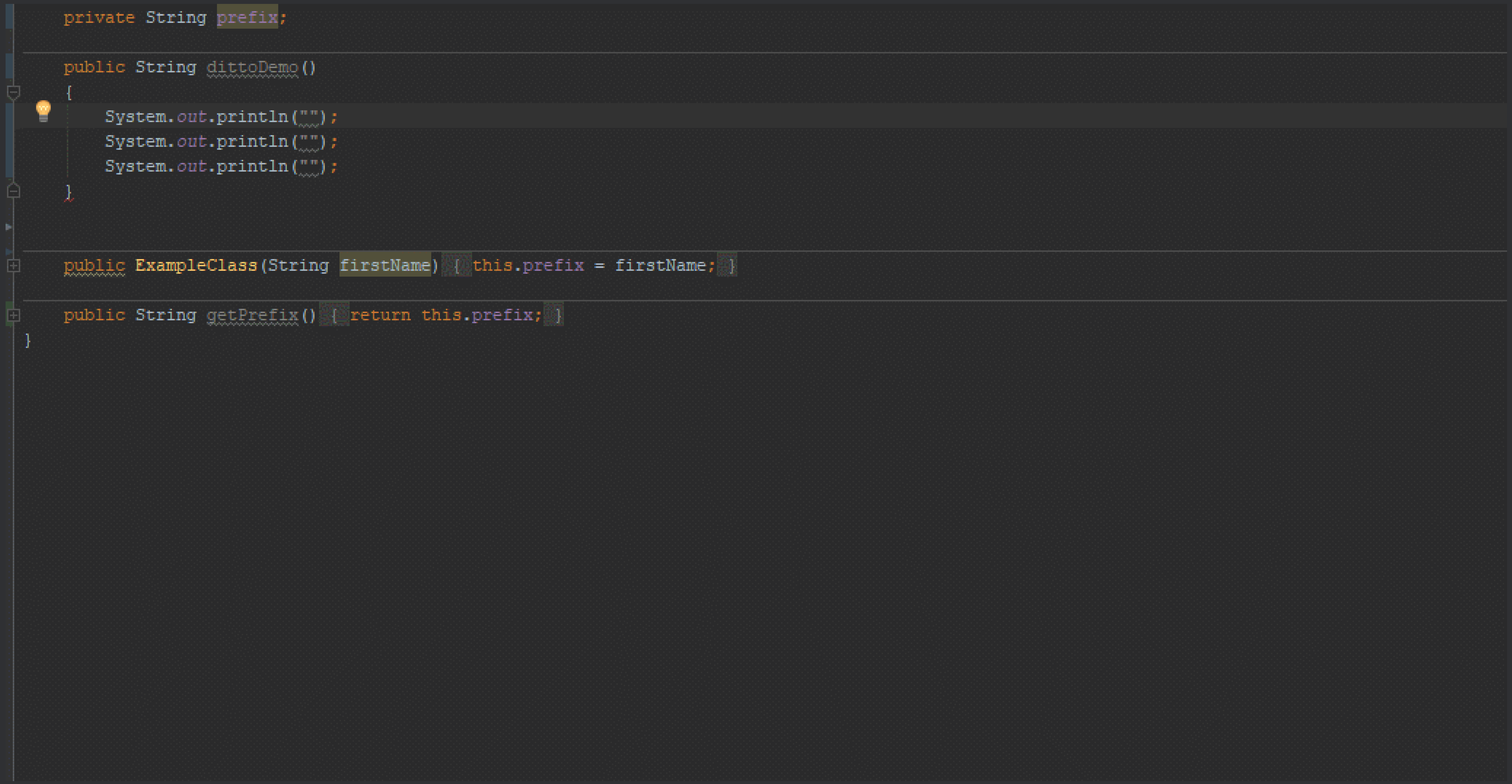
Copy/Paste

Leverage everything you copy!

Clipboard Manager: Ditto

- Fast: *Ctrl* + *Alt* + ` (suggestion)
- Preview for pictures *F3*
- Searchable
- History: 100+ copied values (configurable)
- Advanced options: paste multiple values (multicursors)

Ditto



```
    private String prefix;

    public String dittoDemo()
    {
        System.out.println("");
        System.out.println("");
        System.out.println("");
    }

    public ExampleClass(String firstName) { this.prefix = firstName; }

    public String getPrefix() { return this.prefix; }
}
```

Automation

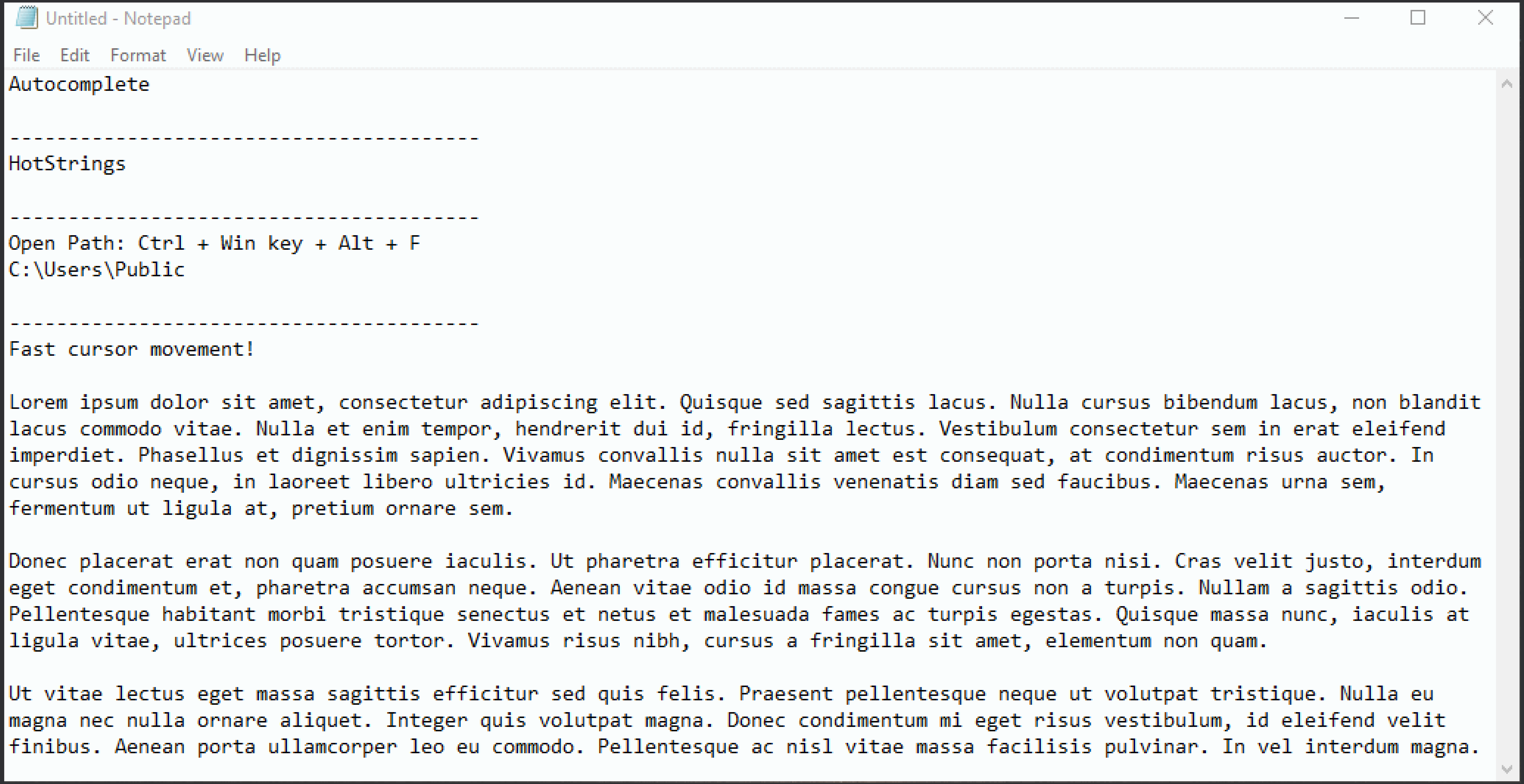
One tool to tool them all?

AUTOHOTKEY

Autohotkey

- Global hotkeys – works everywhere!
- Combining tool shortcuts
- Hot strings
- Autocomplete
- Custom scripts

Autohotkey



```
Autocomplete

----------------------------------------
HotStrings

----------------------------------------
Open Path: Ctrl + Win key + Alt + F
C:\Users\Public

----------------------------------------
Fast cursor movement!
```

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Quisque sed sagittis lacus. Nulla cursus bibendum lacus, non blandit lacus commodo vitae. Nulla et enim tempor, hendrerit dui id, fringilla lectus. Vestibulum consectetur sem in erat eleifend imperdiet. Phasellus et dignissim sapien. Vivamus convallis nulla sit amet est consequat, at condimentum risus auctor. In cursus odio neque, in laoreet libero ultricies id. Maecenas convallis venenatis diam sed faucibus. Maecenas urna sem, fermentum ut ligula at, pretium ornare sem.

Donec placerat erat non quam posuere iaculis. Ut pharetra efficitur placerat. Nunc non porta nisi. Cras velit justo, interdum eget condimentum et, pharetra accumsan neque. Aenean vitae odio id massa congue cursus non a turpis. Nullam a sagittis odio. Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas. Quisque massa nunc, iaculis at ligula vitae, ultrices posuere tortor. Vivamus risus nibh, cursus a fringilla sit amet, elementum non quam.

Ut vitae lectus eget massa sagittis efficitur sed quis felis. Praesent pellentesque neque ut volutpat tristique. Nulla eu magna nec nulla ornare aliquet. Integer quis volutpat magna. Donec condimentum mi eget risus vestibulum, id eleifend velit finibus. Aenean porta ullamcorper leo eu commodo. Pellentesque ac nisl vitae massa facilisis pulvinar. In vel interdum magna.

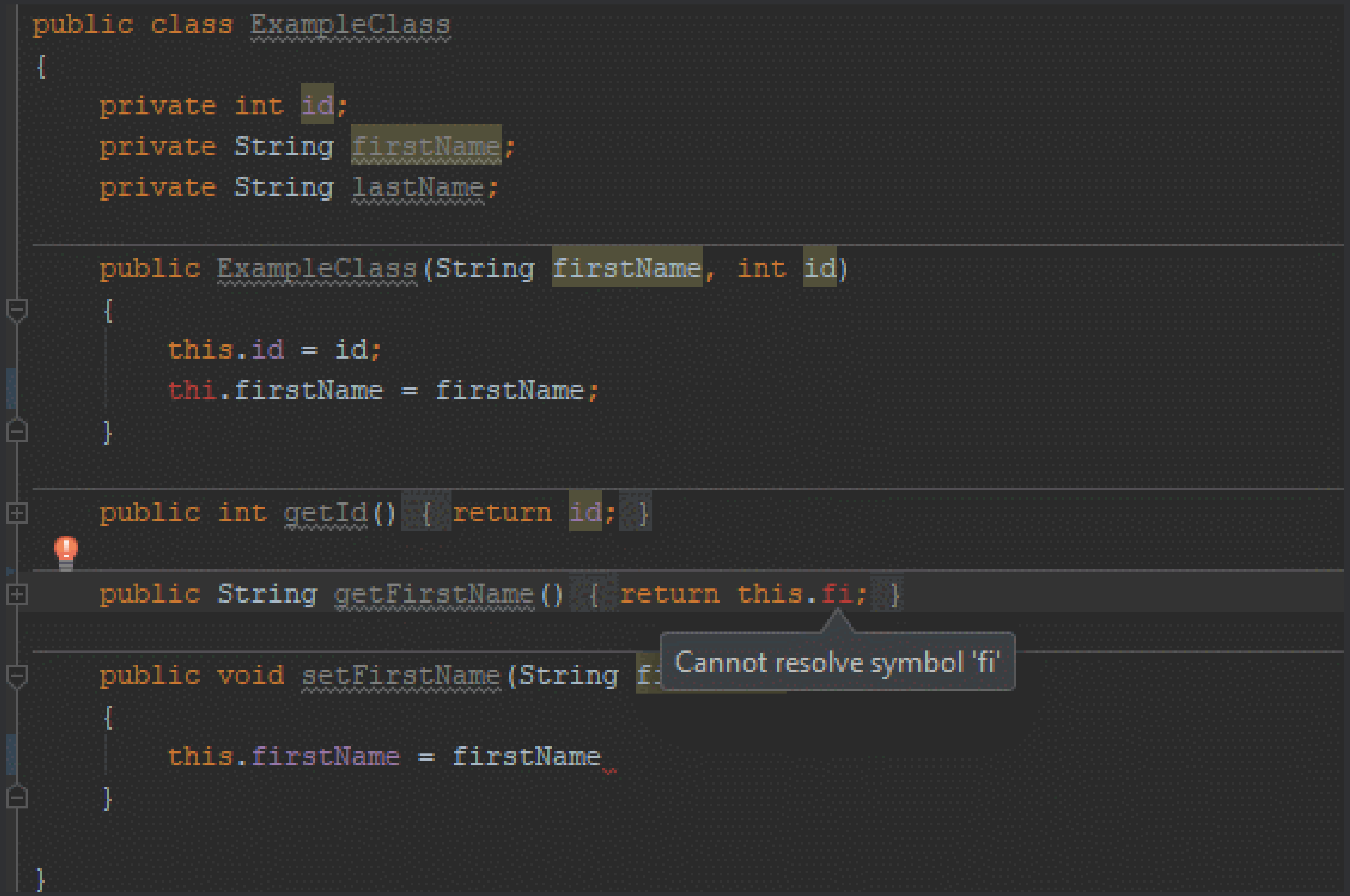
ecx.io
an IBM Company

Thank you.

Questions?

Backup slides

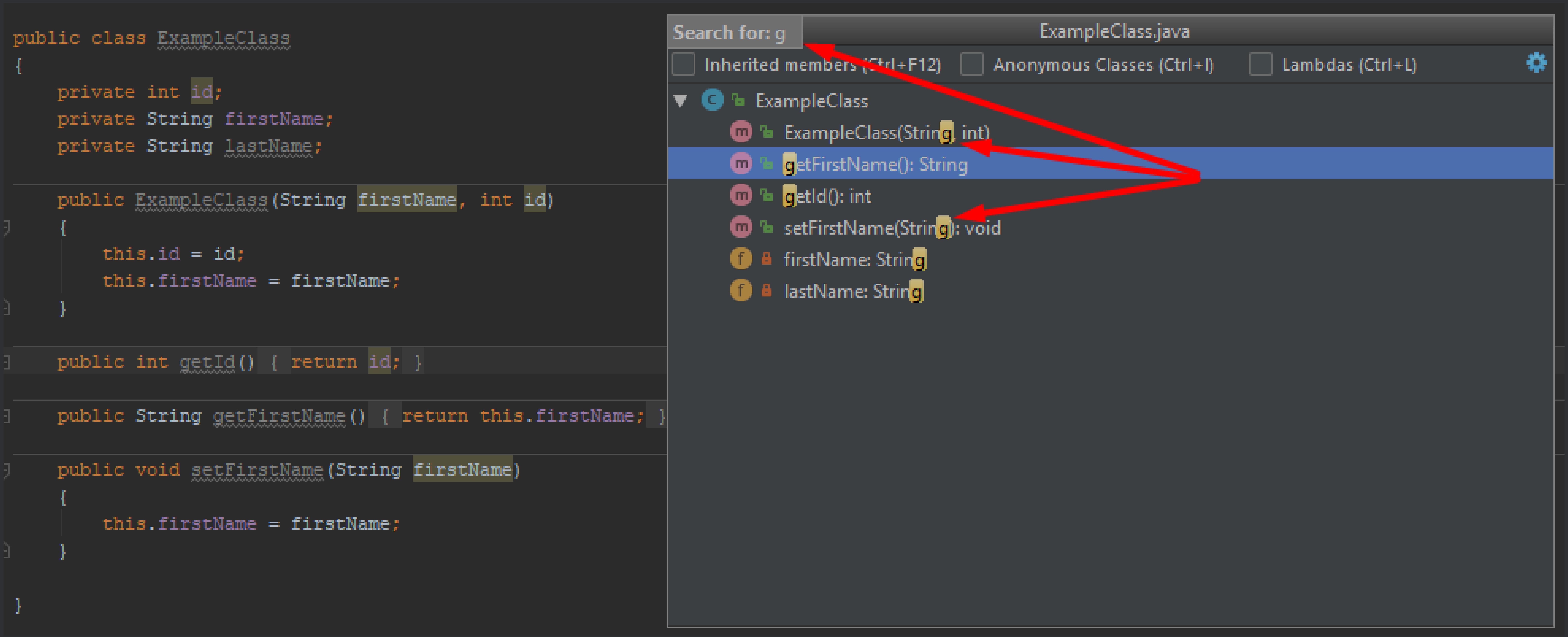
Navigation WITHIN file: moving over errors *F2*



```
public class ExampleClass
{
    private int id;
    private String firstName;
    private String lastName;

    public ExampleClass(String firstName, int id)
    {
        this.id = id;
        thi.firstName = firstName;
    }

    public int getId() { return id; }

    public String getFirstName() { return this.fi; }

    public void setFirstName(String fi
    {
        this.firstName = firstName
    }

}
```

Cannot resolve symbol 'fi'

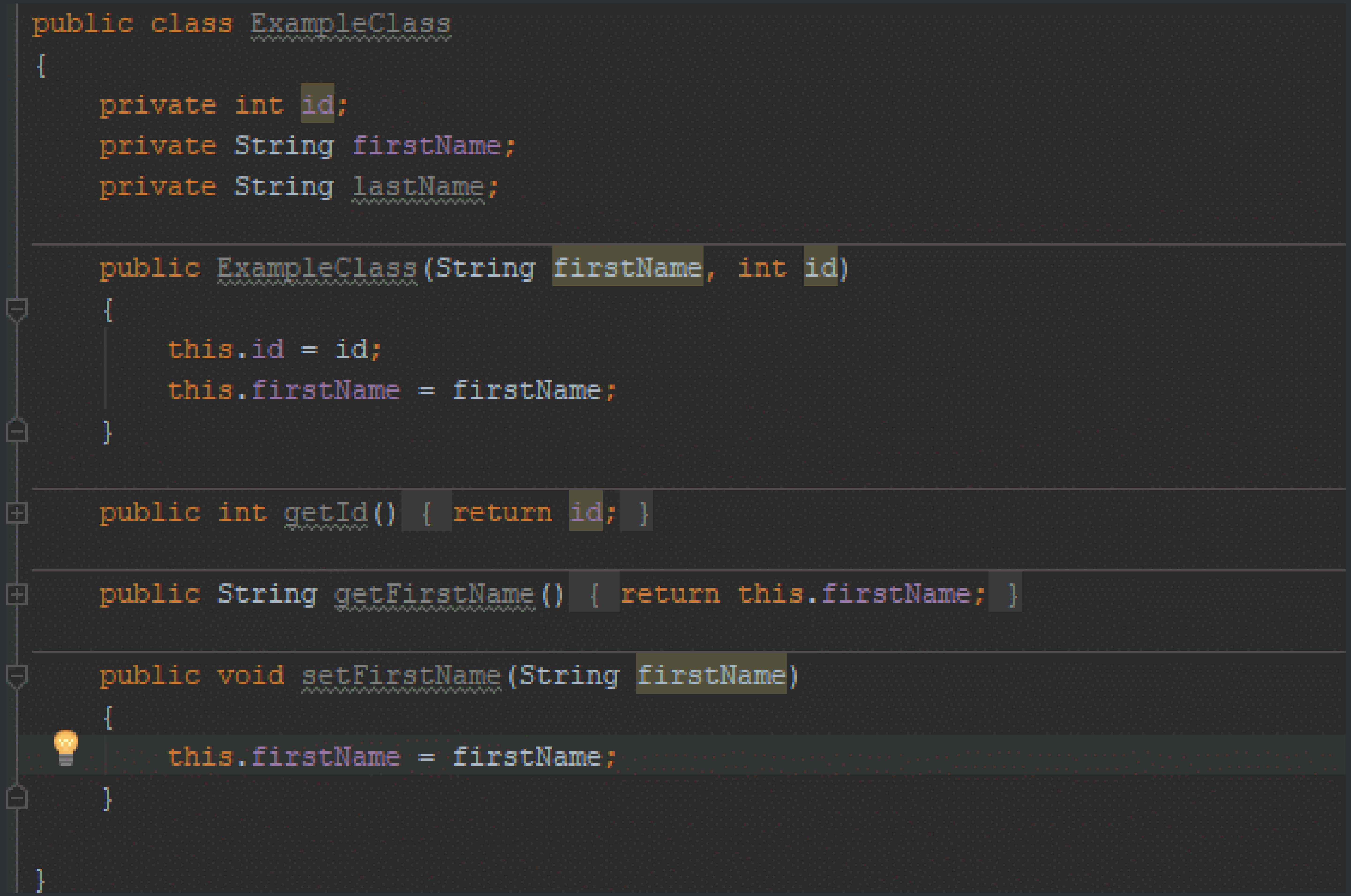
Navigation WITHIN file: navigate to class member *Ctrl* + *F12*



```
public class ExampleClass
{
    private int id;
    private String firstName;
    private String lastName;

    public ExampleClass(String firstName, int id)
    {
        this.id = id;
        this.firstName = firstName;
    }

    public int getId() { return id; }

    public String getFirstName() { return this.firstName; }

    public void setFirstName(String firstName)
    {
        this.firstName = firstName;
    }

}
```

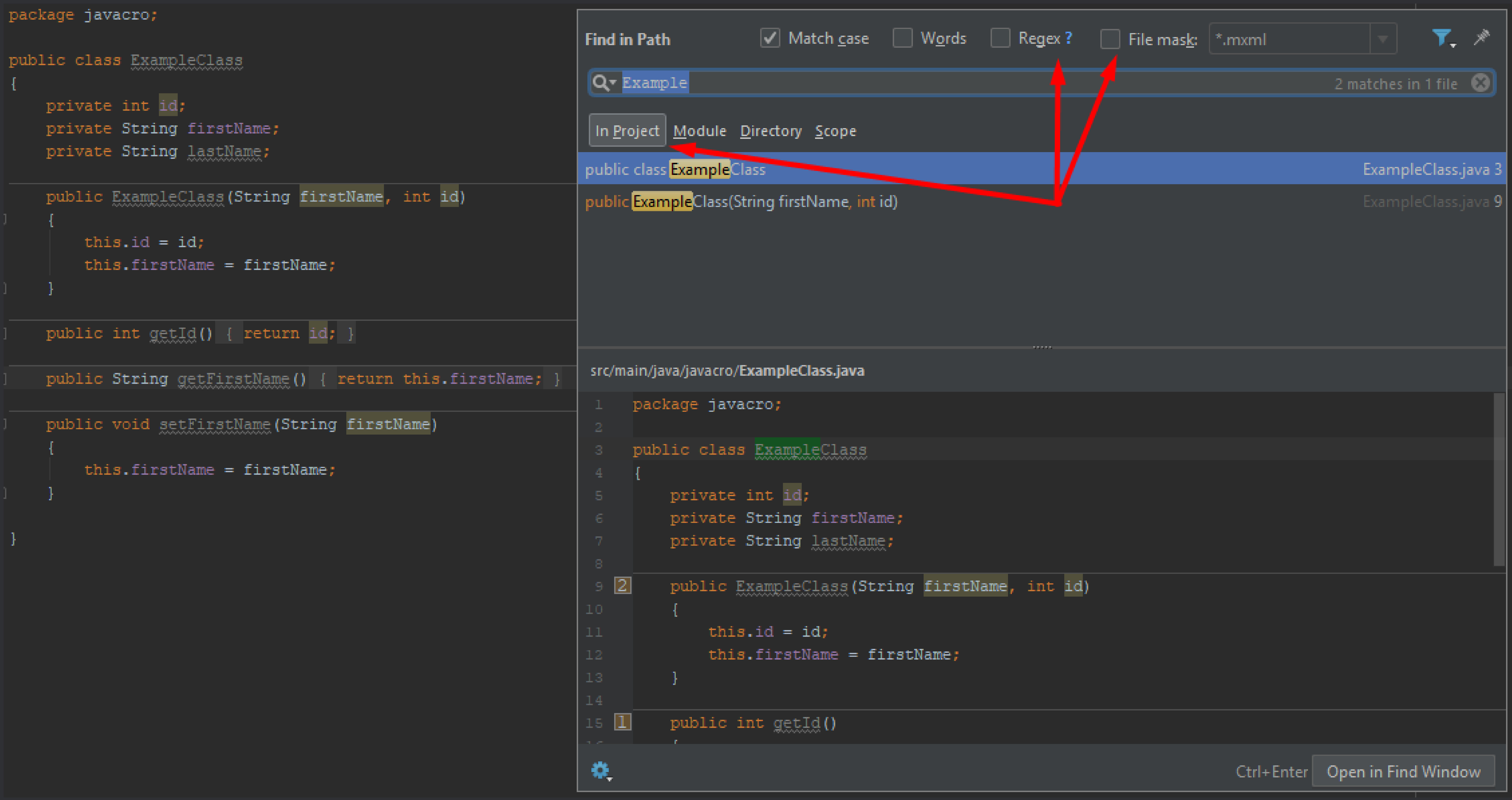
```
Search for: g                ExampleClass.java
[ ] Inherited members (Ctrl+F12)   [ ] Anonymous Classes (Ctrl+I)   [ ] Lambdas (Ctrl+L)
▼ ExampleClass
    ExampleClass(String, int)
    getFirstName(): String
    getId(): int
    setFirstName(String): void
    firstName: String
    lastName: String
```

Navigation WITHIN file: moving between methods *Alt* + *Up* *or* *Down*



```java
public class ExampleClass
{
    private int id;
    private String firstName;
    private String lastName;

    public ExampleClass(String firstName, int id)
    {
        this.id = id;
        this.firstName = firstName;
    }

    public int getId() { return id; }

    public String getFirstName() { return this.firstName; }

    public void setFirstName(String firstName)
    {
        this.firstName = firstName;
    }

}
```

Find file *Ctrl* + *Shift* + *F*



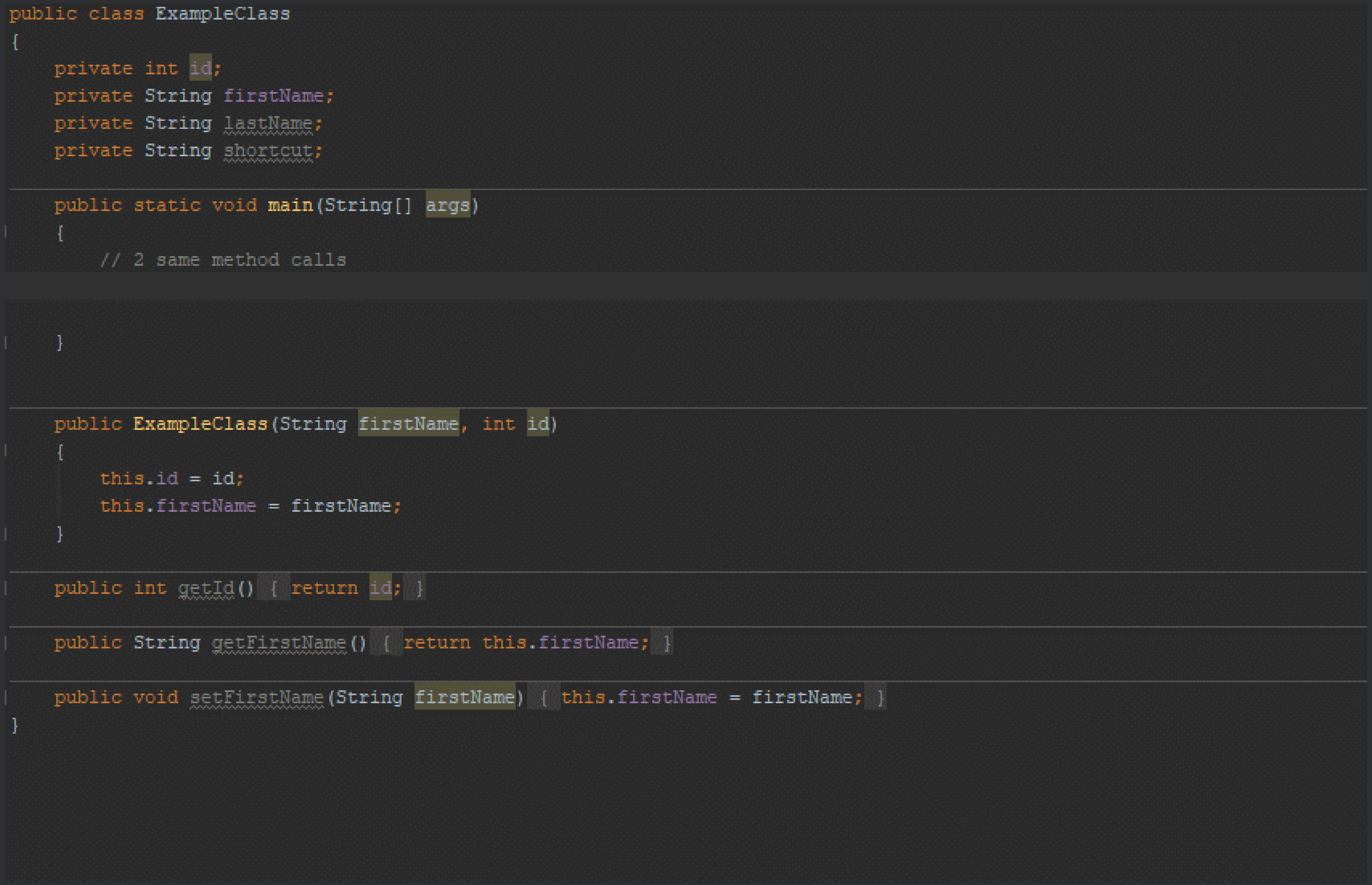
```java
package javacro;

public class ExampleClass
{
    private int id;
    private String firstName;
    private String lastName;

    public ExampleClass(String firstName, int id)
    {
        this.id = id;
        this.firstName = firstName;
    }

    public int getId() { return id; }

    public String getFirstName() { return this.firstName; }

    public void setFirstName(String firstName)
    {
        this.firstName = firstName;
    }

}
```

```
public class ExampleClass                              ExampleClass.java 3
public ExampleClass(String firstName, int id)          ExampleClass.java 9
```

src/main/java/javacro/**ExampleClass.java**

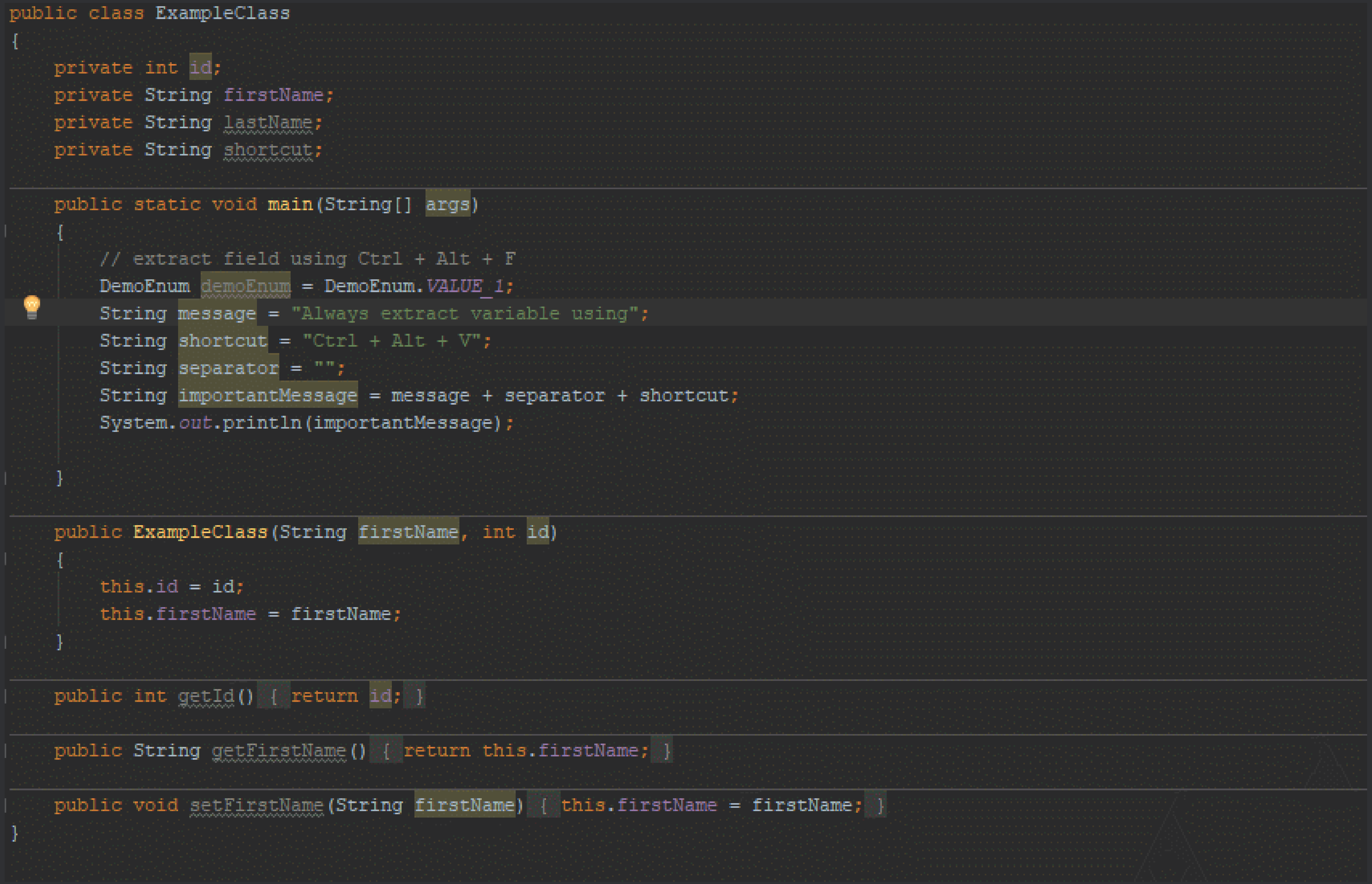
```java
package javacro;

public class ExampleClass
{
    private int id;
    private String firstName;
    private String lastName;

    public ExampleClass(String firstName, int id)
    {
        this.id = id;
        this.firstName = firstName;
    }

    public int getId()
```

Extract Parameter: *Ctrl* + *Alt* + *P*



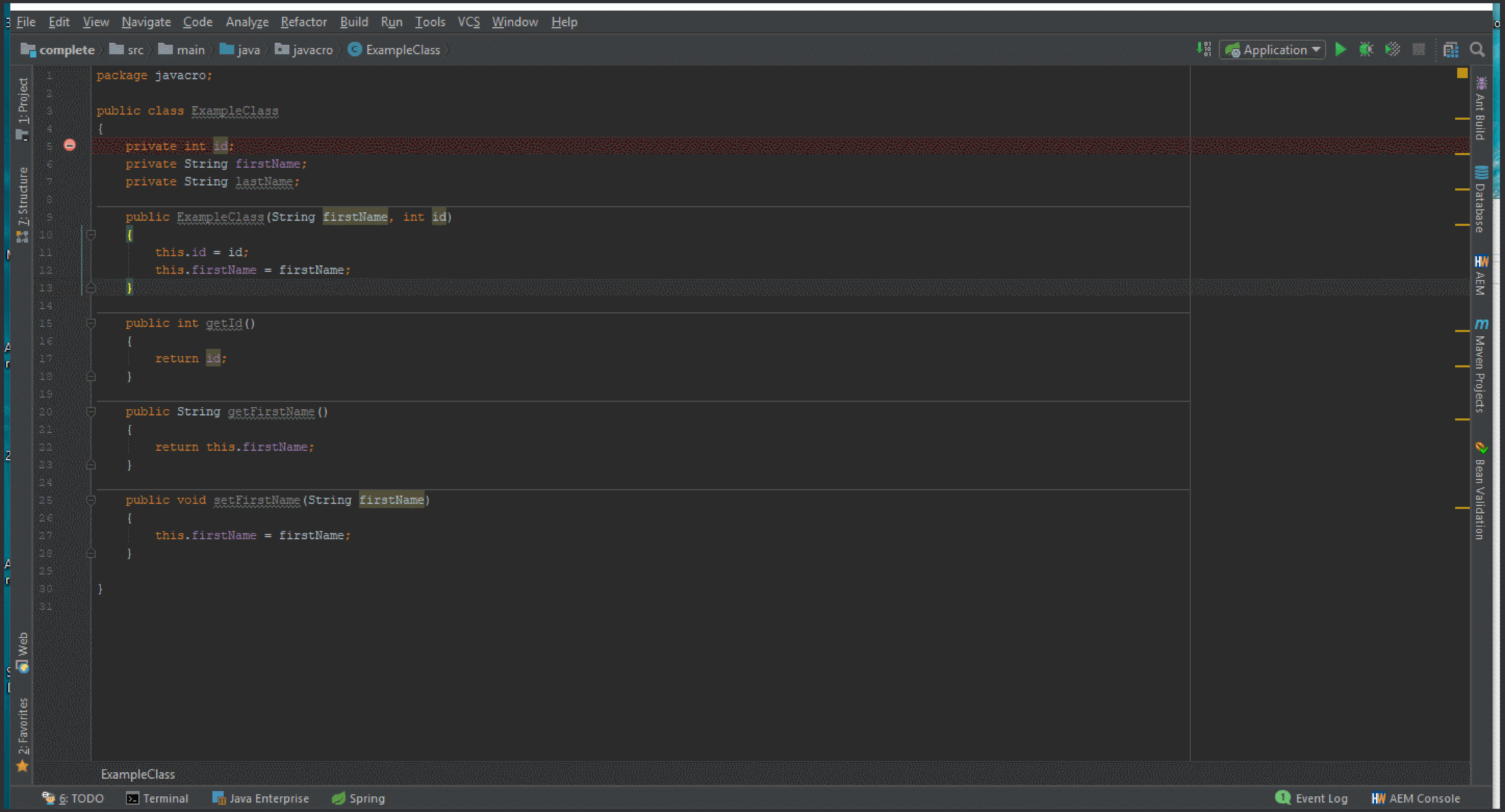
```
public class ExampleClass
{
    private int id;
    private String firstName;
    private String lastName;
    private String shortcut;

    public static void main(String[] args)
    {
        // 2 same method calls


    }


    public ExampleClass(String firstName, int id)
    {
        this.id = id;
        this.firstName = firstName;
    }

    public int getId() { return id; }

    public String getFirstName() { return this.firstName; }

    public void setFirstName(String firstName) { this.firstName = firstName; }
}
```

Extract Field: *Ctrl* + *Alt* + *F*



```
public class ExampleClass
{
    private int id;
    private String firstName;
    private String lastName;
    private String shortcut;

    public static void main(String[] args)
    {
        // extract field using Ctrl + Alt + F
        DemoEnum demoEnum = DemoEnum.VALUE_1;
        String message = "Always extract variable using";
        String shortcut = "Ctrl + Alt + V";
        String separator = "";
        String importantMessage = message + separator + shortcut;
        System.out.println(importantMessage);

    }

    public ExampleClass(String firstName, int id)
    {
        this.id = id;
        this.firstName = firstName;
    }

    public int getId() { return id; }

    public String getFirstName() { return this.firstName; }

    public void setFirstName(String firstName) { this.firstName = firstName; }
}
```

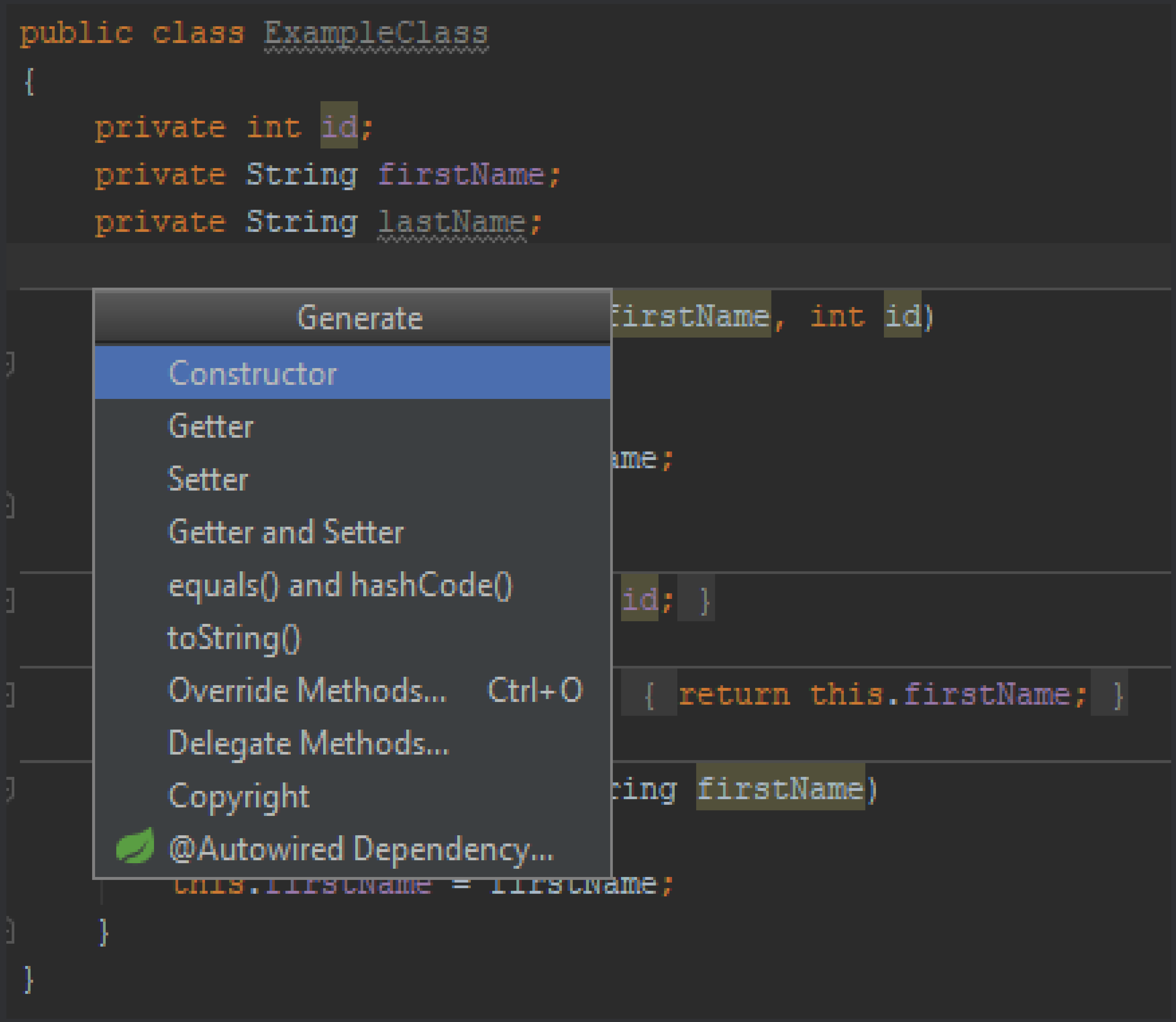
Terminal *Alt* + *F12*



```java
package javacro;

public class ExampleClass
{
    private int id;
    private String firstName;
    private String lastName;

    public ExampleClass(String firstName, int id)
    {
        this.id = id;
        this.firstName = firstName;
    }

    public int getId()
    {
        return id;
    }

    public String getFirstName()
    {
        return this.firstName;
    }

    public void setFirstName(String firstName)
    {
        this.firstName = firstName;
    }

}
```

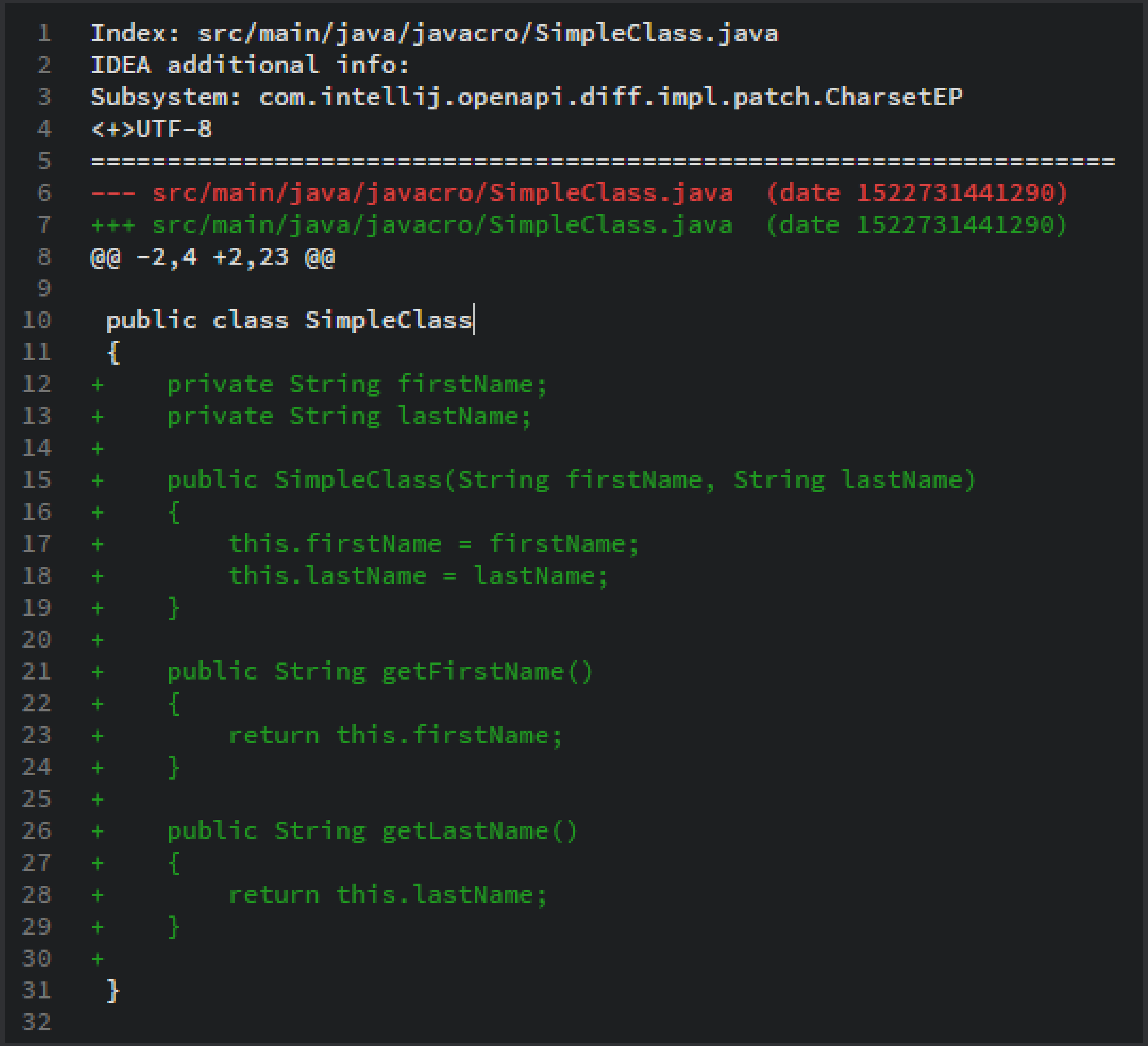
Code Generation *Alt* + *Insert*

- Constructors
- Getters
- Setters
- Equals() and hashCode()
- toString()



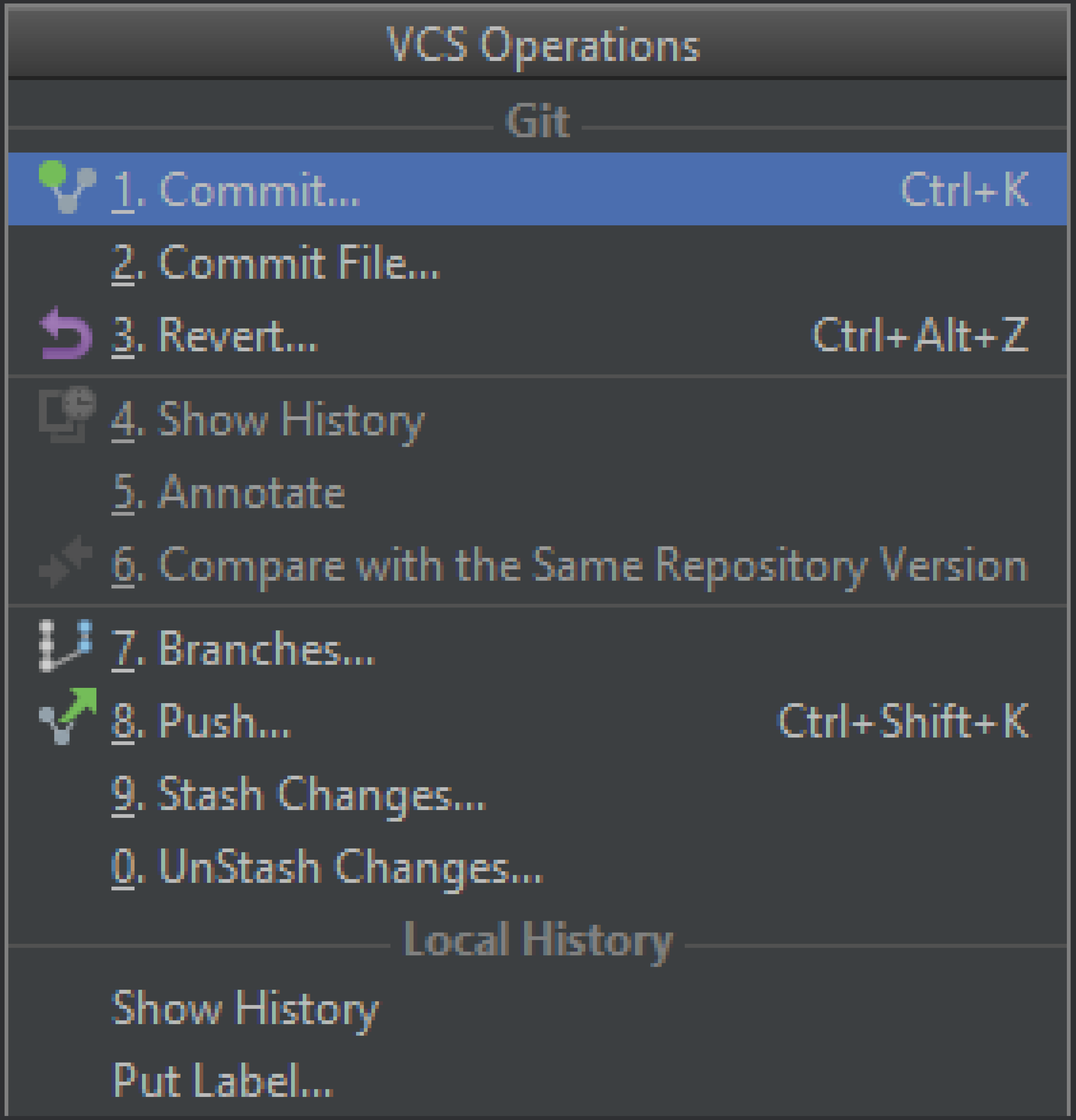
```
public class ExampleClass
{
    private int id;
    private String firstName;
    private String lastName;

                        irstName, int id)

                        ame;

                        id; }

                        { return this.firstName; }

                        ing firstName)
            this.firstName = firstName;
    }
}
```

Local History: Patches



```
Index: src/main/java/javacro/SimpleClass.java
IDEA additional info:
Subsystem: com.intellij.openapi.diff.impl.patch.CharsetEP
<+>UTF-8
===================================================================
--- src/main/java/javacro/SimpleClass.java  (date 1522731441290)
+++ src/main/java/javacro/SimpleClass.java  (date 1522731441290)
@@ -2,4 +2,23 @@

 public class SimpleClass
 {
+    private String firstName;
+    private String lastName;
+
+    public SimpleClass(String firstName, String lastName)
+    {
+        this.firstName = firstName;
+        this.lastName = lastName;
+    }
+
+    public String getFirstName()
+    {
+        return this.firstName;
+    }
+
+    public String getLastName()
+    {
+        return this.lastName;
+    }
+
 }

```

Git integration *Alt* + `



VCS Operations

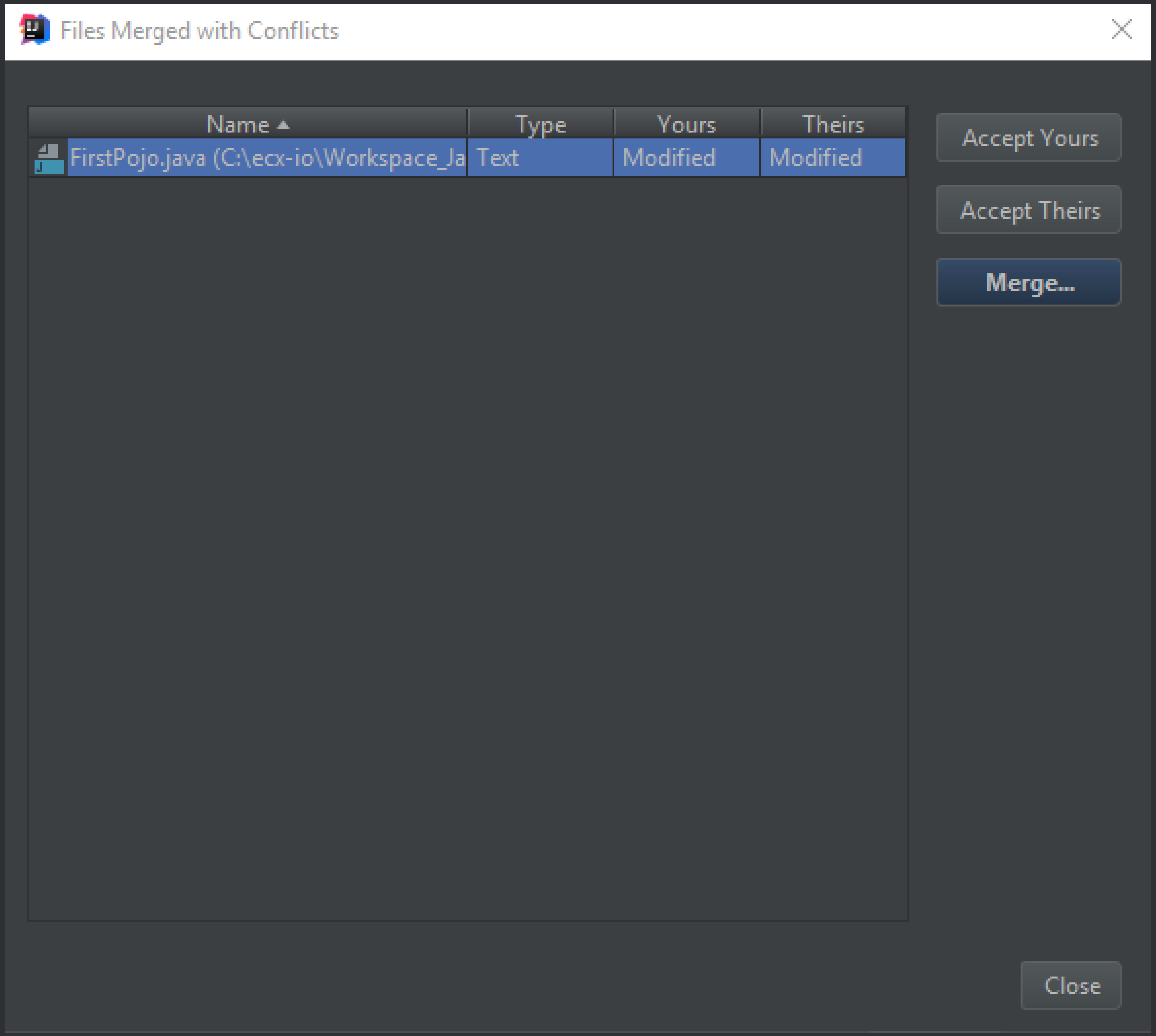
Git

1. Commit... Ctrl+K
2. Commit File...
3. Revert... Ctrl+Alt+Z
4. Show History
5. Annotate
6. Compare with the Same Repository Version
7. Branches...
8. Push... Ctrl+Shift+K
9. Stash Changes...
0. UnStash Changes...

Local History

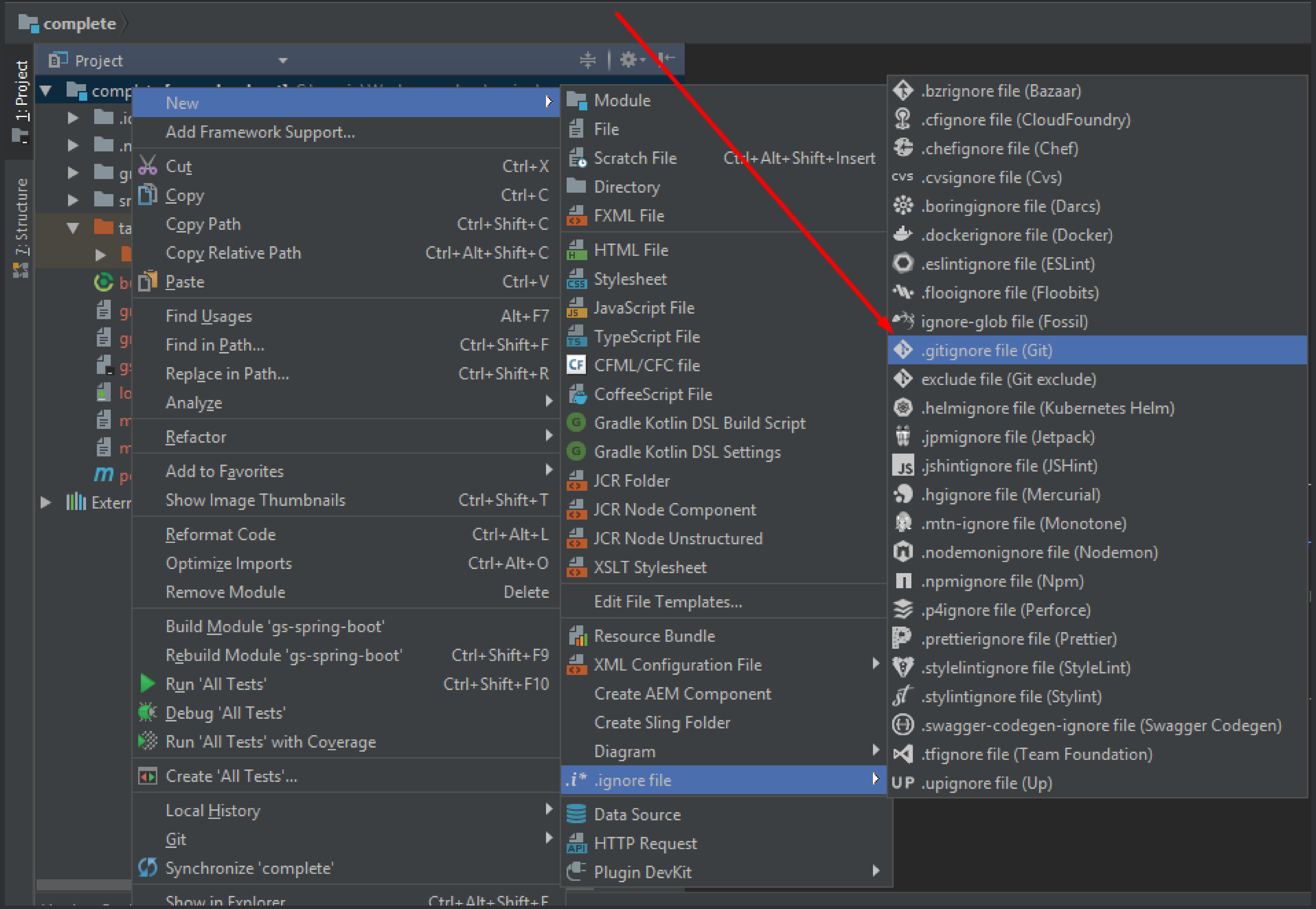
Show History
Put Label...

Git integration: Conflict Resolution


Files Merged with Conflicts

Name	Type	Yours	Theirs
FirstPojo.java (C:\ecx-io\Workspace_Ja	Text	Modified	Modified

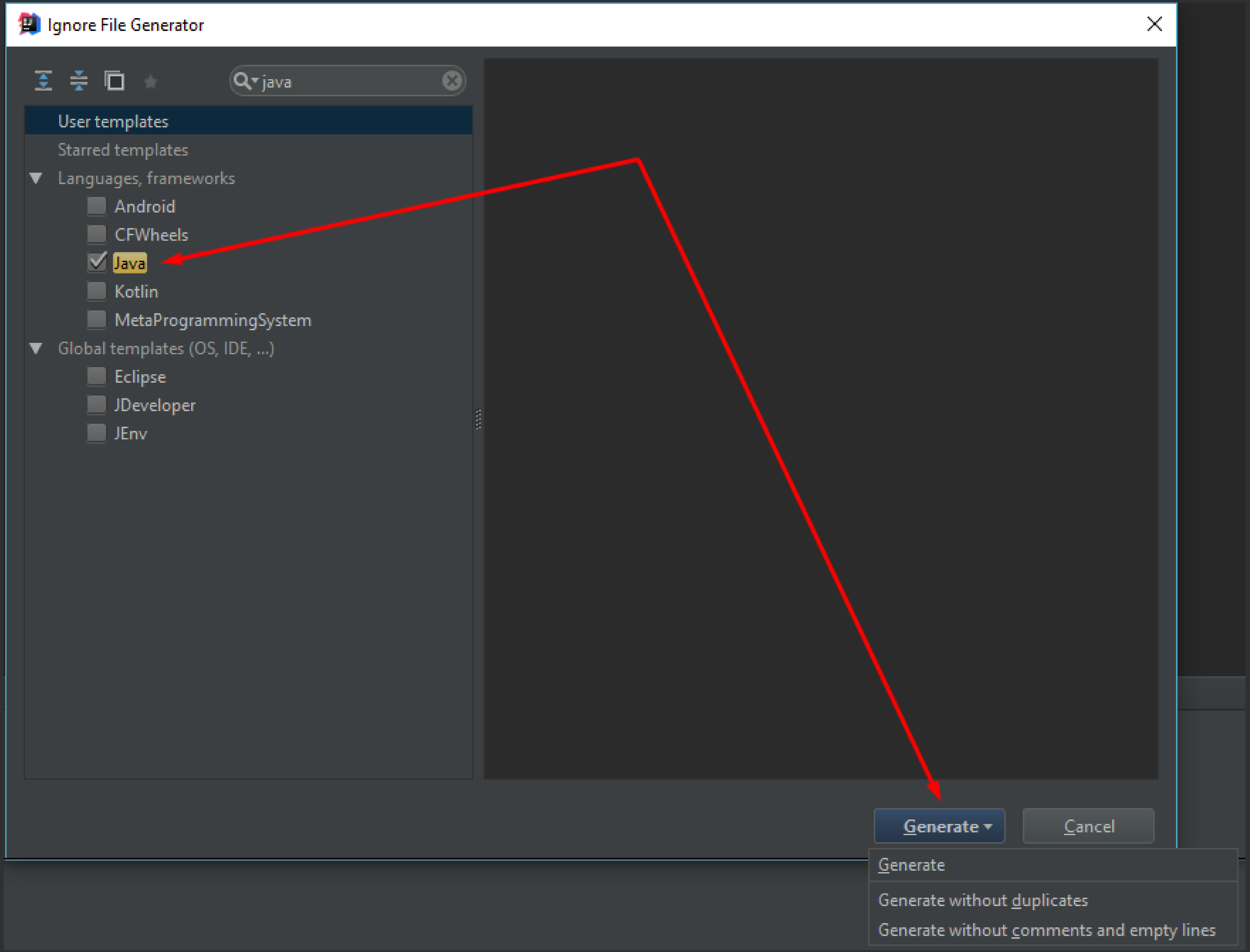
Git .ignore plugin


complete
Project
New
Add Framework Support...
Cut Ctrl+X
Copy Ctrl+C
Copy Path Ctrl+Shift+C
Copy Relative Path Ctrl+Alt+Shift+C
Paste Ctrl+V
Find Usages Alt+F7
Find in Path... Ctrl+Shift+F
Replace in Path... Ctrl+Shift+R
Analyze
Refactor
Add to Favorites
Show Image Thumbnails Ctrl+Shift+T
Reformat Code Ctrl+Alt+L
Optimize Imports Ctrl+Alt+O
Remove Module Delete
Build Module 'gs-spring-boot'
Rebuild Module 'gs-spring-boot' Ctrl+Shift+F9
Run 'All Tests' Ctrl+Shift+F10
Debug 'All Tests'
Run 'All Tests' with Coverage
Create 'All Tests'...
Local History
Git
Synchronize 'complete'
Show in Explorer Ctrl+Alt+Shift+F
Module
File
Scratch File Ctrl+Alt+Shift+Insert
Directory
FXML File
HTML File
Stylesheet
JavaScript File
TypeScript File
CFML/CFC file
CoffeeScript File
Gradle Kotlin DSL Build Script
Gradle Kotlin DSL Settings
JCR Folder
JCR Node Component
JCR Node Unstructured
XSLT Stylesheet
Edit File Templates...
Resource Bundle
XML Configuration File
Create AEM Component
Create Sling Folder
Diagram
.ignore file
Data Source
HTTP Request
Plugin DevKit
.bzrignore file (Bazaar)
.cfignore file (CloudFoundry)
.chefignore file (Chef)
.cvsignore file (Cvs)
.boringignore file (Darcs)
.dockerignore file (Docker)
.eslintignore file (ESLint)
.flooignore file (Floobits)
ignore-glob file (Fossil)
.gitignore file (Git)
exclude file (Git exclude)
.helmignore file (Kubernetes Helm)
.jpmignore file (Jetpack)
.jshintignore file (JSHint)
.hgignore file (Mercurial)
.mtn-ignore file (Monotone)
.nodemonignore file (Nodemon)
.npmignore file (Npm)
.p4ignore file (Perforce)
.prettierignore file (Prettier)
.stylelintignore file (StyleLint)
.stylintignore file (Stylint)
.swagger-codegen-ignore file (Swagger Codegen)
.tfignore file (Team Foundation)
.upignore file (Up)

Git .ignore plugin: features

- Templates
- Coloring ignored files in the Project view
- Fix actions
  - Removing tracked files which match patterns added to .gitignore file
  - Adding unversioned files to .gitignore file

Git .ignore plugin: templates


Ignore File Generator

java

User templates
Starred templates
Languages, frameworks
- [ ] Android
- [ ] CFWheels
- [x] Java
- [ ] Kotlin
- [ ] MetaProgrammingSystem

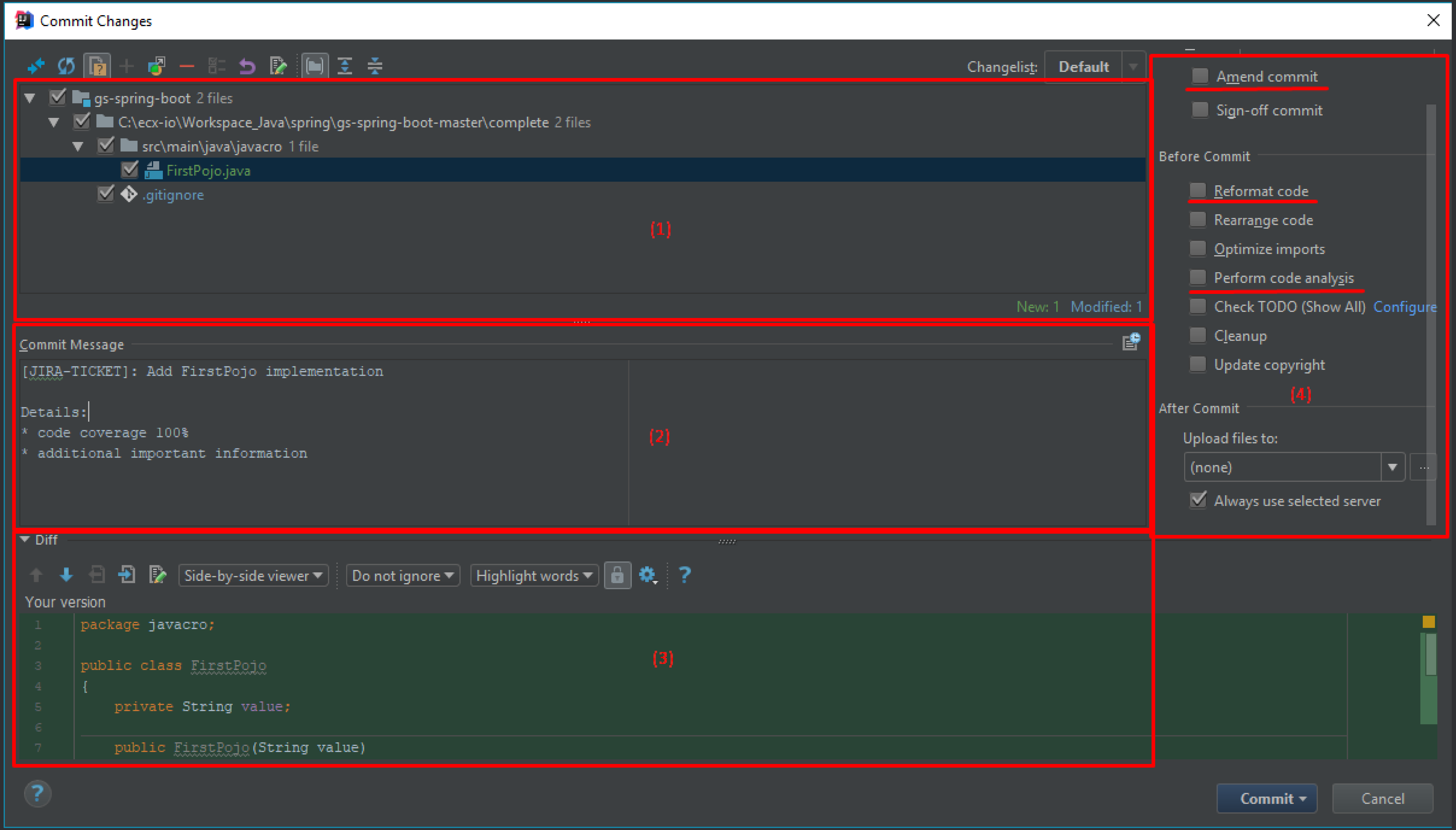
Global templates (OS, IDE, ...)
- [ ] Eclipse
- [ ] JDeveloper
- [ ] JEnv

Generate
Cancel
Generate
Generate without duplicates
Generate without comments and empty lines

Git Integration: Features

- Intuitive commit + push *Ctrl + K*
- Easy conflict resolution
- Simple stash and unstash

Git integration: Commit + Push, *Ctrl* + *K*



Commit Changes

Changelist: Default

- [✓] gs-spring-boot 2 files
  - [✓] C:\ecx-io\Workspace_Java\spring\gs-spring-boot-master\complete 2 files
    - [✓] src\main\java\javacro 1 file
      - [✓] FirstPojo.java
    - [✓] .gitignore

(1)

New: 1 Modified: 1

Commit Message

```
[JIRA-TICKET]: Add FirstPojo implementation

Details:
* code coverage 100%
* additional important information
```

(2)

Diff

Side-by-side viewer | Do not ignore | Highlight words

Your version

```
package javacro;

public class FirstPojo
{
    private String value;

    public FirstPojo(String value)
```

(3)

- [ ] Amend commit
- [ ] Sign-off commit

Before Commit

- [ ] Reformat code
- [ ] Rearrange code
- [ ] Optimize imports
- [ ] Perform code analysis
- [ ] Check TODO (Show All) Configure
- [ ] Cleanup
- [ ] Update copyright

(4)

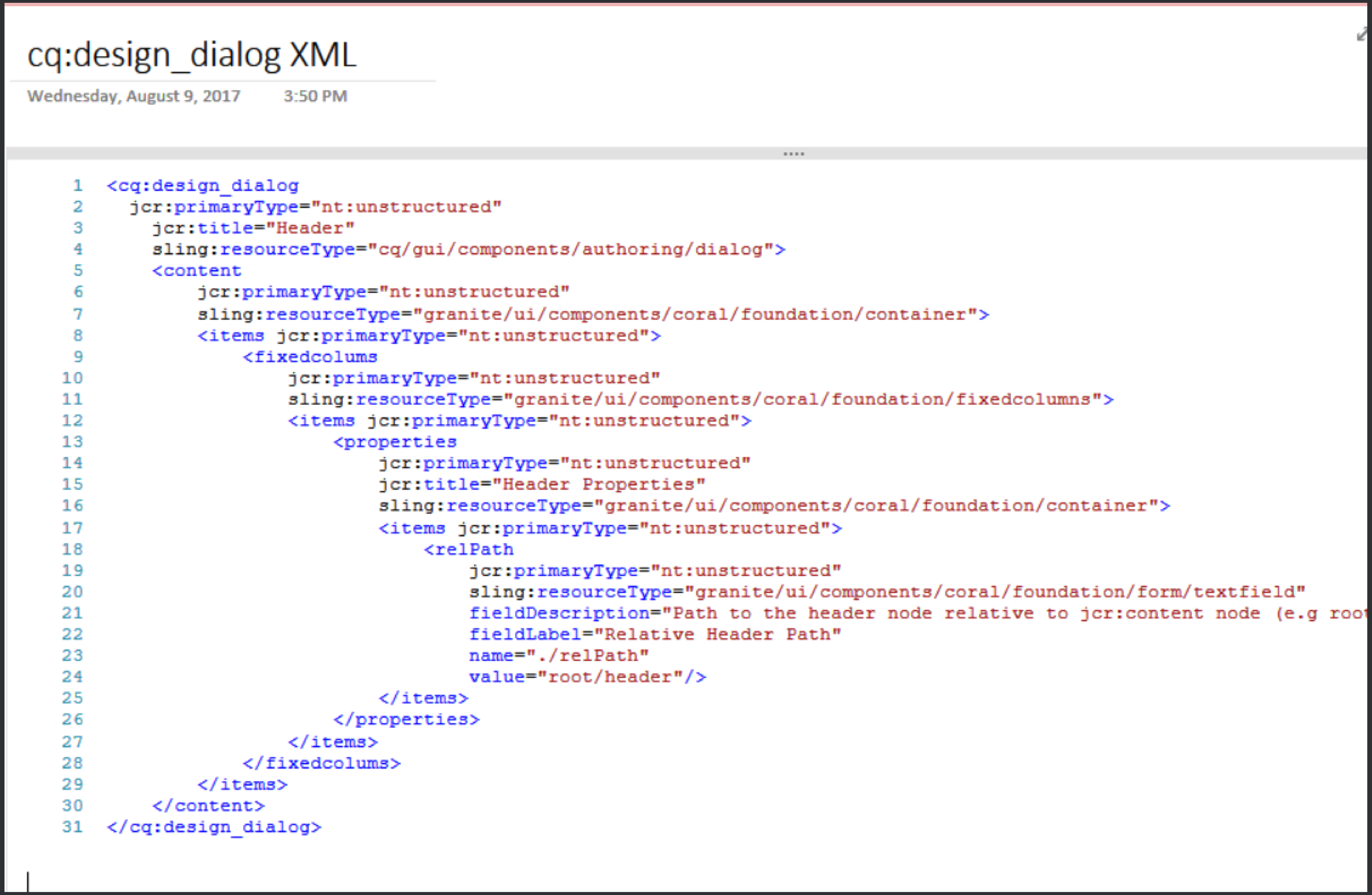
After Commit

Upload files to: (none)

- [✓] Always use selected server

Commit | Cancel

OneNote: NoteHighlight plugin



cq:design_dialog XML

Wednesday, August 9, 2017 3:50 PM

```
<cq:design_dialog
  jcr:primaryType="nt:unstructured"
    jcr:title="Header"
    sling:resourceType="cq/gui/components/authoring/dialog">
    <content
        jcr:primaryType="nt:unstructured"
        sling:resourceType="granite/ui/components/coral/foundation/container">
        <items jcr:primaryType="nt:unstructured">
            <fixedcolums
                jcr:primaryType="nt:unstructured"
                sling:resourceType="granite/ui/components/coral/foundation/fixedcolumns">
                <items jcr:primaryType="nt:unstructured">
                    <properties
                        jcr:primaryType="nt:unstructured"
                        jcr:title="Header Properties"
                        sling:resourceType="granite/ui/components/coral/foundation/container">
                        <items jcr:primaryType="nt:unstructured">
                            <relPath
                                jcr:primaryType="nt:unstructured"
                                sling:resourceType="granite/ui/components/coral/foundation/form/textfield"
                                fieldDescription="Path to the header node relative to jcr:content node (e.g roo
                                fieldLabel="Relative Header Path"
                                name="./relPath"
                                value="root/header"/>
                        </items>
                    </properties>
                </items>
            </fixedcolums>
        </items>
    </content>
</cq:design_dialog>
```

Questions for continuous improvement

- Can the number of **mouse clicks** and **keyboard presses** be reduced even more?
- Has someone already developed a simple solution for our problem?