The IntelliJ IDEA Blog

IntelliJ IDEA – the Leading Java and Kotlin IDE, by JetBrains

Download Follow

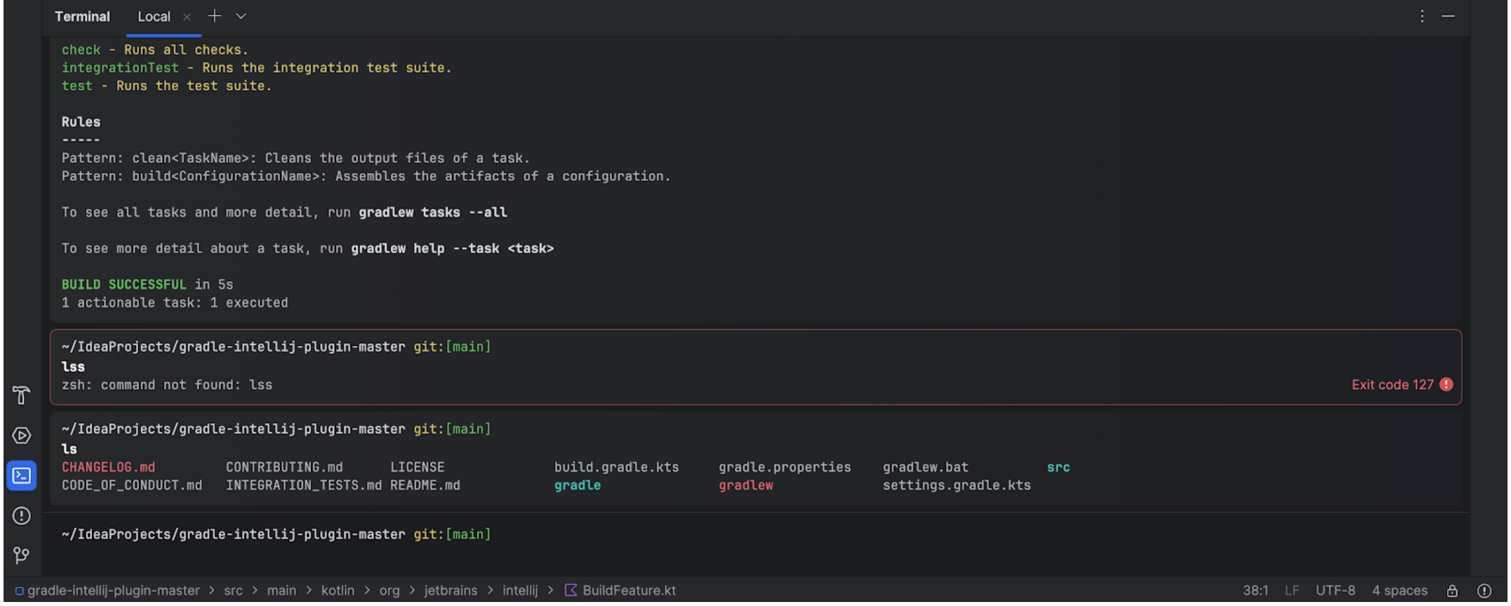
All News Releases Livestreams Tips & Tricks Early Access Program Plugin

IntelliJ IDEA News

The New Terminal (Beta) Is Now in JetBrains IDEs

Anastasia Shabalina
February 21, 2024

TL;DR: The new terminal (Beta) is now in JetBrains IDEs and will be available in the new UI starting from the 2024.1 version of the IDEs. You can switch between the old and new terminal in *Settings/Preferences | Tools | Terminal | Enable New Terminal*. Currently, it supports only Bash, Zsh, and PowerShell, with other shells in development. This blog post covers the architectural and visual changes and outlines our plans.



```
check - Runs all checks.
integrationTest - Runs the integration test suite.
test - Runs the test suite.

Rules
-----
Pattern: clean<TaskName>: Cleans the output files of a task.
Pattern: build<ConfigurationName>: Assembles the artifacts of a configuration.

To see all tasks and more detail, run gradlew tasks --all

To see more detail about a task, run gradlew help --task <task>

BUILD SUCCESSFUL in 5s
1 actionable task: 1 executed
```

```
~/IdeaProjects/gradle-intellij-plugin-master git:[main]
lss
zsh: command not found: lss                                              Exit code 127
```

```
~/IdeaProjects/gradle-intellij-plugin-master git:[main]
ls
CHANGELOG.md       CONTRIBUTING.md      LICENSE              build.gradle.kts     gradle.properties    gradlew.bat          src
CODE_OF_CONDUCT.md INTEGRATION_TESTS.md README.md            gradle               gradlew              settings.gradle.kts
```

```
~/IdeaProjects/gradle-intellij-plugin-master git:[main]
```

gradle-intellij-plugin-master > src > main > kotlin > org > jetbrains > intellij > BuildFeature.kt
38:1
LF
UTF-8
4 spaces

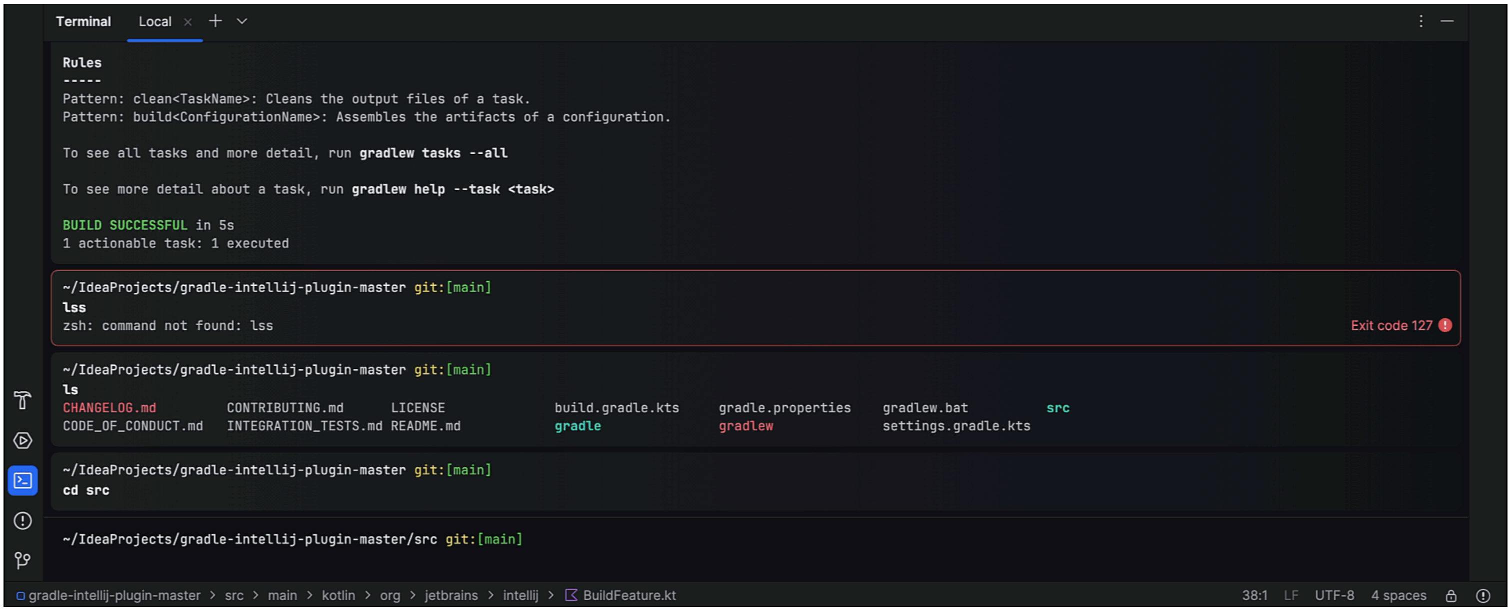
Software developers use terminals for a variety of tasks. Though it’s a basic element of programming, the convenience of a terminal lies in its ability to provide a universal way of managing tasks without needing to be proficient in all UI elements. Modern solutions, on the other hand, come with numerous productivity enhancements, such as smart layouts for the command prompt and output, command completion with a list of options, AI-powered functions, and many others.

Aiming to combine both basic and advanced capabilities, the new terminal is a familiar tool presented in a fresh way and with an expanded set of features. Our goals are to preserve the functionality of working in a terminal, enhance convenience and simplicity, and address long-standing issues. This article will guide you through the changes to the terminal in JetBrains IDEs and our plans for the future.

Visual changes

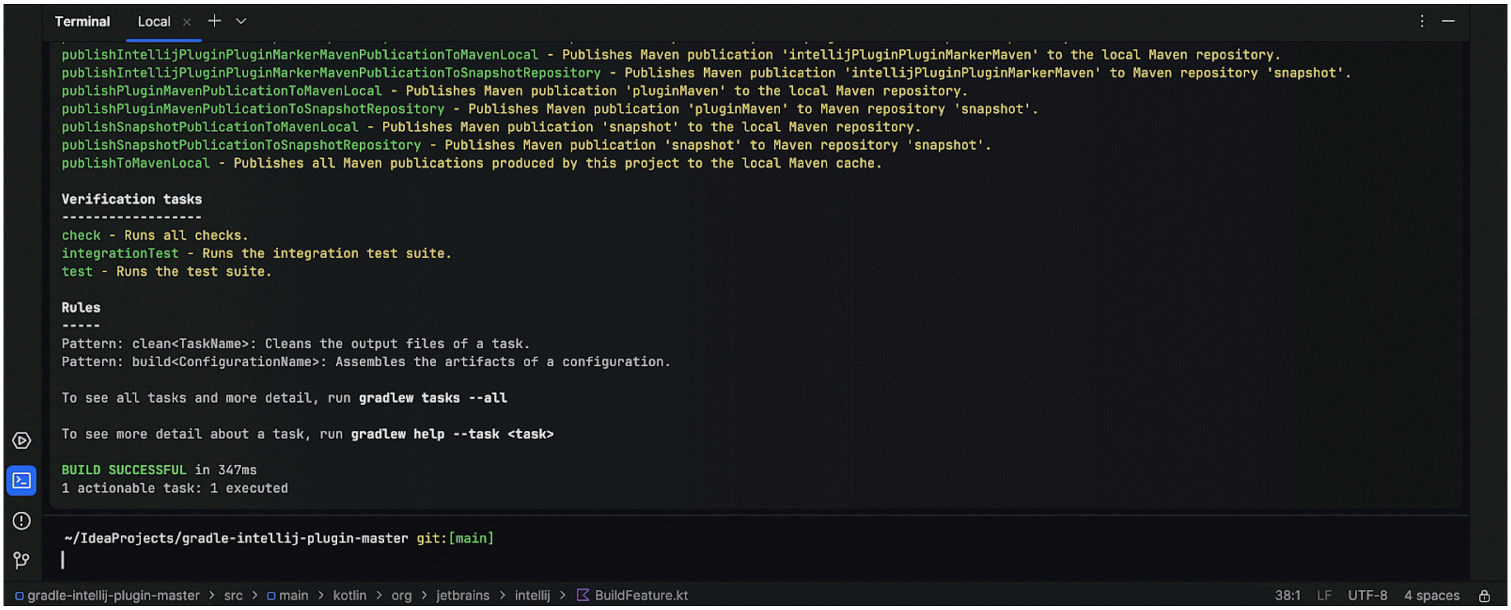
The new terminal aligns with the look and feel of the new UI and is available in all JetBrains IDEs starting from version 2024.1. The primary visual distinction of the new terminal is that each command is now presented in a separate block. This helps you quickly locate the start and end of each command, enhancing the readability of the output. This change brings other new capabilities, such as easy navigation between blocks using arrows and the *⌘↑ / ⌘↓* (macOS) and *Ctrl+↑ /*

Ctrl+↓ (Windows and Linux) shortcuts that enable you to easily switch between the prompt and output.


Terminal
Local
Exit code 127
gradle-intellij-plugin-master > src > main > kotlin > org > jetbrains > intellij > BuildFeature.kt
38:1 LF UTF-8 4 spaces

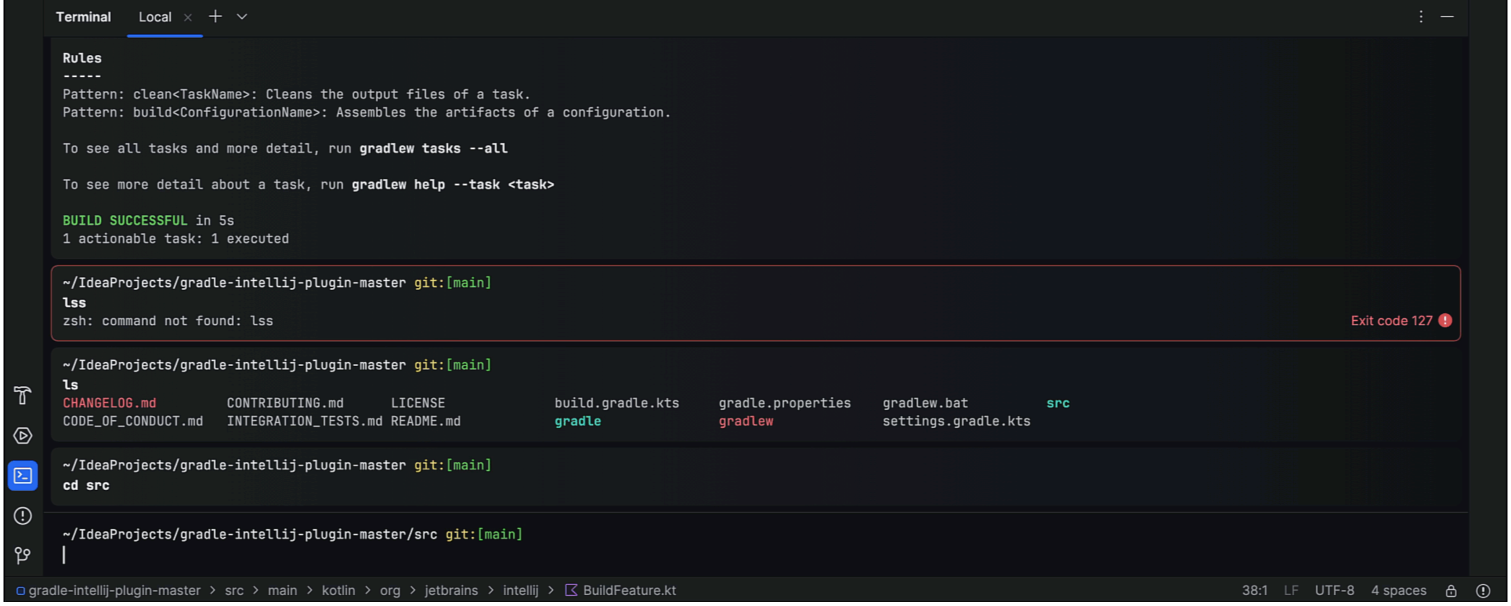
```
Rules
-----
Pattern: clean<TaskName>: Cleans the output files of a task.
Pattern: build<ConfigurationName>: Assembles the artifacts of a configuration.

To see all tasks and more detail, run gradlew tasks --all

To see more detail about a task, run gradlew help --task <task>

BUILD SUCCESSFUL in 5s
1 actionable task: 1 executed

~/IdeaProjects/gradle-intellij-plugin-master git:[main]
lss
zsh: command not found: lss

~/IdeaProjects/gradle-intellij-plugin-master git:[main]
ls
CHANGELOG.md        CONTRIBUTING.md      LICENSE              build.gradle.kts     gradle.properties    gradlew.bat          src
CODE_OF_CONDUCT.md  INTEGRATION_TESTS.md README.md            gradle               gradlew              settings.gradle.kts

~/IdeaProjects/gradle-intellij-plugin-master git:[main]
cd src

~/IdeaProjects/gradle-intellij-plugin-master/src git:[main]
```

Another significant change is the command completion feature that supports commands, paths, arguments, and options, aimed at simplifying command entry.


Terminal
Local
gradle-intellij-plugin-master > src > main > kotlin > org > jetbrains > intellij > BuildFeature.kt
38:1 LF UTF-8 4 spaces

```
publishIntellijPluginPluginMarkerMavenPublicationToMavenLocal - Publishes Maven publication 'intellijPluginPluginMarkerMaven' to the local Maven repository.
publishIntellijPluginPluginMarkerMavenPublicationToSnapshotRepository - Publishes Maven publication 'intellijPluginPluginMarkerMaven' to Maven repository 'snapshot'.
publishPluginMavenPublicationToMavenLocal - Publishes Maven publication 'pluginMaven' to the local Maven repository.
publishPluginMavenPublicationToSnapshotRepository - Publishes Maven publication 'pluginMaven' to Maven repository 'snapshot'.
publishSnapshotPublicationToMavenLocal - Publishes Maven publication 'snapshot' to the local Maven repository.
publishSnapshotPublicationToSnapshotRepository - Publishes Maven publication 'snapshot' to Maven repository 'snapshot'.
publishToMavenLocal - Publishes all Maven publications produced by this project to the local Maven cache.

Verification tasks
------------------
check - Runs all checks.
integrationTest - Runs the integration test suite.
test - Runs the test suite.

Rules
-----
Pattern: clean<TaskName>: Cleans the output files of a task.
Pattern: build<ConfigurationName>: Assembles the artifacts of a configuration.

To see all tasks and more detail, run gradlew tasks --all

To see more detail about a task, run gradlew help --task <task>

BUILD SUCCESSFUL in 347ms
1 actionable task: 1 executed

~/IdeaProjects/gradle-intellij-plugin-master git:[main]
|
```

A new, convenient command history has been added, which allows for filtering and makes it easier to navigate through recently executed commands. The UX designers have also revamped the color scheme – the palette is now aesthetically pleasing and easy to read.


Terminal
Local
gradle-intellij-plugin-master > src > main > kotlin > org > jetbrains > intellij > BuildFeature.kt
38:1
LF
UTF-8
4 spaces

```
Rules
-----
Pattern: clean<TaskName>: Cleans the output files of a task.
Pattern: build<ConfigurationName>: Assembles the artifacts of a configuration.

To see all tasks and more detail, run gradlew tasks --all

To see more detail about a task, run gradlew help --task <task>

BUILD SUCCESSFUL in 5s
1 actionable task: 1 executed

~/IdeaProjects/gradle-intellij-plugin-master git:[main]
lss
zsh: command not found: lss                                                          Exit code 127

~/IdeaProjects/gradle-intellij-plugin-master git:[main]
ls
CHANGELOG.md       CONTRIBUTING.md      LICENSE           build.gradle.kts    gradle.properties   gradlew.bat          src
CODE_OF_CONDUCT.md INTEGRATION_TESTS.md README.md         gradle              gradlew             settings.gradle.kts

~/IdeaProjects/gradle-intellij-plugin-master git:[main]
cd src

~/IdeaProjects/gradle-intellij-plugin-master/src git:[main]
|
```

Architectural changes

The new terminal is not just visual changes, but a new architecture that made these changes possible. Here are the most notable changes:

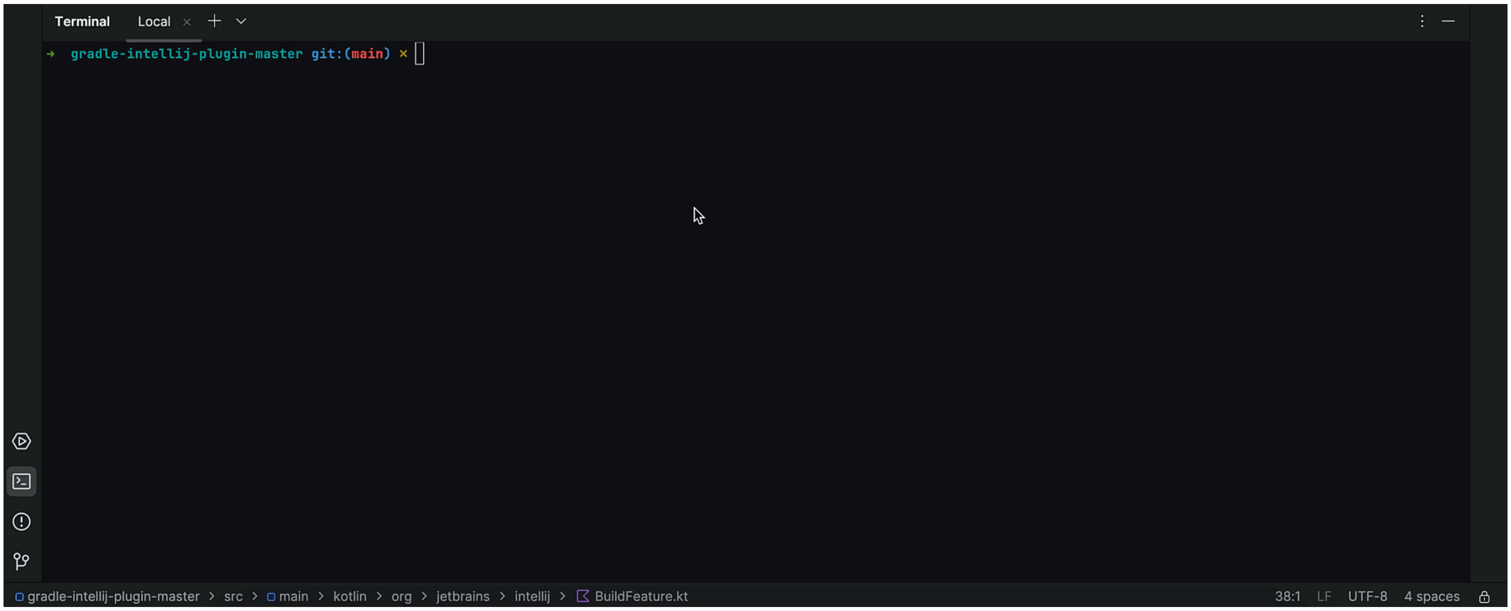
We took full control over the prompt area of the terminal. You can now use editor actions and move the cursor with the mouse during command input. At this stage, some shell shortcuts either do not work or behave incorrectly. We are working to address these issues.

We reworked command completion that integrates all of the IDE’s knowledge into the suggestions. Since we are still working on this update, some completion options may not be available or they may differ from shell-based completion.

We added our own command prompt, which shows the current directory, Git branch, Python virtual environment, and Anaconda environment. This is intended to provide a rich, modern command prompt even if it is not configured in a shell. We understand that some prefer having the same prompt as in the system terminal, and we are working on an option to allow this.

How to enable the new terminal

Starting from version 2024.1, when the *Terminal* tool window is opened for the first time, you will see a promotional window for the new terminal. After that, you can switch between the old and new terminal using the three dots menu in the upper right corner of your *Terminal* tool window or via *Settings/Preferences | Tools | Terminal | Enable New Terminal*.



```
→ gradle-intellij-plugin-master git:(main) ×
```

gradle-intellij-plugin-master > src > main > kotlin > org > jetbrains > intellij > BuildFeature.kt
38:1 LF UTF-8 4 spaces

At the moment, we only support a limited set of shells: Zsh, Bash, and PowerShell. If you use any other shell, you will see the old terminal even if the new setting is enabled.

What's next?

Currently, we've only taken the first step in overhauling the terminal, and there's much more work ahead. Our immediate focus is on improving the stability of the new terminal and refining its current behavior.

We plan to introduce more terminal customization options (fonts, spacing, etc.), improve command completion, add the ability to switch between the built-in and shell prompt, introduce context actions for blocks, and streamline tab management in the terminal. Additionally, we have plans to implement useful AI capabilities, such as assisting in command typing or explaining the obtained results.

How to provide feedback

The new JetBrains IDE terminal, which will be made available from v2024.1, is in Beta and currently undergoing changes and improvements. At this stage, your feedback is crucial to us. Our goal is that the new terminal will have the same capabilities as the old one, while offering a host of beneficial new features.

As we are currently in the development process, we acknowledge that some issues may arise when interacting with the new terminal. Please report any issues on YouTrack. Below you can find several links to familiarize yourself with known issues:

- https://youtrack.jetbrains.com/issue/IDEA-315870/New-Terminal-support-shortcuts-from-the-Shells
- https://youtrack.jetbrains.com/issue/IDEA-338086/New-Terminal-Completion-doesnt-suggest-correct-command-argument-option-to-complete
- https://youtrack.jetbrains.com/issue/IDEA-315916/New-Terminal-command-completion
- https://youtrack.jetbrains.com/issue/IDEA-342682/New-Terminal-The-ability-to-switch-to-a-shell-prompt
- https://youtrack.jetbrains.com/issue/IDEA-346679/New-Terminal-customization-support-custom-shell-settings-color-prompt-completion

We plan to add a survey in the IDE where you can share your impressions of the new terminal. We would also be grateful for any feedback in the comments section below. Thank you!

JetBrains IDEs terminal

Share

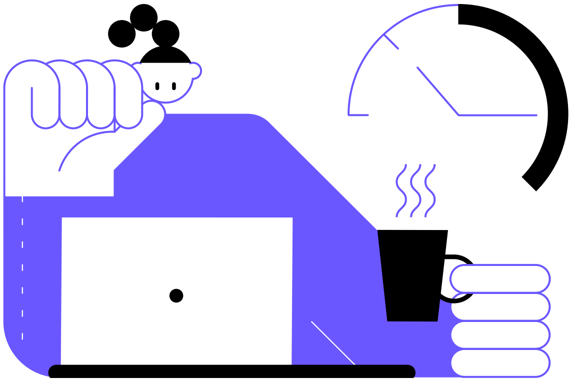
← 'HelloWorld' and 'main()' meet minimalistic

Subscribe to Blog updates

Your email

□ By submitting this form, I agree to the JetBrains Privacy Policy

Submit



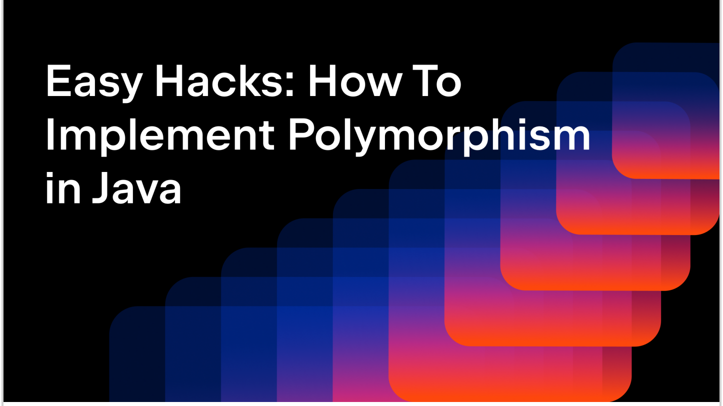
Discover more

‘HelloWorld’ and ‘main()’ meet minimalistic

In this blog post, I'll cover why and how this feature is helpful for students using hands-on coding examples. I’ll also cover some details and frequently asked…

Mala Gupta
February 21, 2024

💬 2



Easy Hacks: How To Implement Polymorphism in Java

Polymorphism is the ability of an object to take on different forms. In programming, this means that a variable or a method can have different behaviors depending on th…

Maarten Balliauw
February 15, 2024

💬 0

Constructor Makeover in Java 22

Typically, we create alternative solutions for tasks that are necessary, but not officially permitted. For instance, executing statements before super() in a…

Mala Gupta
February 11, 2024

💬 1

Mar 6 2024
You’re Invited to IntelliJ IDEA Conf 2024!

We are back and excited to invite you to IntelliJ IDEA Conf 2024, a developer-focused, live, online event that is free for all to attend!

Mala Gupta
February 6, 2024