



Microsoft Java Developer Conference 2024

Code. Cloud. Community.





Red Hat



Microsoft

Sparking Developer Joy: Elevating Developer Experience with VS Code Java





Mohit Suman

Senior Product Manager
Developer Experience



Roland Grunberg

Principal Software Engineer, Team Lead
IDE Extensions



Nick Zhu

Senior Product Manager





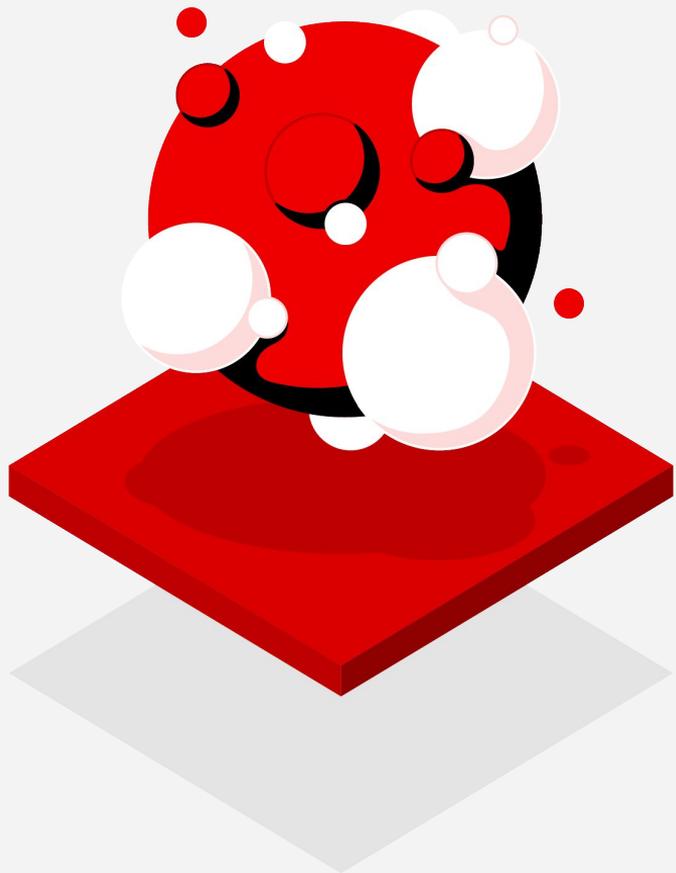
Red Hat

Red Hat

and Microsoft



Microsoft



You do it with
open source.

Level Up Java

The three ways the developer experience will be improved



Inception

Helps start your Java journey without installing and learning a complex IDE.

Provides great microservices support including popular frameworks, container tooling, and cloud integration.



Awareness

Create awareness and identify leads with topic affinity (Java workloads to modernize).

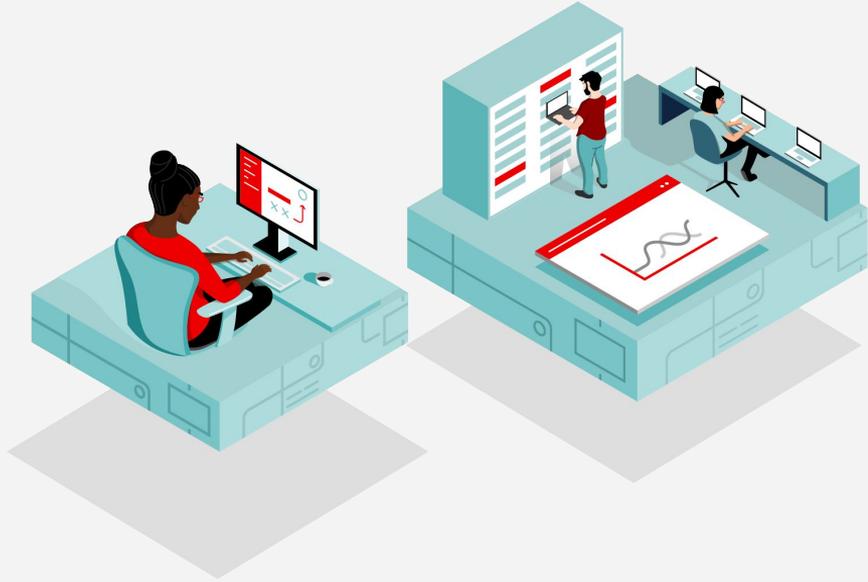
In terms of developer experience, the primary focus here is on the Java ecosystem on local IDE and seamlessly flow in the cloud-native environment.



Downstream ecosystem

Looking downstream, we also see the ecosystem of extensions for Java developers has grown considerably.

We have "**Debugger for Java**", "**Test Runner for Java**", "**Maven for Java**", and "**Project Manager for Java**."



understanding
the features behind
our products

● open source

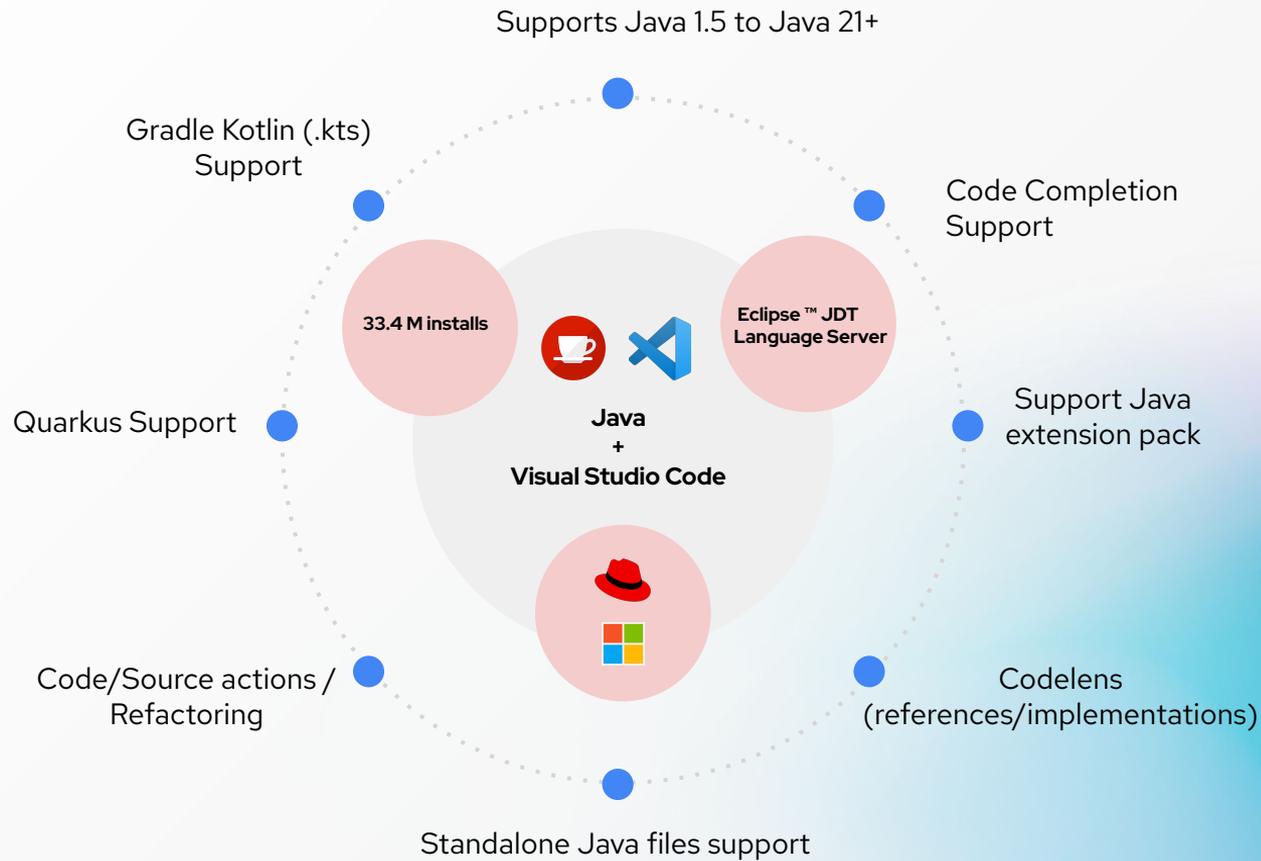
● Java developer
experience



Visual Studio

Code and java

21+ and beyond



Visual Studio Java Engagements



33.4 M Installs

2.5 M active java users

Developer **INNER** "Loop"



VS Code Java Demo

We will go through the **demo of VSCode Java** extension and cover various feature sets available currently. This will have live Q&A, so please feel free to ask them.



Java 21+ We are striving to support the latest language features and enhancements provided by the Java language.



Improved **Gradle support**. Thus delivering overall developer experience.



Build community. While it's easy to discuss accomplishments in terms of the many enhancements, and fixes provided, this project owes its growth to a variety of contributors spanning different ecosystems, and companies with developer blog.



Provide **performance improvements** and Some better **default JVM options** while launching have allowed us to improve our experience on larger projects.

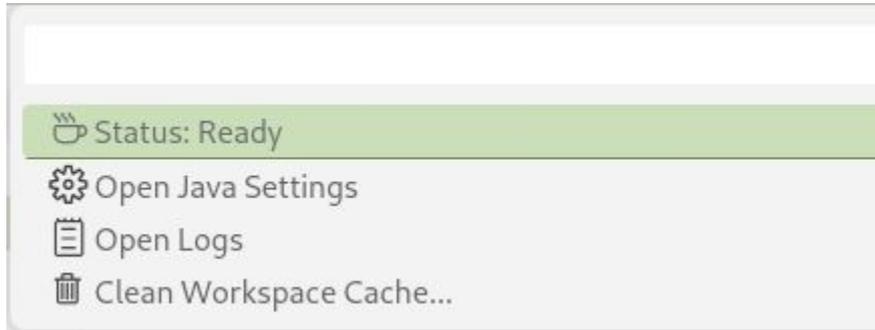
demo time



Performance & Optimization

- Validation (errors/warnings)
- Startup / Project Import (decoupled from autobuild)
- Indexing (shared)
- Completion (improvements to LSP Spec)
- Codebase simplification resulting in more features

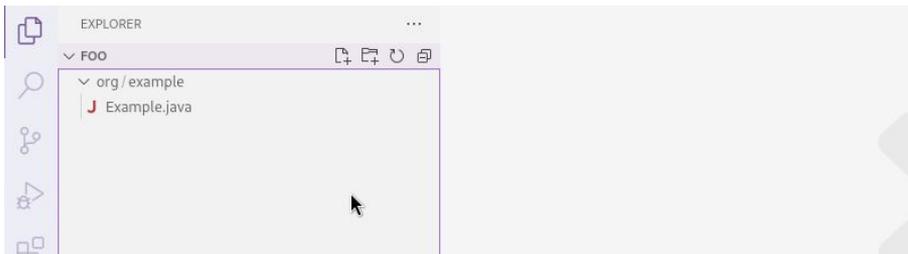
Navigation



```
0 references
public boolean methodExitExample(String location) throws Exception {
    URI uri = new URI(location);
    if (!"file".equals(uri.getScheme())) {
        throw new IOException();
    }
    return true;
}
```

```
0 references
public boolean methodExitExample(String location) throws Exception {
    URI uri = new URI(location);
    if (!"file".equals(uri.getScheme())) {
        throw new IOException();
    }
    return true;
}
```

Editing



```
package org.example;
```

0 references

```
public class Main {
```

Run | Debug | 0 references

```
public static void main(String[] args) {
```

```
    String html = "<html lang=\"en-US\">\n" + //  
        " <head>\n" + //  
        "   <meta charset=\"utf-8\" />\n" + //  
        "   <meta name=\"viewport\" content=\"width=device-width\" />\n" + //  
        "   <title>My test page</title>\n" + //  
        " </head>\n" + //  
        " <body>\n" + //  
        "   <img src=\"images/firefox-icon.png\" alt=\"My test image\" />\n" + //  
        " </body>\n" + //  
        "</html>";
```

```
    System.out.println(html);
```

```
}
```

J Main.java 1 ●

src > org > example > J Main.java > Main

0 references

```
23 public class Main {  
24     public void test() {  
25         String PATH = "/this/is/some/path";  
26  
27         String MESSAGE = "This is a welcome message \n" +  
28             "This message takes multiple lines \n" +  
29             "This is the last piece of text";  
30  
31         List<String> words = List.of("test1", "test2", "test3",  
32             "test3", "test4", "test5");  
33         List<Date> dates = new ArrayList<>();  
34  
35         words.stream().map(mapper:(w) -> {  
36             return w.toUpperCase() + "test";  
37         });  
38  
39         dates.stream().map(mapper:date -> date.getTime());  
40  
41         Object[] obj = Arrays.asList(PATH).toArray();  
42         if (obj[0] instanceof String) {  
43             String tmp = (String)obj[0];  
44             File f = new File(pathname:tmp);  
45             File[] filtered = f.listFiles(new FileFilter() {  
46                 @Override  
47                 public boolean accept(File pathname) {  
48                     return true;  
49                 }  
50             });  
51             if (filtered[0].getAbsolutePath().equals(PATH)) {  
52                 System.out.println(MESSAGE);  
53             }  
54         }  
55     }  
56 }
```

Debugging

RUN AND DEBUG | No Configurations

VARIABLES

- Local
 - jsonString: " {
type: Class (Accessory)@18

WATCH

CALL STACK

- Thread [main] PAUSED ON STEP
 - Main.deserializeData(String, Class)
 - Main.main(String[]) Main.java 15:1
 - Thread [Reference Handler] RUNNING
 - Thread [Finalizer] RUNNING
 - Thread [Signal Dispatcher] RUNNING
 - Thread [Notification Thread] RUNNING
 - Thread [Common-Cleaner] RUNNING

src > org > example > J Main.java > Main > deserializeData(String, Class<?>)

Run | Debug | 0 references

```
6 .....public static void main(String[] args) {  
7 .....    final String jsonString = ""  
8 .....  
9 .....        {  
10 .....            "label": "Mirror",  
11 .....            "price": 59.99,  
12 .....            "type": "Kitchen"  
13 .....        }  
14 .....    }  
15 .....    if (deserializeData(jsonString, type) != null) {  
16 .....        String message = switch (type) {  
17 .....            case Apparel ap -> String.format("Apparel: %s", ap.getName());  
18 .....            case Accessory ac -> String.format("Accessory: %s", ac.getName());  
19 .....            default -> throw new IllegalArgumentException("Invalid type: " + type);  
20 .....        };  
21 .....        System.out.println(message);  
22 .....    }  
23 .....  
24 .....}  
25 .....private static Object deserializeData(String jsonString, Class<T> type) {  
26 .....    Gson gson = new Gson();  
27 .....    return gson.fromJson(jsonString, type);  
28 .....}  
29 .....}
```

Class (Accessory)@18 "class org.example.Accessory"

- annotationData: null
- annotationType: null
- cachedConstructor: null
- classData: null
- classLoader: ClassLoaders\$AppClassLoader@32
- classRedefinedCount: 0
- classValueMap: null
- componentType: null
- enumConstantDirectory: null
- enumConstants: null
- genericInfo: null
- module: Module@33
- name: "org.example.Accessory"
- packageName: "org.example"
- reflectionData: null

Hold Alt key to switch to editor language hover

Language Features

```
J Main.java X
J Main.java > Main > printCopyright(String, int)
Run | Debug | 0 references
1 void main(){
2     int year = 2024;
3     String company = "Some Company Inc.";
4     printCopyright(company, year);
5 }
6
1 reference
7 public void printCopyright(String company, int year){
8     System.out.println(x:STR. " "
9     /*****
10     * Copyright (c) \{year\} \{company\} and others.
11     * All rights reserved. This [x] company String
12     * are made available under [x] year int
13     * which accompanies this di STR StringTemplate.Processor<java.lang.Strin...
14     * https://www.eclipse.org/l clone() Object
15     * equals(Object obj) boolean
16     * SPDX-License-Identifier: finalize() void
17     * getClass() Class<?>
18     * Contributors: hashCode() int
19     * \{company\} -- initia main() void
20     * *****/ notify() void
21     " "]; notifyAll() void
22 }
```

VS Code Java Future Roadmap

Fundamental Experience

- Latest JDK support
- Enhanced code navigation
- Advanced debugging capabilities
- Integrate **javac compiler** as alternative to ECJ

Performance and reliability

- Reduce unreliable sessions
- Optimize JVM count and CPU/memory usage per session

Compatibility

- Improve project success rate of Build Server of Gradle

VS Code Java Future Roadmap

Project Management

- Improved JDK Configuration Experience
- Better project creation / import experience
- UX import in package / create class creation

User Experience

- Better path to code action and refactoring options
- Easier discovery to Java language server options
- Improve setting discovery

AI

- Better integration with GitHub Copilot Chat



<https://github.com/redhat-developer/vscode-java>



<https://marketplace.visualstudio.com/items?itemName=redhat.java>



<https://twitter.com/vscodejava> / <https://twitter.com/JavaAtMicrosoft>

Thank you



Extension Pack for Java

Microsoft  microsoft.com |  26,025,750 installs |       (72) | Free

Popular extensions for Java development that provides Java IntelliSense, debugging, testing, Maven/Gradle support, project management and more

Install

[Trouble Installing?](#)



Language Support for Java(TM) by Red Hat

Red Hat  redhat.com |  33,393,438 installs |      (160) | Free

Java Linting, Intellisense, formatting, refactoring, Maven/Gradle support and more...

Install

[Trouble Installing?](#)