Agenda

1. OpenJDK Quality Group
2. Java Playground
OpenJDK Quality Group
Quality Outreach Program

- Promote the testing of FOSS projects with the latest early-access OpenJDK JDK builds
- Raise awareness on upcoming changes
Quality Outreach Program

- Promote the testing of FOSS projects with the latest early-access OpenJDK JDK builds
  - Identify and address potential issues as early as possible
  - Bandwidth & knowledge
- Raise awareness on upcoming changes
  - Anticipate potential issues as early as possible

⇨ Improve the Java platform quality
⇨ Evolve mindsets from Reactive to Proactive
How?

- Recruit FOSS projects
- Mail Blast ⇔ Feedback ⇔ Bugs
- Regular QO mail blast
  - News & builds updates
  - Heads-ups & call-to-action
    - [https://inside.java/headsup/](https://inside.java/headsup/)
    - Latest blast
      - [https://mail.openjdk.org/pipermail/quality-discuss/2024-June/001135.html](https://mail.openjdk.org/pipermail/quality-discuss/2024-June/001135.html)
- JBS
  - [https://bugs.openjdk.org/browse/JDK-8319782?jql=labels=apache-lucene-found](https://bugs.openjdk.org/browse/JDK-8319782?jql=labels=apache-lucene-found)
Awareness

- **July 2023 - JDK 21 & JDK 22**
  - Note if implicit annotation processing is being used

- **October 2023 - JDK 22**
  - No note is emitted, and annotation processors are not run, restore the previous behavior with `-proc:full java`
  - Backed out!

- **June 2024 - JDK 23**
  - Changes Default Annotation Processing Policy
  - `javac` requires an additional command-line option to enable annotation processing
  - Backported

... *besides that annotation processor change tripping us up, we're green on JDK23EA. Though I had hoped to see ... so that the annotation processor behavior change would be more visible...*
~180 Participating Projects

- Apache Ant
- Apache Derby
  - https://lists.apache.org/thread/7v9r3gmcjsqr78v99xdvvp2q7h892odq
- Apache Lucene
  - https://www.elastic.co/jp/blog/lucene-jvm-bugs
- Apache Tomcat
- Eclipse Collections
- Eclipse Hibernate
- Junit 5
- MyBatis
- ...
The Arrival of Java 22!
Sharat Chander on March 19, 2024

JDK 22 is Now Available!

Oracle is proud to announce the general availability of JDK 22 for developers, enterprises, and organizations globally. The JDK 22 release includes support for Java HotSpot® VM 11, which is also required in some open source containers and some open source server applications.

Additionally, through the OpenJDK Quality Outreach program we would like to thank the following FOSS projects that provided excellent feedback on testing Java 22 early access builds to help improve the quality of the release:

- Apache Syncope (Francesco Chichiricò)
- Apache Tomcat (Mark Thomas)
- ApprovalTests.java (Lars Eckart)
- AssertJ (Stefano Cordio)
- Avaje (Josiah Noel)
- Jetty (Simone Bordet)
- MyBatis (Iwao Ave)
- Parallel Collectors (Grzegorz Piwowarek)
- RxJava (David Karnok)
Benefits

- Recognition
- Improved projects quality
- Improved OpenJDK quality
- OpenJDK author role (*)
Resources

- Quality Outreach
  - [https://wiki.openjdk.org/display/quality/Quality+Outreach](https://wiki.openjdk.org/display/quality/Quality+Outreach)
- Heads-up
  - [https://inside.java/headsup/](https://inside.java/headsup/)
- Mailing list
  - [https://mail.openjdk.org/pipermail/quality-discuss/](https://mail.openjdk.org/pipermail/quality-discuss/)
- Oracle setup-java GitHub action
  - [https://github.com/oracle-actions/setup-java](https://github.com/oracle-actions/setup-java)
Java Playground
The Java Playground

- Simple online tool that helps exploring Java Language features
  - [https://dev.java/playground/](https://dev.java/playground/)
- “No setup required, just type your Java snippet and run it!”

Lorem ipsum, sed ut persipciatis, unde omnis iste natus error sit voluptatem accusantium doloremque laudantium, totam rem aperiam eaque ipsa, quae ab illo inventore veritatis et quasi architecto beatae vitae dicta sunt, expilacio. Nemo enim ipsam voluptatum, quia voluptas sit, aspernatur aut odit aut fugit.

**Regular Mode**

This mode is the default mode, the UI shows the "samples" Dropdown.

```java
// Simple fixed window generator
var strings = List.of("one", "two", "three");
Gatherer<
    ? String, \
    List<String>
> gatherer = new Gatherer<>(
    str -> \
        System.out.printf("%s\n", str));
var result = strings.stream()
    .forEach(gatherer::set)
    .toList();
System.out.println("result = " + result);
```

**Tutorial Mode**

This mode offers a simplified UI targeting tutorial, ex. there's no "Samples" dropdown, a "Reset" button to reset the snippet instead of clearing it.

A record is class declared with the `record` keyword instead of the `class` keyword, here's a working example:

```java
record inside.java
```
The Java Playground

- Java: Java 22, jshell API and Classfile API
- Infra: OCI Functions
- UI: Javascript and Web Component
- Restrictions
  - APIs ⇒ Amber, Collections, ...
  - Runtime
- Not a development tool!
<java-playground id="Hello World">  
  <snippet>
    var name = "Duke";
    System.out.println("👋 Hello, " + name);
  </snippet>
</java-playground>
<java-playground id="Another test">
  <snippet>
    <!!--
    var a = 1;
    var b = a+1;
    var c = b+1;
    if ((a < b) && ( c > b)) System.out.println("<test>");
    -->
  </snippet>
</java-playground>
<java-playground id="A test" gist="48393e32...ce7faacb"> </java-playground>
Demo

**Tutorial Mode**

This mode offers a simplified UI targeting tutorial, ex. there's no "Samples dropdown, a "Reset" button to reset the snippet instead of clearing it.

A record is class declared with the `record` keyword instead of the `class` keyword, here's a working example.

```java
record City(String name, int population) {
}
```

```java
var paria = City.of("Peria", 2_303);
System.out.println(paria);
```

For more information, make sure to check Using Record to Model Immutable Data.

The Playground also prevents the execution of some operations. We know you want to try, so here's an example.

```java
Thread.sleep(1);
throw new Exception("Boom");
```
Roadmap

- Summer
  - JDK 23
  - Dev.java tutorials
  - A11Y
  - ...
- TBC
  - Prefix
  - ...

Copyright © 2024 Oracle and/or its affiliates.
Thank you

David Delabassee
Developer Relations
Java Platform Group
@delabassee