

Walkthrough of the Java ShutdownOnFailure Code

Douglas C. Schmidt

d.schmidt@vanderbilt.edu

www.dre.vanderbilt.edu/~schmidt

Professor of Computer Science

**Institute for Software
Integrated Systems**

**Vanderbilt University
Nashville, Tennessee, USA**



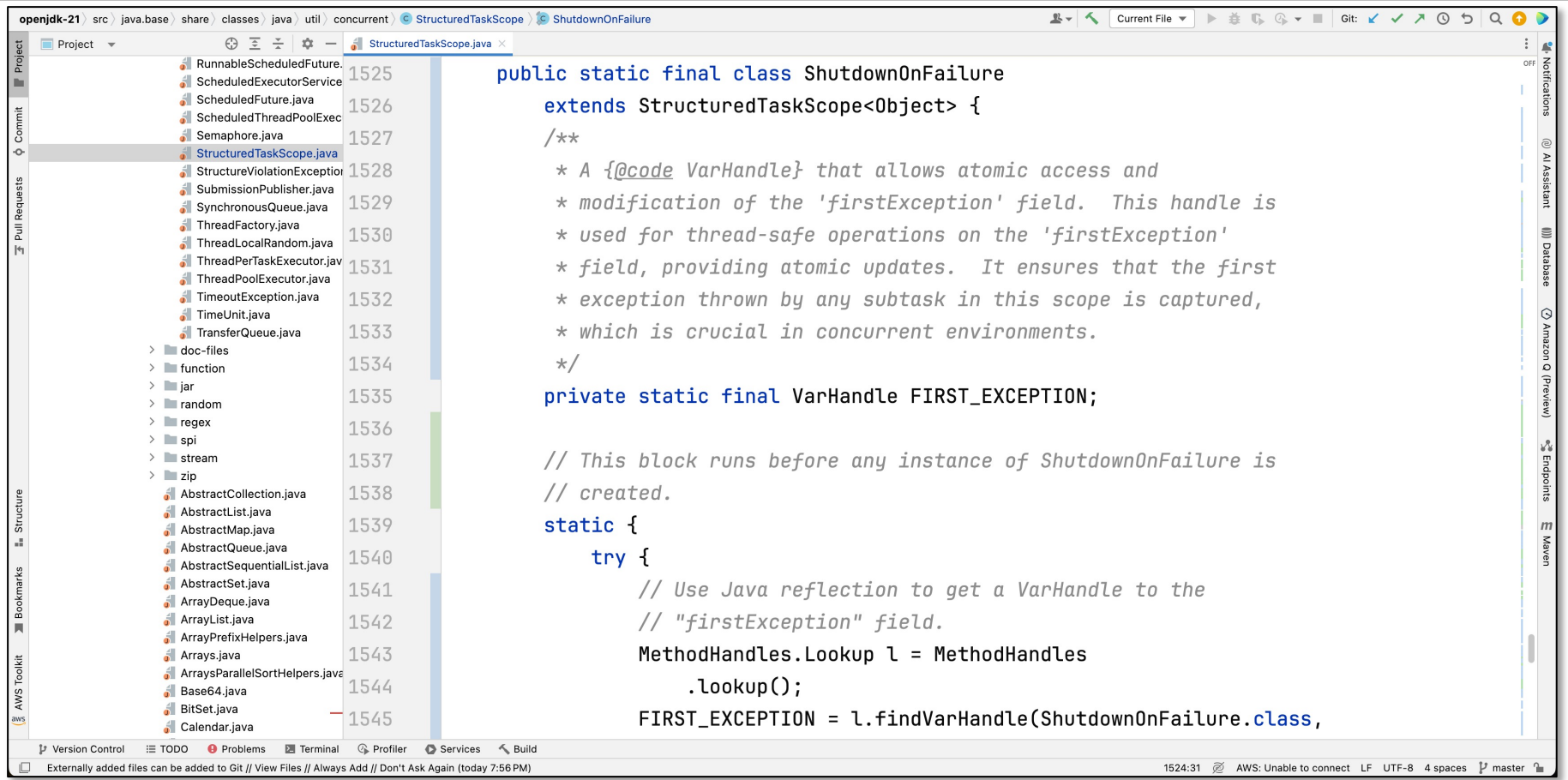
Learning Objectives in this Part of the Lesson

- Understand Java's structured concurrency model
- Recognize the classes used to program Java's structure concurrency model
- Evaluate the design & performance of various Java concurrency models
- Learn how StructuredTaskScope is implemented
 - Focus on ShutdownOnFailure
 - Implements "invoke-all" semantics

```
public static final class ShutdownOnFailure
    extends StructuredTaskScope<Object> {
    private static final VarHandle FIRST_EXCEPTION;
    static {
        try {
            MethodHandles.Lookup l = MethodHandles
                .lookup();
            FIRST_EXCEPTION = l.findVarHandle(ShutdownOnFailure.class,
                "firstException",
                Throwable.class);
        } catch (Exception e) {
            throw new ExceptionInInitializerError(e);
        }
    }
    private volatile Throwable firstException;
```

Walkthrough of the Java ShutdownOnFailure Code

Walkthrough of the Java ShutdownOnFailure Code



The screenshot shows an IDE window with the file `ShutdownOnFailure` open. The left sidebar displays a project tree with various Java classes and packages. The main editor area shows the following code:

```
public static final class ShutdownOnFailure
    extends StructuredTaskScope<Object> {
    /**
     * A {@code VarHandle} that allows atomic access and
     * modification of the 'firstException' field. This handle is
     * used for thread-safe operations on the 'firstException'
     * field, providing atomic updates. It ensures that the first
     * exception thrown by any subtask in this scope is captured,
     * which is crucial in concurrent environments.
     */
    private static final VarHandle FIRST_EXCEPTION;

    // This block runs before any instance of ShutdownOnFailure is
    // created.
    static {
        try {
            // Use Java reflection to get a VarHandle to the
            // "firstException" field.
            MethodHandles.Lookup l = MethodHandles
                .lookup();
            FIRST_EXCEPTION = l.findVarHandle(ShutdownOnFailure.class,
```

See github.com/openjdk/jdk

End of Walkthrough of Java
StructuredTaskScope Code