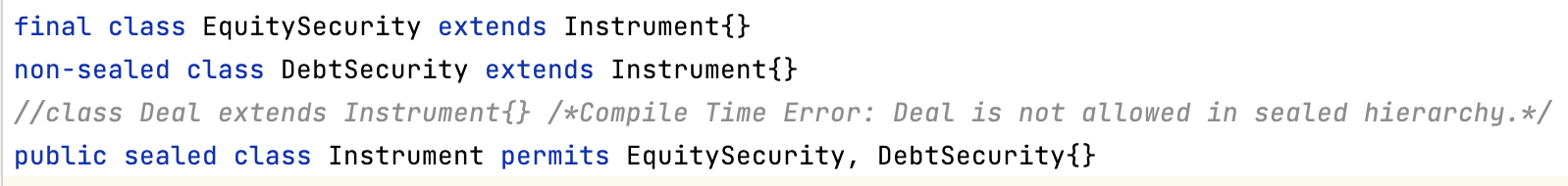
**Java 8 to Java 17 Features and Benefits**

**Features**

1. **Sealed Classes and Interfaces**
2. Description: It was introduced in Java SE 17. A sealed class or interface restricts which classes or interfaces can extend or implement it.
3. Reason: The final modifier does not allow anyone to extend a class. If we want to allow some classes  
   to extend a class, we would need this feature.
4. Constraints: Need to be mindful while implementing Sealed class.
5. Permitted subclass must be accessible by the sealed class at compile time.
6. Permitted subclass must directly extend the sealed class.
7. Permitted subclass must have one of the following modifiers, final, sealed, or non-sealed.
8. Permitted subclass must be in the same Java module.
9. Example:



1. API: The class java.lang.Class has two new methods related to sealed classes and interfaces:
2. permittedSubclasses()
3. isSealed()
4. Reference: https://docs.oracle.com/en/java/javase/17/language/sealed-classes-and-interfaces.html
5. **Record Classes**
6. Description
7. Reason:
8. Constraints:
9. Example:
10. API:
11. Reference:
12. Switch Expression
13. Description
14. Reason:
15. Constraints:
16. Example:
17. API:
18. Reference:
19. Pattern matching for Instance Of
20. Pattern Matching for Switch Expressions and Statements
21. Text Blocks
22. Local Variable Type Reference
23. Project Jigsaw(Java Platform Module System)
24. Project Coin
25. Random Number Generators in Java 17
26. Restore Always-Strict Floating-Point Semantics
27. Dealing with Unsafes
28. Improvements in Deserialization
29. Vector API (Second Incubator)
30. Performance Improvements - Garbage Collection  
      
      
    Sources of Knowledge:  
    1. https://www.linkedin.com/pulse/from-java-8-17-features-benefits-aneshka-goyal/  
    2. https://docs.oracle.com/en/java/javase/17/language/java-language-changes.html#GUID-6459681C-6881-45D8-B0DB-395D1BD6DB9B  
    3. https://www.javatpoint.com/java-17-new-features  
    4. https://www.baeldung.com/java-hidden-classes  
    5. https://blogs.oracle.com/javamagazine/post/a-peek-into-java-17-continuing-the-drive-to-encapsulate-the-java-runtime-internals  
    6. https://snyk.io/blog/serialization-and-deserialization-in-java/  
    7. https://blogs.oracle.com/javamagazine/post/the-unsafe-class-unsafe-at-any-speed  
    8. https://www.baeldung.com/java-security-manager  
    9. https://docs.oracle.com/javacomponents/jmc-5-4/jfr-runtime-guide/about.htm#JFRUH170  
    10. https://kstefanj.github.io/2021/11/24/gc-progress-8-17.html  
    11. https://www.baeldung.com/java-strictfp  
    12. https://www.baeldung.com/jvm-zgc-garbage-collector  
    13. https://www.cesarsotovalero.net/blog/aot-vs-jit-compilation-in-java.html  
    14. https://www.baeldung.com/java-try-with-resources  
    15. https://www.tutorialspoint.com/java9/java9\_inner\_class\_diamond\_operator.htm