Day 7 Revisit (Java 11, 17 Features, Sealed Classes, NIO)

- New Methods in String class – isBlank(), strip(), lines(),
- New methods added in file handling
- ReadString() & WriteString()
- Collections toArray
- New way of running the java code (We can directly pass the java source code to jvm to execute it)
- OpenJDK , Merged JDK & JRE folder
- Exception Handling (Types of Exception)
- Exception Hierarchy (Throwable)
- Ways of Handling Exception (Using throws keyword, using try catch block)
- Runtime Exception (Unchecked Exception) and CompileTime Exception (Checked Exception)
- Types of try/catch block
- Try with finally
- Try with catch and finally
- Try with multiple catch block
- Keywords related to Exception Handling (try, catch, finally, throw, throws)
- Access & Non-access Modifier
- Sealed Classes (Introduced in Java15)
- Managing different versions of JDK (1.8,11,17) – Environment variable and path.
- Sealed class – Restricts no of class which can inherit it.
- Keywords related to sealed class (sealed, permits, non-sealed)
- The class which inherits sealed class should be final, sealed or non-sealed only.
- Pattern Matching (java.util.regex) – Regular Expression
- Pattern, Matcher, MatcherResult(I), PatternSyntaxException
- Searching, StringManipulation, Validations
- NIO & NIO2 – New streams, Local Variable,
- ByteStream, CharacterStream

Ref URL :

https://docs.oracle.com/javase/8/docs/

Agenda

- TDD (Test Driven Development)
- Junit 5
- Mockito
- SonarQube
- Lombok
- Logging
- NFR (Non Functional Requirement) – Performance Related

TDD – Test Driven Development

- Write the Test first even before starting application code.
- TDD ensures all the functionalities, requirements are taken care.
- Test cases will fail first.
- Write the application code to make the test pass.

TDD – is a software development practice that emphasizes writing tests before writing the actual code.

Test Driven Refactoring – It's way(practice) of improving the application code.

JUNIT 5.

Official Site : https://junit.org/junit5/

Junit – Unit Testing framework for Java

JUnit 5 = *JUnit Platform* + *JUnit Jupiter* + *JUnit Vintage*

Junit Annotations

@Before

@Test

@BeforeEach

@BeforeClass

@After

@AfterEach

@AfterAll

Best Practice : Import needed code only (Class/Interface/Exception/Annotation)

Static Import – Importing static methods from a class

Testing – Manual & Automated Testing

Testing categories

- Unit Testing
- Integration Testing
- E2E Testing (End to End Testing)
- Performance Related Testing (Load Testing, Stress)
- Black box/White box/Sand box
- Acceptance Testing (UAT)

Why Testing

- To Ensure all the requirements are implemented
- To Ensure all the functionalities are implemented in expected way.

Env : Dev, Test, Pre-prod, UAT, Production

@Test

@ParameterizedTest

@DisplayName

Lombok - https://projectlombok.org/

Download the Lombok jar -- https://projectlombok.org/download

Install to IDE (Double click to install it)

@Data = For generating getters/setters, toString, hashcode()

@Getter = Generate getter method

Mockito -

Mock Objects (Test Double) /Dummy Objects

Recommended Package/folder structure (Project)

Com.sapient.tms (Base package) – Starter Class (Main method)

Com.sapient.tms.model/entity – All the Entity bean classes (Trainer/Company/SME)

Com.sapient.tms.util – DBUtil

Com.sapient.tms.repo (DAO) – Interfaces & for CRUD operation (readAll(), readById(), insert(), edit/update(), deleteById())

DAO – Data Access Object

DTO – Data Transfer Object

SUT – System Under Test

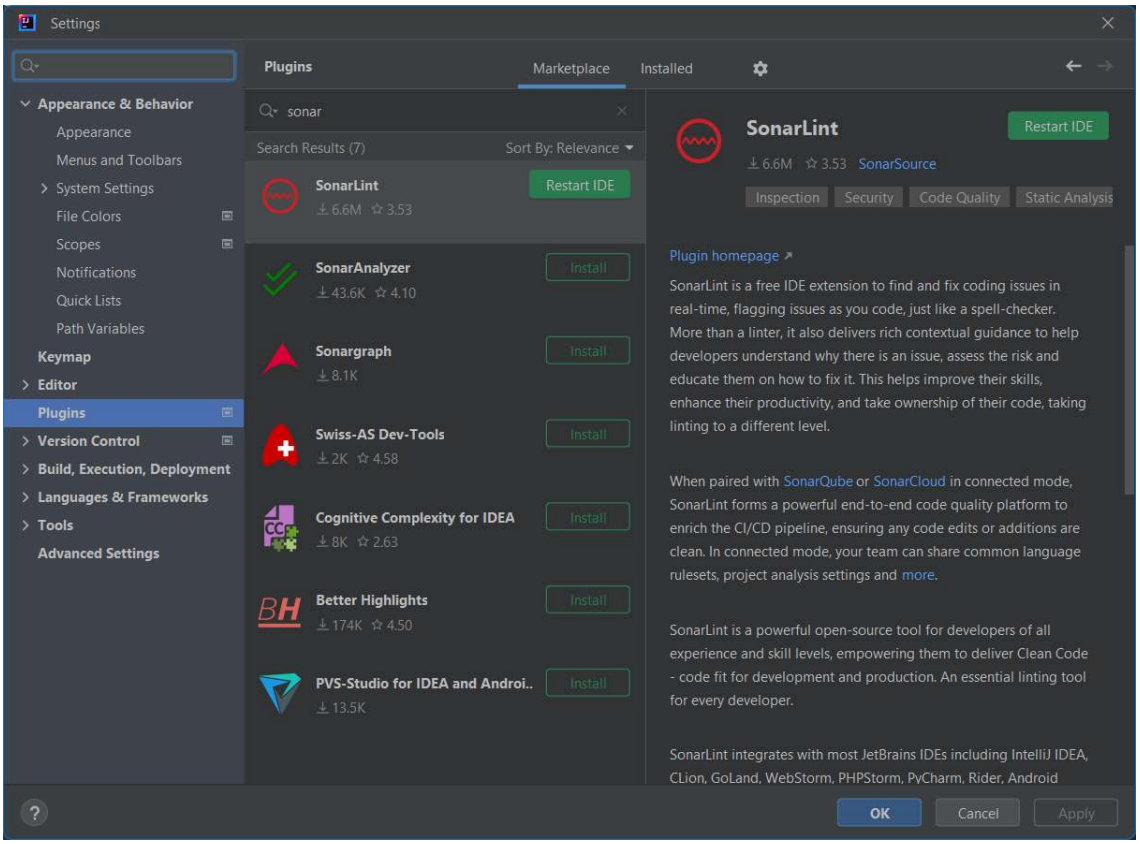
Mockito – Open Source Mock Testing Framework (Mocking framework) for Java based project

Mock/clone/test double/dummy object

Sonarqube – Static Code analysis – helps to write clean, efficient, vulnerability free code. – Code Smell

Sonarlint – Plugin that can be integrated with IDE

Adding sonarlint to IntelliJ – File→ Settings→ plugins (Search for sonarlint)



Running All test with coverage – In intellij IDE

Sonar Cloud -

Logging -- the act of keeping a log of events that occur in a computer system, such as problems, errors or just information on current operations

Logging – Java SE

Log4J, Logging for Java

SLF4J – Simple Logging Façade for Java

Log Messages – Date & Time Stamp – From Which Class/Function : Type of Message (Logging Level) – INFO, DEBUG, WARN, ERR