

1. See `logger.log`, why is it different from the log to console?

- When a log is made with the level INFO, it is printed to the console. The finer level does not print to the console. Therefore the `Logger.log` file contains information that isn't in the console.

2. Where does this line come from? `FINER org.junit.jupiter.engine.execution.ConditionEvaluator logResult Evaluation of condition [org.junit.jupiter.engine.extension.DisabledCondition] resulted in: ConditionEvaluationResult [enabled = true, reason = '@Disabled is not present']`

- This line comes from the `TimerException` thrown from `timeMe` method and `TimefailOverTest`

3. What does `Assertions.assertThrows` do?

- Asserts the executable for the given exception and returns the exception

4. See `TimerException` and there are 3 questions

1. What is `serialVersionUID` and why do we need it? (please read on Internet)

- Each serializable class gets a version number which is what the `serializableVersionUID` is. It is used during deserialization to verify that the sender and receiver of a serialized object have loaded classes for that object that are compatible with respects to serialization. We declare it so that an unexpected `InvalidClassException` isn't thrown.

2. Why do we need to override constructors?

- We need to override the constructors because once we override one, we no longer have the default. Therefore, we have to override the default and any other desired constructor.

3. Why we did not override other Exception methods?

- The super class handles the other methods, we don't need a specified version of these methods. Therefore, we can let the super class functionality work in the subclass.

5. The `Timer.java` has a static block `static {}`, what does it do? (determine when called by debugger)

- This static block takes the logger properties of the static logger of the `Timer` class and passes it as a stream to the global `LogManager`. This means the log level definitions of the logger are applied to the global `LogManager`. It gets called the first time the `Timer` class is used. The first time it's used is in the `failOverTestEdge` method of the `teaser`.

6. What is README.md file format how is it related to bitbucket?

(<https://confluence.atlassian.com/bitbucketserver/markdown-syntax-guide-776639995.html>)

- The .md file format means it can contain Markdown and a restricted set of HTML tags. Also, Bitbucket's Server will display the contents on the source page of the repository automatically.

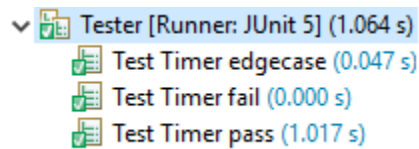
7. Why is the test failing? what do we need to change in Timer? (fix that all tests pass and describe the issue)

- The test was failing due to the NullPointerException from timeNow. When the time to wait is less than 0 it throws the TimerException but in the finally block it uses this variable which is still null. There are many ways to fix the issue, but I assume the Logger needs to be told there was a problem. My fix was to add a conditional statement that check to see if timeToWait is less than 0 and log the proper information.

8. What is the actual issue here, what is the sequence of Exceptions and handlers (debug)

- As stated in the previous answer, the TimerException and even InterruptedException are handled but a NullPointerException is not handled.

9. Make a printScreen of your eclipse JUnit5 plugin run (JUnit window at the bottom panel)



10. Make a printScreen of your eclipse Maven test run, with console

```
H:\exceptionrunner>mvn -Dtest=Tester#passingTest test
[INFO] Scanning for projects...
[INFO] -----< edu.baylor.ecs.si:ExceptionTester >-----
[INFO] Building ExceptionTester 0.0.1-SNAPSHOT
[INFO] -----[ jar ]-----
[WARNING] The POM for org.junit.platform:junit-platform-runner:jar:1.0.0 is invalid, transitive dependencies (if any)
will not be available, enable debug logging for more details
[WARNING] The POM for org.junit.vintage:junit-vintage-engine:jar:4.12.0 is invalid, transitive dependencies (if any)
will not be available, enable debug logging for more details
[INFO] --- maven-resources-plugin:2.6:resources (default-resources) @ ExceptionTester ---
[INFO] Using 'UTF-8' encoding to copy filtered resources.
[INFO] Copying 1 resource
[INFO] --- maven-compiler-plugin:3.1:compile (default-compile) @ ExceptionTester ---
[INFO] Nothing to compile - all classes are up to date
[INFO] --- maven-resources-plugin:2.6:testResources (default-testResources) @ ExceptionTester ---
[INFO] Using 'UTF-8' encoding to copy filtered resources.
[INFO] skip non existing resourceDirectory H:\exceptionrunner\src\test\resources
[INFO] --- maven-compiler-plugin:3.1:testCompile (default-testCompile) @ ExceptionTester ---
[INFO] Nothing to compile - all classes are up to date
[INFO] --- maven-surefire-plugin:2.19:test (default-test) @ ExceptionTester ---

-----
T E S T S
-----
Running edu.baylor.ecs.si.Tester
2018-10-19 10:36:33 INFO edu.baylor.ecs.si.Timer <clinit> starting the app
2018-10-19 10:36:33 INFO edu.baylor.ecs.si.Timer timeMe Calling took: 0
2018-10-19 10:36:33 INFO edu.baylor.ecs.si.Timer timeMe * should take: 0
2018-10-19 10:36:33 WARNING edu.baylor.ecs.si.Timer timeMe Cannot be less than zero
2018-10-19 10:36:34 INFO edu.baylor.ecs.si.Timer timeMe Calling took: 1016
2018-10-19 10:36:34 INFO edu.baylor.ecs.si.Timer timeMe * should take: 1000
Tests run: 3, Failures: 0, Errors: 0, Skipped: 0, Time elapsed: 1.156 sec - in edu.baylor.ecs.si.Tester

Results :

Tests run: 3, Failures: 0, Errors: 0, Skipped: 0

[INFO] -----
[INFO] BUILD SUCCESS
[INFO] -----
[INFO] Total time: 11.788 s
[INFO] Finished at: 2018-10-19T10:36:34-05:00
[INFO] -----
```

11. What category of Exceptions is TimerException and what is NullPointerException

- A TimerException extends Exception which makes it a checked exception. A NullPointerException extends RuntimeException which makes it an unchecked exception.

12. Push the updated/fixed source code to your own repository.

<https://github.com/benagueros/Lab-Logger-with-JUnit>