# **TestLogFile**

This is an sample project to demonstrate how to handle event details from a file named logfile.txt using Java

**# Important**

Make sure you have JAVA 8 or above installed to run this test

**Libraries support**

gson,

hsqldb,

json-simple,

maven-surefire-plugin

**# More Information**

1. Run **TestEntry**.java as RunAs->Java Application

**Following Output is shown in Console:**

Sep 11, 2021 9:23:19 PM org.hsqldb.persist.Logger logInfoEvent

INFO: checkpointClose start

Sep 11, 2021 9:23:19 PM org.hsqldb.persist.Logger logInfoEvent

INFO: checkpointClose synched

Sep 11, 2021 9:23:19 PM org.hsqldb.persist.Logger logInfoEvent

INFO: checkpointClose script done

Sep 11, 2021 9:23:19 PM org.hsqldb.persist.Logger logInfoEvent

INFO: checkpointClose end

Table Created Successfully

{"id":"scsmbstgra","state":"STARTED", "type":"APPLICATION\_ LOG", "host":"12345", "timestamp" :1491377495212}

{"id" :"scsmbstgrb", "state":"STARTED", "timestamp": 1491377495213}

{"id" :"scsmbstgrc", "state":"FINISHED" , "timestamp" :1491377495218}

{"id" :"scsmbstgra", "state":"FINISHED", "type":"APPLICATION\_LOG", "host":"12345", "timestamp":1491377495217}

{"id" :"scsmbstgrc", "state":"STARTED", "timestamp": 1491377495210}

{"id" :"scsmbstgrb", "state":"FINISHED", "timestamp" :1491377495216}

1 rows effected

Rows inserted successfully

1 rows effected

Rows inserted successfully

1 rows effected

Rows inserted successfully

scsmbstgra | 5 | APPLICATION\_ LOG | 12345 | true

scsmbstgrb | 3 | null | null | false

scsmbstgrc | 8 | null | null | true

Sep 11, 2021 9:23:20 PM org.hsqldb.persist.Logger logInfoEvent

INFO: Database closed

**# Implementation Details:**

1. TestEntry.java starts the server dynamically, which internally creates eventDetails\_tbl table

2. Takes the path to logfile.txt as an input argument

3. Deserializes the specified Json into an object of the specified class i.e., Creates an Entry Object for each json line by line using Gson.

4. # Program that can handle very large files (gigabytes)

The lines of file(logfile.txt) are not stored in memory. Instead code or pragram does some processing for each line and throw it away, without holding all them in memory.

5. HashMap hm stores data in Integer,CreateEntryObject

6. HashMap hm1 stores data in String,Object

7. Each iteration of hm1, inserts the data to DB

8. Query the DB

9. Deletes the table

10. Stops the Server

**# Unit Test Coverage Details:**

U1: Let the logfile.txt have below content as part of HappyPath Validation,

a) Validate that eventDetails\_tbl table is created successfully

b) Validate that table displays EventId, EventDuration, Type, Host and Alert

C) Validate that Alert column is flaged as "true" for event timeduration > 4

d) Validate that null value is displayed for everyh row for inapplicable column

{"id":"scsmbstgra","state":"STARTED", "type":"APPLICATION\_ LOG", "host":"12345", "timestamp" :1491377495212}

{"id" :"scsmbstgrb", "state":"STARTED", "timestamp": 1491377495213}

{"id" :"scsmbstgrc", "state":"FINISHED" , "timestamp" :1491377495218}

{"id" :"scsmbstgra", "state":"FINISHED", "type":"APPLICATION\_LOG", "host":"12345", "timestamp":1491377495217}

{"id" :"scsmbstgrc", "state":"STARTED", "timestamp": 1491377495210}

{"id" :"scsmbstgrb", "state":"FINISHED", "timestamp" :1491377495216}

U2: EventDuration Validation: Let the logfile.txt have an event, with EventDuration>4(Eg: Event A,B,C EventDuration be 4,5 10 respectively)

U3: EventDuration Validation: Let the logfile.txt have an event, with EventDuration<4(Eg: Event A,B,C EventDuration be 3,2,0 respectively)

U4: EventDuration Validation: Let the logfile.txt have an event, with EventDuration=4(Eg: Event A,B,C EventDuration be 4,2,0 respectively)

**Error Scenarios:**

U4: Status Validation: Let the logfile.txt have an event, which has status "STARTED" but no "FINISHED"(Eg: Event A with only 'STARTED')

U5: Status Validation: Let the logfile.txt have an event, which has status "FINISHED" but no "STARTED"(Eg: Event A with only 'FINISHED')

U6: Status Validation: Let the logfile.txt have an event, which has status null (Eg: Event A with only '')

U7: EventId Validation: Let the logfile.txt have an event repeated.(Eg. Event A started ..finished, Event A again started and finished)

U8: Empty Line Validation: Let the logfile.txt have aempty line after an event, Validate that application handles gracefully

U9: Let the logfile.txt have 64KB or 65,535 characters, Validate that application handles gracefully

