

Package Structure:

```
\com.l1a
    \- Main.java
com.l1a.analytics
    \- Analyzer.java
com.l1a.model
    \- User.java
com.l1a.utilities
    \- Utils.java
    \- Logger.java
```

Environment:

The project uses Java 8 for compilation and JRE 1.8 as a runtime environment.

Compile:

Use the following code within the root directory of the project to build all the classes.

```
Javac <Source Files.java> -d <Path>
```

```
javac src/com/l1a/*.java src/com/l1a/analytics/*.java src/com/l1a/utilities/*.java
src/com/l1a/model/*.java -d bin
```

Run:

Please use the following code from within the root directory of the project to run the compiled code. Please make sure to change the argument to reflect the absolute path of the logs folder.

```
java -cp bin com.l1a.Main <path/to/data/logs>
```

Eg. java -cp bin com.l1a.Main C:/Users/sbhch/Desktop/LivongoLogAnalytics/data/logs

For Windows, if there are any spaces in the path, please ensure that the whole path is enclosed within quotes.

Eg. java -cp bin com.l1a.Main "C:/My Special/Path/To/Logs"

Create JAR:

Please use the following command to create a jar for the project from within the root/bin directory of the project.

```
jar cmvf ../META-INF/MANIFEST.MF ../LivongoLogAnalytics.jar *
```

Run JAR:

Please use the following command to run the JAR file.

```
java -jar <path/to/JAR/file> <path/to/data/logs>
```

Eg. java -jar LivongoLogAnalytics.jar C:/Users/sbhch/Desktop/LivongoLogAnalytics/data/logs

Data:

The data is provided in the /data/logs folder along with the rest of the files. Please make sure to provide the absolute path to the data.

Output:

The output is displayed on the standard output following the given report. The list of unique users and top 5 users based on page views are displayed along with the number of sessions, longest session and shortest session respectively for each user.

A log file is also created when the project or JAR is executed. The log file stores all the exceptions / errors during the execution. If a proper output is not obtained as expected, please view the log file to see the cause.

Properties File:

The number of top users to be displayed in the report can be changed by changing the value of topK in the info.properties file in the project root directory (or the one with the JAR when the JAR is executed). Please ensure that the info.properties file is always present with the JAR so that it executes properly.

Testing:

There is a Test package along with a Test class which was used for testing the parser and other manual checks. Please do not compile it with the rest of the classes.

Assumptions:

- The argument for a directory is an absolute path to the logs folder.
- The longest and shortest session timings are calculated without any rounding off. For eg. if the starting time is 00:01:00 and ending time is 00:01:59, longest and shortest are still considered as 0 min as only 59 seconds have passed and the 1-minute mark has not been reached yet.
- Log file lines which do not have a valid syntax for time or does not contain a proper user ID has been ignored.
- While sorting users based on their page views, users having the same page views are ordered alphanumerically (descending).
- The log files are in ascending order of time. Each successive access occurs later than its predecessor for a given user. This assumption was later reversed due to observations made while running on Mac OS. The log files were parsed in unsorted manner giving rise to wrong time entries. Corrections to the algorithm was made to accommodate this unusual behavior in Mac OS and possibly other OS as well.