## Task

Program should monitor the incoming folder for log files (i.e file1.csv).   
Log file contains information about user navigation. Log file is comma delimited file (see CSV format) with header. Each line consists of four fields: time (unix timestamp), ID user, URL (to which the user has passed), number of seconds (how much time the user spent).

Implement calculation of the average time spent by the user (ID User) on the page.  
Put the result into a CSV file with “avg\_” prefix (i.e. avg\_file1.csv) and with header ordered by ID User in ASC order. Fields and order described below: ID user, URL, Average

Calculation of average time should take time of session into account, and provide statistic for each day separately. For each day additional string should be put in file to separate dates (see example below).

In case if session was started on one day and ended on another – for each day we need to count part of session time, which was spend on each day from particular session and this amount should be distributed between this dates.

For example, if we have only string

1455839970,user1,http://ru.wikipedia.org,100

Then in output file we need to create 4 rows:

18-FEB-2016

1455839970,user1,http://ru.wikipedia.org,30

19-FEB-2016

1455839970,user1,http://ru.wikipedia.org,70

**Sample Input:**

1455812018,user2,http://ru.wikipedia.org,100  
1455812019,user10,http://hh.ru,30  
1455812968,user3,http://google.com,60  
1455812411,user10,http://hh.ru,90  
1455812684,user3,http://vk.com,50

**Sample Output:**

18-FEB-2016

user2,https://ru.wikipedia.org,100  
user3,http://google.com,60  
user3,http://vk.com,50  
user10,http://hh.ru,60

* Program should be created in OOP concept.
* Each file should be processed in independent process.
* Up to 10 files should be processed in parallel; other files should be kept in queue and processed by first free process.
* Incoming files can be deleted or moved if required.
* Input and output folders should be pointed in property file.
* Program should starts from command line and work until killed or stopped (stop functionality are welcome).
* Instruction how to run the program is required (if we need to run run.bat/run.sh, ant or maven.   
   or any other instruction to run program from command line)

## Additional Tasks:

### XML task

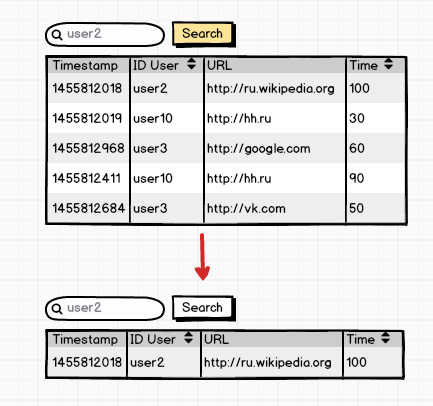
Design and implement input and output formats in XML:

* XSD should be created
* Program should support XML and CSV format both in this case

### UI task

Design and implement UI in order to show user log data.

* Should be possible to search in table via ‘ID User’, data should be filtered out after click on ‘Search’ button. See screenshots below.
* Should be possible to sort via ID User and Time in ASC/DESC order.
* Sort and search should be implemented on client side.



### JDBC task

Implement the same task but with usage of DataBase. All data should be storied in relevant table structure. If UI task is done, all data for UI should be gathered from database, and search should be performed on server side.