Assignment

All tasks should be written in Python 3.x (preferred) or another familiar language.

Scripts should be saved in separate files (workbooks). Please add readable instruction of executing (simple readme file).

It would be appreciated if you put your solution on a platform that supports version system control (github, gitlab, bitbucket, gitkraken, sourceforge...)

The following tasks are to create simple ETL scripts that will result in datasets based on 3 input CSV files.

1. **Task 1**

Load received files:

a. test.csv which contains
   i. id - ID of a test
   ii. student_id - ID of a student who solved the test
   iii. class_id - ID of the class
   iv. created_at - date of test creation by a teacher
   v. updated_at - date of last test update (by student or system)
   vi. last_event_time - last event time on test
   vii. overall_score - overall score obtained by student
   viii. test_status - test status
   ix. institution_id - ID of institution (school)
   x. authorized_at - date when the student started solving the test
   xi. confidence_level - level of confidence
   xii. speaking_score - score for speaking part of test
   xiii. writing_score - score for writing part of test
   xiv. reading_score - score for reading part of the test
   xv. listening_score - score for listening part of test
   xvi. test_level_id - level of test (difficulty of test)
   xvii. licence_id - licence id

b. test_level.csv
   i. id - id of test level
   ii. name - name of level
   iii. displayName - displayed name of level
   iv. created_at - date of create level
   v. updated_at - date of update of level

c. class.csv
   i. id - id of class
   ii. institution_id - id of institution
   iii. owner_id - id of owner

iv. name - name
v. created_at - date of create class
vi. updated_at - date of update class
vii. teaching_hours - teaching hours
viii. latest_test_time - latest test time
ix. has_student_with_scored_test - are in class students with scored tests?

to your work catalogue.

2. **Task 2**

Check if files are correct according to data in columns etc. If not please delete these rows. Will be in the plus if you do this as a simple script - not manually.

3. **Task 3**

Create first final dataset which will contain information about frequency of tests utilization by classes. The dataset has to have structure as follows:

a. test_utilization.csv
    a. class_id
    b. class_name
    c. teaching_hours
    d. test_id
    e. test_level
    f. test_created_at
    g. test_authorized_at
    h. class_test_number !!!! - this should be calculated - described below

In this task you have to enumerate each solved test in class and save these numbers in class_test_number. Of course all tests taken to this dataset should be authorized (column authorized_at != NULL).

class_id	class_name	teaching_hours	test_id	test_created_at	test_authorized_at	test_level	class_test_number
aaa	aaa	50	1	2019-01-03	2019-01-04	1	1
aaa	aaa	50	2	2019-01-03	2019-01-04	1	2
aaa	aaa	50	3	2019-01-03	2019-01-04	1	3
bbb	bbb	50	4	2019-01-03	2019-01-04	1	1
bbb	bbb	50	5	2019-02-01	2019-02-01	2	2
bbb	bbb	50	6	2019-02-01	2019-02-03	2	3
bbb	bbb	50	7	2019-02-01	2019-02-03	2	4

4. **Task 4**

Create a second final dataset which will contain information about average overall scores for tests in classes. The dataset has to have a structure as follows:

b. test_average_scores.csv
    a. class_id
    b. class_name
    c. teaching_hours
    d. test_created_at
    e. test_authorized_at
    f. avg_class_test_overall_score !!!! - this should be calculated - described below

In this task you have to calculate the average overall score for a particular class. Please take just tests marked as SCORING_SCORED in test_status column. Of course all tests taken to this dataset should be authorized (column authorized_at != NULL).

class_id	class_name	teaching_hours	test_created_at	test_authorized_at	avg_class_test_overall_score
aaa	aaa	50	2019-01-03	2019-01-04	5.4
bbb	bbb	50	2019-01-03	2019-01-04	7.2

5. **Task 5**

   Please save and store these two datasets produced in the 2 earlier tasks in the catalogue from which you retrieved data for calculation.

   File name for first dataset: **test_utilization**
   File name for second dataset: **test_average_scores**

6. **Task 6 - not obligatory**

   Please prepare a script which allow you to load the earlier prepared datasets to tables in a database.

   In this task we would like to see how you will solve these kind of problems. This script does not have to work.

7. **Nice to have**

1. If you are familiar with the testing of code, please cover your code by unit tests.