SCALE FOR PROJECT PISCINE PYTHON DATA SCIENCE (/PROJECTS/PISCINE-PYTHON-DATA-SCIENCE) / DAY 05 (/PROJECTS/PISCINE-PYTHON-DATA-SCIENCE-DAY-05)

You should evaluate 1 student in this team

★

Git repository

`git@vogsphere.kzn.21-school.ru:vogsphere/intra-uuid-83d485b`

Introduction

The methodology of School 21 makes sense only if
peer-to-peer assessments are done seriously.
This document will help you to do it properly.

- Please, stay courteous, polite, respectful and constructive in all
communications during this assessment. The bond of trust
between community 21 and you depends on it.

- Highlight possible malfunctions of the work done by the person
and take the time to discuss and debate it.

- Keep in mind that sometimes there can be differences
in interpretation of the tasks and the scope of features.
Please, stay open-minded to the vision of the other.

Guidelines

- Evaluate only the files that are on the GIT
repository of the student or group.

- Doublecheck that the GIT repository is the one
corresponding to the student
or the group as long as to the project.

- Meticulously check that nothing malicious has been used

to mislead you and have you assess something except
the content of the official repository.

- If you have not finished the project yet, it is compulsory
to read the entire instruction before starting the review.

- Use the special flags in the scale to report an empty
or non-functional solution as long as a case of cheating.
In these cases, the assessment is completed
and the final grade is 0 (or in a case of cheating is -42).
However, except for a case of cheating, you are encouraged to continue
reviewing the project to identify the problems that caused the situation
in order to avoid them for the next assessment.

- You must stop giving points from the first wrong exercise
even if the following exercises are correct.

Attachments

subject.pdf (https://cdn.intra.42.fr/pdf/pdf/49869/en.subject.pdf)

surname.json (/uploads/document/document/8786/surname.json)

feed-views.log (/uploads/document/document/8787/feed-views.log)

auto.csv (/uploads/document/document/8788/auto.csv)

Preliminaries

Respect the rules

- The repository contains the work of the student (or group).
- The student is able to explain their work
at any time during the assessment.
- The general rules and any rules specific to the day are
respected throughout the assessment.

Yes No

Piscine Python | Data Science MODULE 05

Any hardcoded result is worth zero for the exercise.

Exercise 00 – Load and save

- Run all the cells in the notebook, they should work without errors
- Run df.count(), the result should be exactly like this:
user 1072
dtype: int64
- Run df2 = pd.read_csv("d05/data/feed-views-semicolon.log", sep=';'),
it should work without errors
- Run after that df2.head(), the result should be like this:
date_time user
0 2020-04-17 12:01:08.463179 artem
1 2020-04-17 12:01:23.743946 artem
2 2020-04-17 12:35:52.735016 artem
3 2020-04-17 12:36:21.401412 oksana
4 2020-04-17 12:36:22.023355 oksana
In all other cases, the test is failed.

☑ Yes ☒ No

Exercise 01 – Basic operations

- Run all the cells in the notebook, they should work without errors
- Run views.info(), it should include:
datetime 1076 non-null datetime64[ns]
- The result of views.count() must be:
datetime 1076
year 1076
month 1076
day 1076
hour 1076
minute 1076
second 1076
daytime 1076
dtype: int64
- The result of views.daytime.value_counts() should be this:
evening 509
afternoon 252
early evening 145
night 129
morning 36
early morning 5
Name: daytime, dtype: int64
- The result of views.loc[views.daytime == 'night'].hour.idxmax()
should be 'konstantin'
- The result of views.loc[views.daytime == 'morning'].hour.idxmin()
should be 'alexander'
- The result of views.hour.mode() should be 22
- The result of views.daytime.mode() should be evening
- The value of iqr should be 9.0
In all other cases, the test is failed.

Yes No

Exercise 02 – Preprocessing

- Run all the cells in the notebook, they should work without errors
- Run df2 = pd.read_json('data/auto.json', orient='records'),
it should work without errors
- Run df2 = df2.count(), the result should be:
CarNumber 725
Refund 725
Fines 725
Make 725
Model 716
dtype: int64
- Run df2['Fines'].mean(), the result should be:
8594.586466165412
- Run df2['Refund'].mean(), the result should be:
1.5172413793103448
In all other cases, the test is failed.

Yes No

Exercise 03 – Selects and aggregations

- Run all the cells in the notebook, they should work without errors
- The dimensions of df.loc[df['Model'].isin(models)]
is 593 rows × 4 columns? where models is the list
containing 'Focus' and 'Corolla'
- Run df.groupby(['Make', 'Model']).agg('Fines').count(),
the result should be:
Make Model
Ford Focus 575
Mondeo 6
Skoda Octavia 48
Toyota Camry 16
Corolla 18
Volkswagen Golf 20
Jetta 6
Passat 22
Touareg 5
Name: Fines, dtype: int64
- The top-3 car numbers by the number of fines are:
Y7689C197RUS, 92928M178RUS, 7788KT197RUS?
- The top-1 car number by the sum of fines is X758HY197RUS?
In all other cases, the test is failed.

Yes No

Exercise 04 – Enrichment and transformations

- Run all the cells in the notebook, they should work without errors
- All the floats are displayed with only 2 decimals
- The result of concat_rows.count() is
CarNumber 925
Refund 925
Fines 925
Make 925
Model 914
dtype: int64
- The result of fines.count() in the code of the student is
CarNumber 925
Refund 925
Fines 925
Make 925
Model 914
Year 925
dtype: int64
- The values of the SURNAME column in the owners do not have
unwanted characters ('[', ']', '"')
- Run len(owners), the result should be 531 before deleting
20 samples and adding 3 more
- Run len(fines), the result should be 930 after enriching dataframe
- The result of the first merge should be 900 rows × 7 columns
in dimensions
- The result of the second merge should be 933 rows × 7 columns
in dimensions
- The result of the third merge should be 930 rows × 7 columns
in dimensions
- The result of the fourth merge should be 903 rows × 7 columns
in dimensions
- The result of the pivot_table has the same structure as it is
in the subject, the values can be different
In all other cases, the test is failed.

Yes No

Exercise 05 – Pandas optimizations

- Run all the cells in the notebook, they should work without errors
- The result of optimized_df.info(memory_usage='deep')
should be like this:
0 CarNumber 930 non-null category
1 Refund 930 non-null int8
2 Fines 930 non-null float32
3 Make 930 non-null category

4 Model 919 non-null category
5 Year 930 non-null int16
6 strange 930 non-null float32
- Run df (the initial one that was cleaned), you should get the error:
NameError: name 'df' is not defined
In all other cases, the test is failed.

☑ Yes ☒ No

Ratings

Don't forget to check the flag corresponding to the defense

✔ Ok ★ Outstanding project

Empty work | No author file | Invalid compilation | Norme | Cheat | Crash

Leaks | Forbidden function

Conclusion

Leave a comment on this evaluation

Finish evaluation