Documentation:

Refactory plan:

As we already make full-coverage functions, we counted the functions for the aim of refactory classes and cont the repeated functions for reduce functions or reconstruct functions.

1. Identify functionality:

Notebook:Computing final:

Block:

Read data by file path

End:

**Data cleaning**:

Block:

drop\_columns(dataframe, columns\_to\_drop):

Drop columns, drop entire columns by input columns list[]

End:

Block:

detect\_missing\_values(dataframe):

Counting for all the missing value of the columns in input Dataframe

End:

None value handled by try catch block

Block:

#convert string to designed pattern or num

convert\_comma\_to\_dot(column):

convert\_price(column):

convert\_engine(column):

convert\_kms(column):

convert\_seats(seats\_str):

keep\_first\_word(input\_string):

Usage: df[‘expected\_column’].apply(convert\_)

#convert string to num by include and

extract\_first\_integer(ownership\_str):

process\_ownership(df):

Convert string format to desired format

End:

Block:

#draw correlation heatmap

correlation\_heatmap(df, columns):

Draw correlation heatmap for input df by input columns list[]

End:

Block:

detect\_missing\_values(dataframe):

Counting for all the missing value of the columns in input Dataframe

End:

Block:

columns\_binary(df, column\_names: list, true\_value, false\_value):

Convert to binary value

End:

Block:

detect\_missing\_values(dataframe):

Counting for all the missing value of the columns in input Dataframe

End: