FIT5196-S1-2023 Assessment 3

This is an individual assessment and worth 30% of your total mark for FIT5196.

Due date: See Moodle

For this assessment, you are required to write Python code (*any Python packages are allowed)* to integrate several datasets into one single schema. The input and output of this assessment are shown below:

Table 1. The input and output of the task

Inputs	**Output file**	**Jupyter Notebook and .py files**
<student_no>.xml (file), <student_no>.json (file), Vic_suburb_boundary (directory), Vic_GTFS_data (directory), Lga_to_suburb.pdf (file), house.speakingsame.com (website) (you need to figure out how to scrape this website by yourself. *Check week 9 materials for more information. Any Python package is allowed*)	<student_no>_A3_solution.csv	<student_no>_ass3.ipynb <student_no>_ass3.py

The .py file should be generated from your Jupyter Notebook file and it will be used for plagiarism checks. (see appendix)

Each of you is given several datasets in various formats and the initial data is about housing information in Victoria, Australia. **You can find your own dataset (different for each student) here**, and you can find **the supplementary data (same for all students)** here. In this assignment, you need to perform the following tasks.

Task 1: Data Integration (55%)

In this task, you are required to integrate the input datasets from several sources into one dataset with the following schema.

Table2. Description of the final schema. See sample output for more information about columns datatypes and value formats. (Read what default values mean in Note3)

COLUMN	DESCRIPTION
property_id	A unique id for the property
lat	The property latitude
lng	The property longitude
addr_street	The property address
suburb (15%)	The property suburb. **Default value: "NA"**
lga (10%)	The property **local government area** (LGA). **Default value: "NA"**
closest_train_station_id (5%)	The closest train station to the property using Haversine distance. **Default value: "NA"**
distance_to_closest_train_station (5%)	The Haversine distance from the closest train station to the property. **Default value: "NA"**
travel_min_to_MC (15%)	So we assumed that there was a big vaccination centre placed at the Melbourne Central building. This column is the **rounded average travel time** (minutes) of the **direct journeys** (see the definition of the direct journeys in **Note 2**) from the closest train station to the "Melbourne Central" station on all the weekdays (Monday to Friday) **departing** between **7 to 9 am**. For example, if there are 3 direct trips departing from the closest train station to the Melbourne Central station on weekdays between 7-9 am and each takes 6, 7, and 8 minutes respectively, then the value of this column for the property should be round((6+7+8)/3). If **there are no direct journeys** between the closest station and Melbourne Central Station, the value should be set to **"no direct trip is available"**. **If the closest station to a property is Melbourne Central Station itself, then the value should be set to 0**. **Default value: "NA"**
direct_journey_flag (5%)	A Boolean attribute indicating whether there is a direct journey to the Melbourne Central station from the closest station between 7-9 am on all the weekdays (i.e., the service operates all days from Monday to Friday.). This flag is 1 if there is a direct trip (i.e. no transfer between trains is required to get from the closest train station to the Melbourne Central station) and 0 otherwise. **If the closest station to a property is**

	Melbourne Central Station itself, then the value should be set to 1. Default value: "NA"
number_of_houses (5%)	The number of houses in the property suburb must be scrapped from the house.speakingsame.com website. **Default value: "NA"**
number_of_units (5%)	The number of units in the property suburb must be scrapped from the house.speakingsame.com website. **Default value: "NA"**
municipality (5%)	The municipality of the property suburb must be scrapped from the house.speakingsame.com website. **Default value: "NA"**
aus_born_perc (5%)	The percentage of the Australian-born population in the property suburb must be scrapped from the house.speakingsame.com website. **Default value: "NA"**
median_income (5%)	The median income of the population in the property suburb must be scrapped from the house.speakingsame.com website. **Default value: "NA"**
median_house_price (5%)	The median 'house' price in the property suburb must be scrapped from the house.speakingsame.com website. **Default value: "NA"**
population (5%)	The population in the property suburb must be scrapped from the house.speakingsame.com website. **Default value: "NA"**

Note 1: the output CSV file must have the exact same columns as specified on the schema. Please note that the output files which are not in the correct format, as specified in the integrated schema, will not be marked.

Note 2: direct journey means that you can reach Melbourne Central Station without changing your train at any point in the journey. So, when you board the train at the closest station, you can directly go to the Melbourne Central Station without moving to another vehicle.

Note 3: if you decide not to calculate any of the required columns, then you must still have that column in your final dataframe with all the values as the 'default value'. Please note that the output files which are not in a correct format, as specified in the integrated schema, won't be marked.

Note 4: No external data is allowed to calculate the values of the integrated schema. For example, to calculate the suburb, you can only use the shape files provided in Google Drive. The only external source of information is http://house.speakingsame.com/suburb.php website.

Note 5: shapefile data and lga_to_suburb.pdf data can be outdated and incorrect. You don't need to fix them or check their validity.

Note 6: for Haversine distance (link), use 6378 km as the radius of the earth.

Note 7: for more information about GTFS files read here (link).

Note 8: In Table 2, the numbers in front of some of the columns in the format of (a%) are the allocated mark associated with that column. For example, column "suburb" carries 15% of the total output mark of task 1. Also, please note that we are aware that the summation of percentages is 90%. The other 10% goes to the issue(s) that may appear during data integration tasks and you should find and resolve them.

Task 2: data reshaping (20%)

In this task, you need to study the effect of different normalization/transformation methods (i.e. standardisation, min-max normalization, log, power, box-cox transformation) on the columns scrapped from the website (i.e., **number_of_houses**, **number_of_units**, **population, aus_born_perc, median_income, median_house_price**) and observe and explain their effect assuming **we want to develop a linear model to predict the "median_house_price" using the other attributes**. When reshaping the data, we have two main criteria. First, we want our features to be on the same scale and second, we want our features to have as much linear relationship as possible with the target variable (i.e., **median_house_price**). You need to first explore the data to see if any scaling or transformation is necessary (if yes why? and if not, also why?) and then perform appropriate actions and document your results and observations. Please note that you don't need to actually build a linear model and just need to prepare the data for a linear regression model.

Task 3: Documentation (25%)

The main focus of the documentation would be on the quality of your explanation on task 2 but similar to the previous assignments, your Notebook file should be in a decent format with proper sections and subsections.

Deliverables

You must submit the following files on Moodle to have a successful submission. **We will not mark incomplete submissions!**

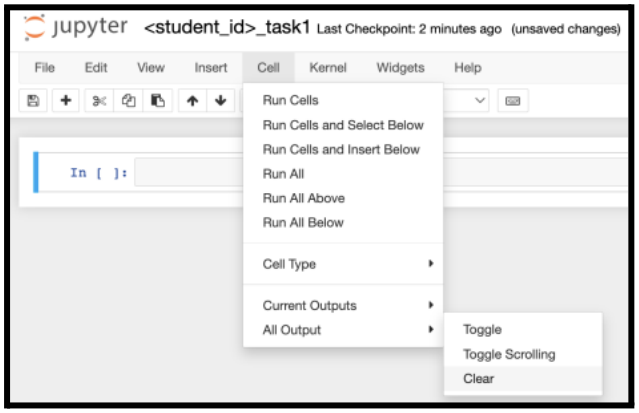
1. <student_no>_A3_solution.csv
2. <student_no>_ass3.ipynb (which contains both task 1 and task 2 documentation each having their own sections) (having a ToC is highly recommended) **(all outputs must be preserved in the .ipynb)**
3. <student_no>_ass3.py (this file is used for plagiarism checking)

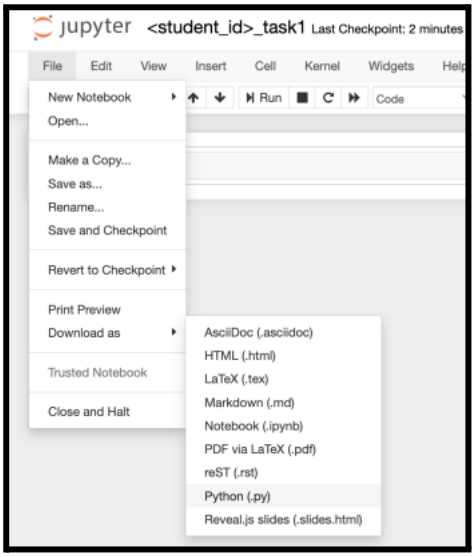
Please zip all above files in <student_no>_ass3.zip (do not use any other format: rar, 7z, etc.)

Appendix

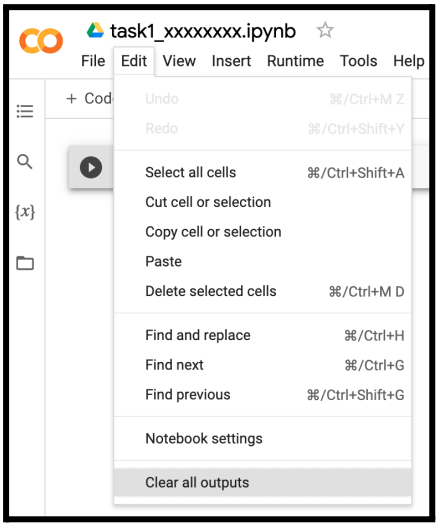
1. To generate the .py file, you need to clear all the cell outputs, and then download it.

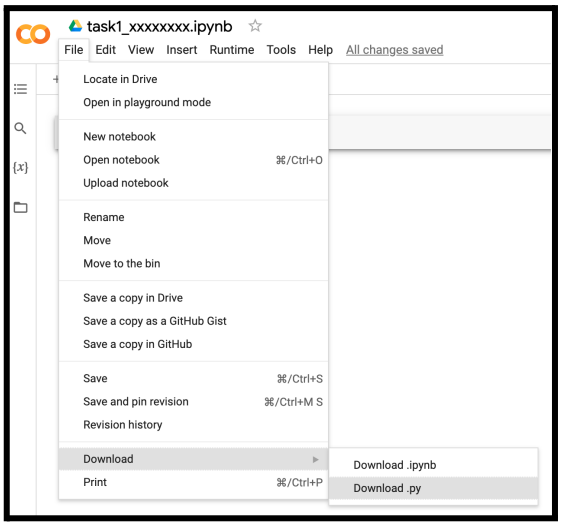
   Jupyter notebook:





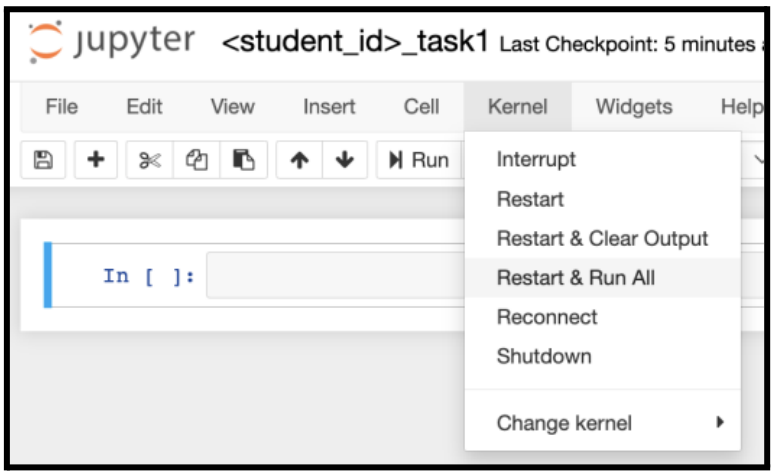
   Google colab:





2. To generate cell outputs before submitting an .ipynb file, you need to run all the cells before saving your file.

   Jupyter notebook:



   Google Colab: