Data Wrangling Contest

Working with *pandas* Data Frames

Last updated: 20 March 2023
by Dr J Girish

Task

We will study the dataset called nycflights13. It gives information about all 336,776 flights that departed in 2013 from the three New York (in the US) airports (EWR, JFK, and LGA) to destinations in the United States, Puerto Rico, and the American Virgin Islands.

Our aim is to use *pandas* to come up with results equivalent to those that correspond to example SQL queries.

Create a single Jupyter/IPython notebook (see the *Artefacts* section below for all the requirements), where you perform what follows.

1. Establish a connection with a new SQLite database on your disk.
2. Export all the CSV files to the said database.
3. For each of the SQL queries below (each query in a separate section), write the code that yields equivalent results using *pandas* only and explain – in your own words – what it does.

Here are the SQL queries:

1.
```
SELECT DISTINCT engine FROM planes
```

2.
```
SELECT DISTINCT type, engine FROM planes
```

3.
```
SELECT COUNT(*), engine FROM planes GROUP BY engine
```

4.
```
SELECT COUNT(*), engine, type FROM planes
GROUP BY engine, type
```

5.
```
SELECT MIN(year), AVG(year), MAX(year), engine, manufacturer
FROM planes
GROUP BY engine, manufacturer
```

6.
```
SELECT * FROM planes WHERE speed IS NOT NULL
```

7.
```
SELECT tailnum FROM planes
WHERE seats BETWEEN 150 AND 190 AND year >= 2012
```

8.
```
SELECT tailnum, manufacturer, seats FROM planes
WHERE manufacturer IN ("BOEING", "AIRBUS", "EMBRAER") AND seats>390
```


9.
```
SELECT DISTINCT year, seats  FROM planes
WHERE year >= 2012 ORDER BY year ASC, seats DESC
```

10.
```
SELECT DISTINCT year, seats  FROM planes
WHERE year >= 2012 ORDER BY seats DESC, year ASC
```

11.
```
SELECT manufacturer, COUNT(*) FROM planes
WHERE seats > 200 GROUP BY manufacturer
```

12.
```
SELECT manufacturer, COUNT(*) FROM planes
GROUP BY manufacturer HAVING COUNT(*) > 10
```

13.
```
SELECT manufacturer, COUNT(*) FROM planes
WHERE seats > 200 GROUP BY manufacturer HAVING COUNT(*) > 10
```

14.
```
SELECT manufacturer, COUNT(*) AS howmany
FROM planes
```
```
GROUP BY manufacturer
ORDER BY howmany DESC LIMIT 5
```

15.
```
SELECT
    flights.*,
    planes.year  AS plane_year, planes.speed AS
    plane_speed, planes.seats AS plane_seats
FROM flights LEFT JOIN planes ON flights.tailnum=planes.tailnum
```

16.
```
SELECT planes.*, airlines.* FROM
(SELECT DISTINCT carrier, tailnum FROM flights) AS cartail
INNER JOIN planes ON cartail.tailnum=planes.tailnum
INNER JOIN airlines ON cartail.carrier=airlines.carrier
```

17.
```
SELECT
    flights2.
    *,
    atemp,
    ahumid
FROM (
    SELECT * FROM flights WHERE origin='EWR'
) AS flights2
  LEFT
 JOIN (
SELECT
        year, month, day,
        AVG(temp) AS atemp,
        AVG(humid) AS ahumid
    FROM weather
    WHERE origin='EWR'
    GROUP BY year, month, day
```

Do not include full outputs of the SQL queries in the report!

Artefacts

The solution to the task must be included in a single Jupyter/IPython notebook (an .ipynb file) running against a Python 3 kernel.

At the start of the notebook, you need to provide: your **name**, **student number**, and **email address**.