sql-session-2

September 23, 2024

```
[1]: import pandas as pd
     import sqlite3 as sql
```

```
[2]: db = 'factbook.db'
```

```
[3]: conn = sql.connect(db)
```

```
[4]: conn
```

```
[4]: <sqlite3.Connection at 0x18723a054e0>
```

```
[5]: def read_q(q):
         return pd.read_sql_query(q, conn)
```

```
[6]: read_q('select * from sqlite_master')
```

```
[6]:     type             name         tbl_name  rootpage  \
     0  table  sqlite_sequence  sqlite_sequence         3
     1  table            facts            facts        47
     2  table           cities           cities         2

                                                  sql
     0              CREATE TABLE sqlite_sequence(name,seq)
     1  CREATE TABLE "facts" ("id" INTEGER PRIMARY KEY…
     2  CREATE TABLE cities (\n        id integer prim…
```

```
[7]: read_q('select * from facts')
```

```
[7]:       id code             name       area  area_land  area_water    population  \
     0      1   af      Afghanistan   652230.0   652230.0         0.0  3.256434e+07
     1      2   al          Albania    28748.0    27398.0      1350.0  3.029278e+06
     2      3   ag          Algeria  2381741.0  2381741.0         0.0  3.954217e+07
     3      4   an          Andorra      468.0      468.0         0.0  8.558000e+04
     4      5   ao           Angola  1246700.0  1246700.0         0.0  1.962535e+07
     ..   …    …              …          …          …           …
     256  257   zh   Atlantic Ocean        NaN        NaN         NaN           NaN
     257  258   xo     Indian Ocean        NaN        NaN         NaN           NaN
```

```
258  259   zn   Pacific Ocean        NaN        NaN         NaN             NaN
259  260   oo  Southern Ocean        NaN        NaN         NaN             NaN
260  261   xx           World        NaN        NaN         NaN    7.256490e+09

     population_growth  birth_rate  death_rate  migration_rate
0                 2.32       38.57       13.89            1.51
1                 0.30       12.92        6.58            3.30
2                 1.84       23.67        4.31            0.92
3                 0.12        8.13        6.96            0.00
4                 2.78       38.78       11.49            0.46
..                 …          …          …              …
256                NaN         NaN         NaN             NaN
257                NaN         NaN         NaN             NaN
258                NaN         NaN         NaN             NaN
259                NaN         NaN         NaN             NaN
260               1.08       18.60        7.80             NaN

[261 rows x 11 columns]
```

[9]:

```
read_q('select * from cities')
```

[9]:

```
      id         name  population  capital  facts_id
0      1   Oranjestad       37000        1       216
1      2  Saint John'S       27000        1         6
2      3    Abu Dhabi      942000        1       184
3      4        Dubai     1978000        0       184
4      5      Sharjah      983000        0       184
..   …          …          …       …       …
392  393      Mbabane       66000        1       167
393  394        Sanaa     2419000        1       193
394  395         Aden      784000        0       193
395  396       Lusaka     1802000        1       194
396  397       Harare     1542000        1       195

[397 rows x 5 columns]
```

1 Task 1

- write a query that shows all the information about 'Pakistan'

[8]:

```
read_q('select * from facts where name="Pakistan"')
```

[8]:

```
    id code      name    area  area_land  area_water  population  \
0  132   pk  Pakistan  796095     770875       25220   199085847

   population_growth  birth_rate  death_rate  migration_rate
0               1.46       22.58        6.49            1.54
```

- return all cities of country Pakistan

[10]:
```
read_q("""SELECT *
FROM cities
WHERE facts_id=132 """)
```

[10]:
```
    id        name  population  capital  facts_id
0  273     Karachi    13876000        0       132
1  274      Lahore     7566000        0       132
2  275  Faisalabad     3038000        0       132
3  276  Rawalpindi     2164000        0       132
4  277      Multan     1775000        0       132
5  278   Islamabad      919000        1       132
```

```
--------------------------------------------------------------------------------------------------
Primary key is a unique identifier for every record/row. It can not be null and
should be unique.

Foreign key is a primary key in its own table. It is being referenced in another
table to create a relationship.
```

ONE - TO - MANY - one Country can have Multiple Cities

[12]:
```
country = read_q('select * from facts')
city = read_q('select * from cities')
```

[13]:
```
country.head()
```

[13]:
```
   id code         name       area  area_land  area_water  population  \
0   1   af  Afghanistan   652230.0   652230.0         0.0  32564342.0
1   2   al      Albania    28748.0    27398.0      1350.0   3029278.0
2   3   ag      Algeria  2381741.0  2381741.0         0.0  39542166.0
3   4   an      Andorra      468.0      468.0         0.0     85580.0
4   5   ao       Angola  1246700.0  1246700.0         0.0  19625353.0

   population_growth  birth_rate  death_rate  migration_rate
0               2.32       38.57       13.89            1.51
1               0.30       12.92        6.58            3.30
2               1.84       23.67        4.31            0.92
3               0.12        8.13        6.96            0.00
4               2.78       38.78       11.49            0.46
```

[14]:
```
city.head()
```

[14]:
```
   id           name  population  capital  facts_id
0   1     Oranjestad       37000        1       216
1   2   Saint John'S       27000        1         6
2   3      Abu Dhabi      942000        1       184
```

```
3   4         Dubai     1978000        0      184
4   5       Sharjah      983000        0      184
```

```
[15]: city['facts_id'] = city['facts_id'].astype(int)
```

```
[16]: df = pd.merge(country, city, left_on='id', right_on='facts_id', how='inner')
```

```
[17]: df
```

```
[17]:      id_x code                      name_x       area  area_land  area_water  \
      0       1   af                 Afghanistan   652230.0   652230.0         0.0
      1       2   al                     Albania    28748.0    27398.0      1350.0
      2       3   ag                     Algeria  2381741.0  2381741.0         0.0
      3       3   ag                     Algeria  2381741.0  2381741.0         0.0
      4       4   an                     Andorra      468.0      468.0         0.0
      ..    …    …                           …         …         …          …
      392   243   gq                        Guam      544.0      544.0         0.0
      393   245   cq  Northern Mariana Islands      464.0      464.0         0.0
      394   246   rq                 Puerto Rico    13791.0     8870.0      4921.0
      395   247   vq              Virgin Islands     1910.0      346.0      1564.0
      396   255   wi              Western Sahara   266000.0   266000.0         0.0

           population_x  population_growth  birth_rate  death_rate  migration_rate  \
      0      32564342.0               2.32       38.57       13.89            1.51
      1       3029278.0               0.30       12.92        6.58            3.30
      2      39542166.0               1.84       23.67        4.31            0.92
      3      39542166.0               1.84       23.67        4.31            0.92
      4         85580.0               0.12        8.13        6.96            0.00
      ..              …                  …           …           …              …
      392      161785.0               0.54       16.82        5.12            6.34
      393       52344.0               2.18       18.32        3.71            7.16
      394     3598357.0               0.60       10.86        8.67            8.15
      395      103574.0               0.59       10.31        8.54            7.67
      396      570866.0               2.82       30.24        8.34             NaN

           id_y             name_y  population_y  capital  facts_id
      0       6              Kabul       3097000        1         1
      1      10             Tirana        419000        1         2
      2       7            Algiers       2916000        1         3
      3       8               Oran        783000        0         3
      4      12  Andorra La Vella         23000        1         4
      ..    …                  …             …        …        …
      392   143            Hagatna        169000        1       243
      393    99             Saipan         56000        1       245
      394   295           San Juan       2475000        1       246
      395   388  Charlotte Amalie          60000        1       247
      396   391           Laayoune        237000        0       255
```

```
[397 rows x 16 columns]
```

[18]:

```
df[df['id_x'] == 132]
```

[18]:

```
     id_x code    name_x      area  area_land  area_water  population_x  \
250   132   pk  Pakistan  796095.0   770875.0     25220.0   199085847.0
251   132   pk  Pakistan  796095.0   770875.0     25220.0   199085847.0
252   132   pk  Pakistan  796095.0   770875.0     25220.0   199085847.0
253   132   pk  Pakistan  796095.0   770875.0     25220.0   199085847.0
254   132   pk  Pakistan  796095.0   770875.0     25220.0   199085847.0
255   132   pk  Pakistan  796095.0   770875.0     25220.0   199085847.0

     population_growth  birth_rate  death_rate  migration_rate  id_y  \
250               1.46       22.58        6.49            1.54   273
251               1.46       22.58        6.49            1.54   274
252               1.46       22.58        6.49            1.54   275
253               1.46       22.58        6.49            1.54   276
254               1.46       22.58        6.49            1.54   277
255               1.46       22.58        6.49            1.54   278

         name_y  population_y  capital  facts_id
250     Karachi      13876000        0       132
251      Lahore       7566000        0       132
252  Faisalabad       3038000        0       132
253  Rawalpindi       2164000        0       132
254      Multan       1775000        0       132
255   Islamabad        919000        1       132
```

----

2 JOINS

```
using a common column (which is present in both tables), you can join two or more
tables to retrieve information.
```

Return Pakistan and its cities

[19]:

```
# country name, city name
```

[20]:

```
read_q(""" select f.name, c.name
from facts f inner join
cities c on
f.id = c.facts_id
""")
```

[20]:

```
                     name          name
0                   Aruba    Oranjestad
1     Antigua and Barbuda  Saint John'S
2    United Arab Emirates     Abu Dhabi
3    United Arab Emirates         Dubai
4    United Arab Emirates       Sharjah
..                    …             …
392             Swaziland       Mbabane
393                 Yemen         Sanaa
394                 Yemen          Aden
395                Zambia        Lusaka
396              Zimbabwe        Harare

[397 rows x 2 columns]
```

[21]:

```
read_q(""" select f.name, c.name
from facts f inner join
cities c on
f.id = c.facts_id
where f.id = 132
""")
```

[21]:

```
       name        name
0  Pakistan     Karachi
1  Pakistan      Lahore
2  Pakistan  Faisalabad
3  Pakistan  Rawalpindi
4  Pakistan      Multan
5  Pakistan   Islamabad
```

3 Task 2

- return evry country and its capital city

[36]:

```
read_q(""" select f.name Country, c.name Cities
from facts f inner join
cities c on
f.id = c.facts_id
where c.capital = 1 and f.id = 132
""")
```

[36]:

```
    Country     Cities
0  Pakistan  Islamabad
```

4 Task 3

- return the name of the country and its relative city and its population which has a population of more than 10 million.

[43]:

```
read_q(""" select f.name Country, c.name City, c.population City_population
from facts f inner join
cities c on
f.id = c.facts_id
where c.population > 10000000
""")
```

[43]:

```
          Country                              City  City_population
0       Argentina                      Buenos Aires         13528000
1      Bangladesh                             Dhaka         15391000
2          Brazil                         Sao Paulo         19924000
3           China                          Shanghai         20208000
4           China                           Beijing         15594000
5           China                         Guangzhou         10849000
6           Egypt                             Cairo         11169000
7          France         Marseille-Aix-en-Provence         14890100
8           India                         New Delhi         22654000
9           India                            Mumbai         19744000
10          India                           Kolkata         14402000
11          Japan                             Tokyo         37217000
12          Japan                        Osaka-Kobe         11494000
13         Mexico                       Mexico City         20446000
14        Nigeria                             Lagos         11223000
15       Pakistan                           Karachi         13876000
16    Philippines                            Manila         11862000
17         Russia                            Moscow         11621000
18         Turkey                          Istanbul         11253000
19  United States                   New York-Newark         20352000
20  United States  Los Angeles-Long Beach-Santa Ana         13395000
```

5 Task 4

- Country name, Total population, Urban Areas Population

[55]:

```
read_q(""" select f.name Country, f.population Total_Population, sum(c.
 ↪population) Urban_population
from facts f inner join cities c
on f.id = c.facts_id
group by f.name
order by 3 desc
""")
```

```
[55]:                                        Country  Total_Population  \
      0                                        India        1251695584
      1                                        China        1367485388
      2                                United States         321368864
      3                                        Japan         126919659
      4                                       Brazil         204259812
      ..                                         …                 …
      205            Falkland Islands (Islas Malvinas)              3361
      206                                   Anguilla             16418
      207                          Wallis and Futuna             15613
      208  Saint Helena, Ascension, and Tristan da Cunha              7795
      209                                      Palau             21265

           Urban_population
      0            82035000
      1            66849000
      2            60116000
      3            60077000
      4            38086000
      ..                …
      205              2000
      206              2000
      207              1000
      208              1000
      209              1000

      [210 rows x 3 columns]
```

6 Task 5

- find out the percentage of urban to total population

```
[58]: read_q(""" select f.name Country, f.population Total_Population, sum(c.
        ↪population) Urban_population,
      round((sum(c.population)/cast(f.population as float)) * 100, 3) per_Urban
      from facts f inner join cities c
      on f.id = c.facts_id
      group by f.name
      order by 3 desc
      """)
```

```
[58]:                                        Country  Total_Population  \
      0                                        India        1251695584
      1                                        China        1367485388
      2                                United States         321368864
      3                                        Japan         126919659
      4                                       Brazil         204259812
```

```
..                                               …               …
205              Falkland Islands (Islas Malvinas)              3361
206                                       Anguilla             16418
207                              Wallis and Futuna             15613
208  Saint Helena, Ascension, and Tristan da Cunha              7795
209                                          Palau             21265

     Urban_population  per_Urban
0            82035000      6.554
1            66849000      4.888
2            60116000     18.706
3            60077000     47.335
4            38086000     18.646
..                  …        …
205              2000     59.506
206              2000     12.182
207              1000      6.405
208              1000     12.829
209              1000      4.703

[210 rows x 4 columns]
```

```
[ ]:
```