Brian_Reppeto540Week_11_12

March 1, 2024

DSC 540 Week 11 & 12 Data Wrangling with Python:

Chapter 8

Author: Brian Reppeto 2/19/2024 Activity 11

[1]:
```
# import libraries

import sqlite3
```

[4]:
```
# create connection to petsdb

conn = sqlite3.connect("petsdb")
```

[5]:
```
# a tiny function to make sure the connection is successful
def is_opened(conn):
    try:
        conn.execute("SELECT * FROM persons limit 5")
        return True
    except sqlite3.ProgrammingError as e:
        print("Connection closed {}".format(e))
        return False
```

[6]:
```
# print if the connection is open true/false

print(is_opened(conn))
```

```
True
```

[7]:
```
# close connection

conn.close()
```

[8]:
```
# print if connection is closed or open

print(is_opened(conn))
```

```
Connection closed Cannot operate on a closed database.
False
```

Task 2

[9]:
```
# connect to petsdb

conn = sqlite3.connect("petsdb")
```

[10]:
```
# create a cursor object from the petsdb

cur = conn.cursor()
```

[11]:
```
# create a for loop that iterates over the results of a sql group by

for ppl, age in cur.execute("select count(*), age from persons group by age"):
    print("We have {} people aged {}".format(ppl, age))
```

```
We have 2 people aged 5
We have 1 people aged 6
We have 1 people aged 7
We have 3 people aged 8
We have 1 people aged 9
We have 2 people aged 11
We have 3 people aged 12
We have 1 people aged 13
We have 4 people aged 14
We have 2 people aged 16
We have 2 people aged 17
We have 3 people aged 18
We have 1 people aged 19
We have 3 people aged 22
We have 2 people aged 23
We have 3 people aged 24
We have 2 people aged 25
We have 1 people aged 27
We have 1 people aged 30
We have 3 people aged 31
We have 1 people aged 32
We have 1 people aged 33
We have 2 people aged 34
We have 3 people aged 35
We have 3 people aged 36
We have 1 people aged 37
We have 2 people aged 39
We have 1 people aged 40
We have 1 people aged 42
We have 2 people aged 44
We have 2 people aged 48
We have 1 people aged 49
We have 1 people aged 50
We have 2 people aged 51
```

```
We have 2 people aged 52
We have 2 people aged 53
We have 2 people aged 54
We have 1 people aged 58
We have 1 people aged 59
We have 1 people aged 60
We have 1 people aged 61
We have 2 people aged 62
We have 1 people aged 63
We have 2 people aged 65
We have 2 people aged 66
We have 1 people aged 67
We have 3 people aged 68
We have 1 people aged 69
We have 1 people aged 70
We have 4 people aged 71
We have 1 people aged 72
We have 5 people aged 73
We have 3 people aged 74
```

Task 3

[13]:
```
# create a for loop that iterates over the results of a sql group by

for ppl, age in cur.execute("select count(*), age from persons group by age␣
 ↪order by count(*) DESC"):
    print("Highest number of people {} came from {} age group".format(ppl, age))
    break
```

```
Highest number of people 5 came from 73 age group
```

Task 4

[17]:
```
# count the # of rows in the persons table where the last_name is NULL
# iterate through the result to print each row

result = cur.execute("select count (*) from persons where last_name IS null")
for row in result:
    print(row)
```

```
(60,)
```

Task 5

[21]:
```
# using a for loop interate over the results
# find out how many people own more than one pet

result = cur.execute("select count(*) from (select count(owner_id) from \
                      pets group by owner_id having count(owner_id) >1)")
for row in result:
```

```
    print("{} People whom have more than one pets".format(row[0]))
```

```
43 People whom have more than one pets
```

Task 6

[23]:
```
# using a for loop interate over the results
# find out how many pets recieved treatment

result = cur.execute("select count(*) from pets where treatment_done=1")
for row in result:
    print(f"The number of pets that had treatment: {row[0]}")
```

```
The number of pets that had treatment: 36
```

Task 7

[25]:
```
# using a for loop interate over the results
# find out how many pets recieved treatment that thr type of pet is know

result = cur.execute("select count(*) from pets where treatment_done=1 and␣
 ↪pet_type is not null")
for row in result:
    print(f"The number of pets that had treatment where the type of pet is know:
 ↪ {row[0]}")
```

```
The number of pets that had treatment where the type of pet is know: 16
```

Task 8

[26]:
```
# using a for loop interate over the results
# find how many pets are from east port

result = cur.execute("select count(*) from pets JOIN persons ON pets.owner_id =␣
 ↪persons.id \
                where persons.city='east port'")
for row in res:
    print(f"The number of pets that had treatment where the type of pet is know:
 ↪ {row[0]}")
```

```
The number of pets that had treatment where the type of pet is know: 49
```

Task 9

[28]:
```
# using a for loop interate over the results
# find how many pets are from east port and received treatment

result = cur.execute("select count(*) from pets JOIN persons ON pets.owner_id =␣
 ↪persons.id \
                where persons.city='east port' and pets.treatment_done=1")
for row in result:
```

```
    print(f"The number of pets that had treatment where the type of pet is know:
↪ {row[0]}")
```

```
The number of pets that had treatment where the type of pet is know: 11
```

[ ]: