

# Integrare Python e SQL

Informatica@SEFA 2018/2019 - Laboratorio 9

Massimo Lauria <massimo.lauria@uniroma1.it>  
<http://massimolauria.net/courses/infosefa2018/>

Lunedì, 17 Dicembre 2018

# Integrare Python e SQL

```
# Passo 1. Importare il modulo SQLite
```

```
import sqlite3
```

```
# Passo 2. Aprire la connessione col file SQLite
```

```
filedb = "../dataset/registro_automobilistico_db.sqlite"
```

```
conn=sqlite3.connect(filedb)
```

```
# Passo 3. Eseguire query e vederne i risultati
```

```
for row in conn.execute('select * from Modelli'):
```

```
    print(row)
```

```
# Pass 4. Chiudere la connessione
```

```
conn.close()
```

# Risultato della query

```
('001', 'Panda', '001', 3)
('002', 'Vespa', '004', 4)
('003', 'Brava', '001', 2)
('004', 'Mondeo', '003', 3)
('005', 'V-10', '005', 2)
('006', 'Ducato', '001', 5)
('007', 'Clio', '006', 5)
('008', 'Corolla', '007', 4)
('009', 'Coupè', '001', 1)
('010', 'Golf', '008', 4)
('011', 'Megane', '006', 2)
('012', 'Seicento', '001', 2)
('013', 'Laguna', '006', 2)
('014', 'Civic', '009', 3)
```

# Variazione con il costrutto `with`

Così come per i file, anche per aprire i DB sqlite si può usare il costrutto `with`. Una volta usciti dal blocco `with` la connessione viene chiusa automaticamente.

```
import sqlite3
filedb = "../dataset/registro_automobilistico_db.sqlite"
with sqlite3.connect(filedb) as conn:
    for row in conn.execute('select * from Modelli'):
        print(row)
```

1  
2  
3  
4  
5  
6  
7  
8

# Esercizio 14

Scrivere una funzione

```
simple_query(tabelle,colonne)
```

esegua la query SQL

```
select <colonne> from <tabelle>
```

e restituisca la lista di righe in output, espresse come una lista di tuple.

# Esercizio 14 (esempio)

```
from lab09 import simple_query 1
from pprint import pprint      # pretty-print 2
pprint(simple_query("Combustibili", 3
                    "Descrizione_Combustibile")) 4
pprint(simple_query("Modelli","*")) 5
```

```
[('Benzina',), ('Gasolio',), ('GPL',), ('Metano',)]
[('001', 'Panda', '001', 3),
 ('002', 'Vespa', '004', 4),
 ('003', 'Brava', '001', 2),
 ('004', 'Mondeo', '003', 3),
 ('005', 'V-10', '005', 2),
 ('006', 'Ducato', '001', 5),
 ('007', 'Clio', '006', 5),
 ('008', 'Corolla', '007', 4),
 ('009', 'Coupè', '001', 1),
 ('010', 'Golf', '008', 4),
 ('011', 'Megane', '006', 2),
 ('012', 'Seicento', '001', 2),
 ('013', 'Laguna', '006', 2),
 ('014', 'Civic', '009', 3)]
```

# Plot di punti

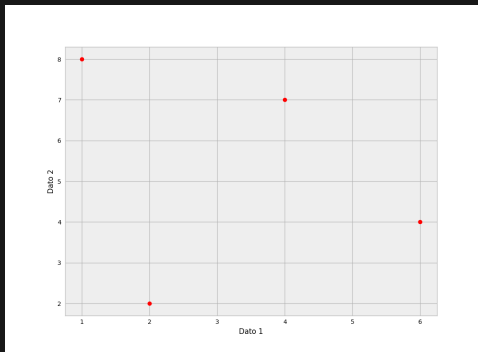
Di default i dati passati a `plot` sono uniti da linee così che il tutto approssimi una curva. Per alcuni tipi di dato ha più senso mostrare i dati senza interpolazione.

```
from matplotlib.pyplot import plot,xlabel,ylabel,savefig      1
                                                                2
x=[1,2,6,4]                                                    3
y=[8,2,4,7]                                                    4
xlabel('Dato 1')                                                5
ylabel('Dato 2')                                                6
plot(x,y,'ro')          # 'ro' determina il tipo di plot      7
savefig('assets/lab09_esempio.png')                            8
```

# Plot

```
x=[1,2,6,4] y=[8,2,4,7]
```

Il plot contiene quattro punti di coordinate  
(1,8), (2,2), (6,4), (4,7)





# Esercizio 15

Scrivere una funzione

```
plot_query(filename)
```

che estragga dalla tabella `Veicoli` i dati di `Cilindrata` e `Velocita` di ogni veicolo.

- costruisca un plot come quello dell'esempio
- ogni punto corrisponde ad un veicolo
- la coordinata  $x$  è la sua cilindrata
- la coordinata  $y$  è la sua velocità
- salva il plot come un'immagine nel file `filename`

Si escludano i veicoli per cui almeno uno di questi due valori sia `NULL`.

# Esercitazione

1. scrivere **un** programma python contenente
  - le funzioni che risolvono gli esercizi
  - nient'altro
  - il file deve chiamarsi `lab09.py`
2. scrivete le vostre funzioni nel file `lab09.py`
3. scaricate il file test `test_lab09.py`
4. eseguite, nella cartella che contiene entrambi,

```
$ python3 test_lab09.py
```

5. migliorate fino ad ottenere una cosa **COME**

```
.....  
-----  
Ran 4 tests in 0.005s  
  
OK
```