In [5]:
```
#import statements
import pymysql
import pandas as pd
import plotly.plotly as py
import plotly.graph_objs as go
from plotly.graph_objs import *
```

DATABASE DESIGN & IMPLEMENTATION (MSCA 31005)

File: PythonMySQL Database Assignment </h3>

Desc: Python to MySQL connection

Authors: Manivassakam Mouttayen

Date: 11/05/2016

References: https://plot.ly/python/graph-data-from-mysql-database-in-python/ (https://plot.ly/python/graph-data-from-mysql-database-in-python/)

Installation:

- https://www.continuum.io/downloads (https://www.continuum.io/downloads)
- pip install pymysql
- pip install plotly

PyMySQL is an interface for connecting to a MySQL database server from Python. It implements the Python Database API v2.0 and contains a pure-Python MySQL client library.

In [38]:
```
#!/usr/bin/python3

# Open database connection
db = pymysql.connect("localhost","root","mvm123","sakila" )

# prepare a cursor object using cursor() method
cursor = db.cursor()

# execute SQL query using execute() method.
cursor.execute("SELECT VERSION()")

# Fetch a single row using fetchone() method.
data = cursor.fetchone()

print ("Database version : %s " % data)

# disconnect from server
db.close()
```

```
Database version : 5.7.15-log
```

In [39]:

```
#!/usr/bin/python3

# Open database connection
db = pymysql.connect("localhost","root","mvm123","sakila" )

# prepare a cursor object using cursor() method
cursor = db.cursor()

# Prepare SQL query to INSERT a record into the database.
# list cdetails from the customers table
#
sql = "SELECT first_name,last_name,COUNT(film_actor.actor_id)from actor \
       INNER JOIN film_actor ON actor.actor_id = film_actor.actor_id \
       GROUP BY actor.first_name, actor.last_name, actor.actor_id \
       HAVING COUNT(film_actor.actor_id) >= '25';"
try:
   # Execute the SQL command
   cursor.execute(sql)

   # Fetch all the rows in a list of lists.
   rows = cursor.fetchall()
except:
    print ("Error: unable to fetch data")
```

In [25]:

```
df = pd.DataFrame( [[ij for ij in i] for i in rows] )
#df.rename(columns={0: 'customerNumber', 1: 'businessName', 2: 'contactFirstName', 3: 'contactLastName', 4:'postCode'}, inplace=True);
#df = df.sort_values(['businessName'], ascending=[True]);
```

In [26]:

```
df.head(20)
```

Out[26]:

	0	1	2
0	AL	GARLAND	26
1	ALAN	DREYFUSS	27
2	ALBERT	JOHANSSON	33
3	ALBERT	NOLTE	31
4	ALEC	WAYNE	29
5	ANGELA	HUDSON	34
6	ANGELA	WITHERSPOON	35
7	ANGELINA	ASTAIRE	31
8	ANNE	CRONYN	27
9	AUDREY	BAILEY	27
10	AUDREY	OLIVIER	25
11	BELA	WALKEN	30
12	BEN	WILLIS	33
13	BOB	FAWCETT	25
14	BURT	DUKAKIS	29
15	CAMERON	ZELLWEGER	33
16	CARMEN	HUNT	26
17	CATE	HARRIS	28
18	CATE	MCQUEEN	30
19	CHRIS	BRIDGES	27

In [52]:

```
# prepare a cursor object using cursor() method
cursor = db.cursor()

sql = "select CONCAT(customer.last_name, ', ', customer.first_name) AS CUSTOME
RS,customer.customer_id AS RENTAL_CUSTOMER_ID \
 from customer where customer_id \
    in (select customer_id from rental)\
 order by RENTAL_CUSTOMER_ID LIMIT 5;"

sql2 = "SELECT LEFT(address,LOCATE(' ',address) - 1) as StreetNumber from addr
ess LIMIT 5;"

sql3 = "select title as Title from film where length(title)=10 order by title
 LIMIT 5;"

sql4 = "select CONCAT(actor.first_name, ' ', actor.last_name) AS ActorNames fr
om actor where last_name like 'A%' or \
last_name like 'B%' or last_name like 'C%' order by last_name LIMIT 5;"


   # Execute the SQL command
cursor.execute(sql)
print ("CUSTOMER AND RENTAL ID")
for(RENTAL_CUSTOMER_ID) in cursor:
    print(RENTAL_CUSTOMER_ID)
print ("")

cursor.execute(sql2)
print ("Street Numbers extracted from street address")
for (StreetNumber) in cursor:
        print(StreetNumber)
print ("")

cursor.execute(sql3)
print ("Find film titles that contains exactly 10 characters")
for (Title) in cursor:
        print(Title)

print ("")

cursor.execute(sql4)

print ("Actors whose last name starts with character A, B or C.")
for (ActorNames) in cursor:
        print(ActorNames)

print ("")
```

```
CUSTOMER AND RENTAL ID
('SMITH, MARY', 1)
('JOHNSON, PATRICIA', 2)
('WILLIAMS, LINDA', 3)
('JONES, BARBARA', 4)
('BROWN, ELIZABETH', 5)

Street Numbers extracted from street address
('47',)
('28',)
('23',)
('1411',)
('1913',)

Find film titles that contains exactly 10 characters
('ALONE TRIP',)
('BASIC EASY',)
('BUGSY SONG',)
('CAUSE DATE',)
('CHILL LUCK',)

Actors whose last name starts with character A, B or C.
('CHRISTIAN AKROYD',)
('KIRSTEN AKROYD',)
('DEBBIE AKROYD',)
('CUBA ALLEN',)
('KIM ALLEN',)
```