

# Un exemple simple d'utilisation de Python pour accéder au SGBD MySQL

## Télécharger puis installer le connecteur Python édité par MySQL

(Choisir le 'Platform Indépendant' en cas de problème)

### Connector/Python 2.0.2

Select Platform:

Microsoft Windows

Looking for previous GA versions?

Windows (Architecture Independent),  
MSI Installer  
Python 2.7

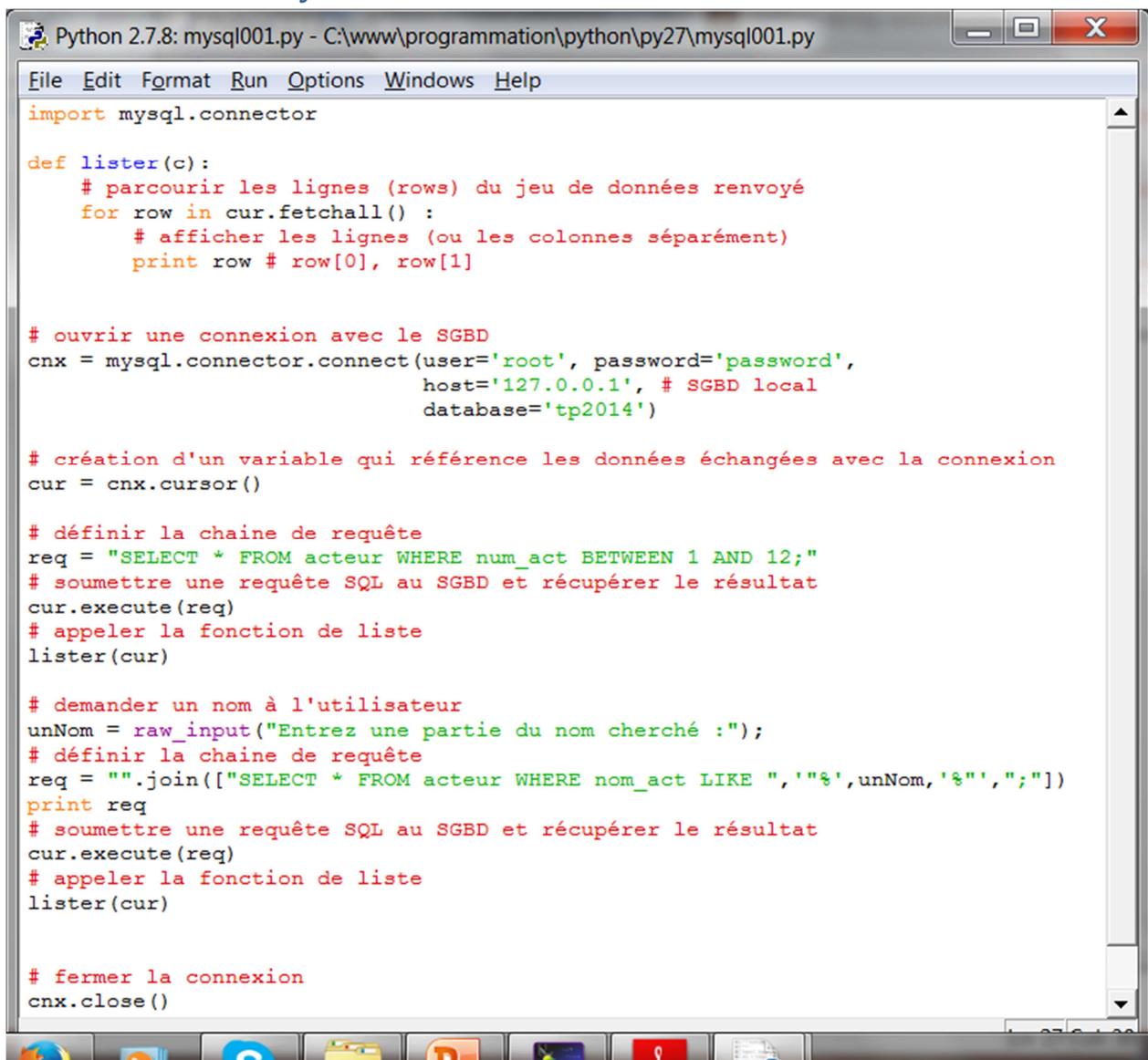
2.0.2 137.5K

Download

(mysql-connector-python-2.0.2-py2.7.msi)

MD5: a828923068cf56ac23ec0d0de55e0002 | [Signature](#)

## Créer un code source Python



```
Python 2.7.8: mysql001.py - C:\www\programmation\python\py27\mysql001.py
File Edit Format Run Options Windows Help
import mysql.connector

def lister(c):
    # parcourir les lignes (rows) du jeu de données renvoyé
    for row in cur.fetchall():
        # afficher les lignes (ou les colonnes séparément)
        print row # row[0], row[1]

# ouvrir une connexion avec le SGBD
cnx = mysql.connector.connect(user='root', password='password',
                              host='127.0.0.1', # SGBD local
                              database='tp2014')

# création d'un variable qui référence les données échangées avec la connexion
cur = cnx.cursor()

# définir la chaîne de requête
req = "SELECT * FROM acteur WHERE num_act BETWEEN 1 AND 12;"
# soumettre une requête SQL au SGBD et récupérer le résultat
cur.execute(req)
# appeler la fonction de liste
lister(cur)

# demander un nom à l'utilisateur
unNom = raw_input("Entrez une partie du nom recherché :");
# définir la chaîne de requête
req = "".join(["SELECT * FROM acteur WHERE nom_act LIKE ", "'%'" + unNom + "%'", ";"])
print req
# soumettre une requête SQL au SGBD et récupérer le résultat
cur.execute(req)
# appeler la fonction de liste
lister(cur)

# fermer la connexion
cnx.close()
```