

Módulo 3: UI y DDBB

Desarrollo en lenguaje Python

Año de realización: 2021

PROFESORA

Carmen Bartolomé Valentín-Gamazo



Unión Europea
Fondo Social Europeo
El FSE invierte en tu futuro

Tema 3-3

Otros DBMS



Unión Europea
Fondo Social Europeo
El FSE invierte en tu futuro

EOI Escuela de
organización
industrial

Módulo 2: el core de Python

Índice

1. SQL resumen
2. MySQL
3. PostgreSQL
4. SQLAlchemy
5. Datasets y dataframes





SQL cheatsheet

SQL CHEAT SHEET <http://www.sqltutorial.org>

MANAGING TABLES

CREATE TABLE t (
 id INT PRIMARY KEY,
 name VARCHAR NOT NULL,
 price INT DEFAULT 0
);
Create a new table with three columns

DROP TABLE t ;
Delete the table from the database

ALTER TABLE t ADD column;
Add a new column to the table

ALTER TABLE t DROP COLUMN c ;
Drop column c from the table

ALTER TABLE t ADD constraint;
Add a constraint

ALTER TABLE t DROP constraint;
Drop a constraint

ALTER TABLE t1 RENAME TO t2;
Rename a table from t1 to t2

ALTER TABLE t1 RENAME c1 TO c2 ;
Rename column c1 to c2

TRUNCATE TABLE t;
Remove all data in a table

USING SQL CONSTRAINTS

CREATE TABLE t(
 c1 INT, c2 INT, c3 VARCHAR,
 PRIMARY KEY (c1,c2)
);
Set c1 and c2 as a primary key

CREATE TABLE t1(
 c1 INT PRIMARY KEY,
 c2 INT,
 FOREIGN KEY (c2) REFERENCES t2(c2)
);
Set c2 column as a foreign key

CREATE TABLE t(
 c1 INT, c1 INT,
 UNIQUE(c2,c3)
);
Make the values in c1 and c2 unique

CREATE TABLE t(
 c1 INT, c2 INT,
 CHECK(c1 > 0 AND c1 >= c2)
);
Ensure c1 > 0 and values in c1 >= c2

CREATE TABLE t(
 c1 INT PRIMARY KEY,
 c2 VARCHAR NOT NULL
);
Set values in c2 column not NULL

MODIFYING DATA

INSERT INTO t(column_list)
VALUES(value_list);
Insert one row into a table

INSERT INTO t(column_list)
VALUES (value_list), ...;
Insert multiple rows into a table

INSERT INTO t1(column_list)
SELECT column_list
FROM t2;
Insert rows from t2 into t1

UPDATE t
SET c1 = new_value;
Update new value in the column c1 for all rows

UPDATE t
SET c1 = new_value,
 c2 = new_value
WHERE condition;
Update values in the column c1, c2 that match the condition

DELETE FROM t;
Delete all data in a table

DELETE FROM t
WHERE condition;
Delete subset of rows in a table



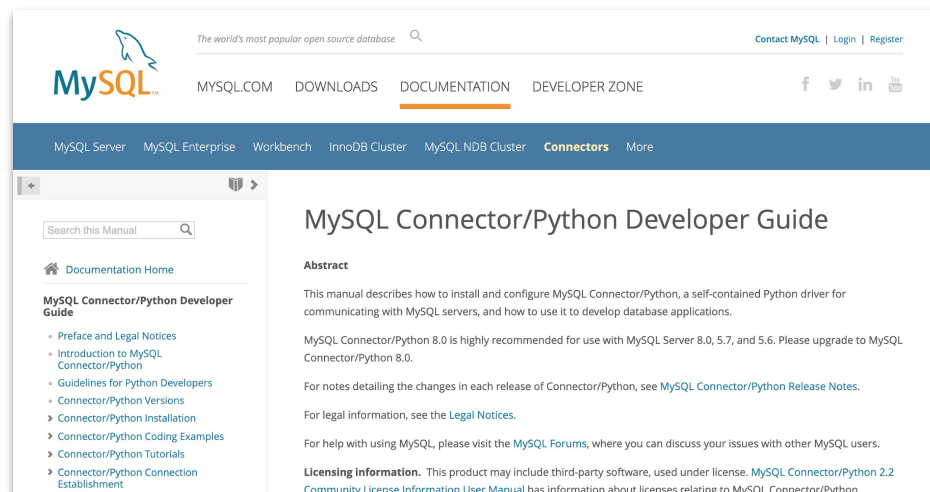
Unión Europea
Fondo Social Europeo
El FSE invierte en tu futuro

EOI Escuela de
organización
industrial

Módulo 2: el core de Python

MySQL

MySQL es una base de datos relacional Open Source muy extendida. Funciona sobre servidor, así que es necesario instalar el driver correspondiente para Python. Se puede descargar de la web de MySQL, pero lo mejor es instalarlo vía Pip.



```
$ pip install mysql-connector-python
```





Módulo 2: el core de Python

```
import mysql.connector
from mysql.connector import Error

def create_connection(host_name, user_name, user_password):
    connection = None
    try:
        connection = mysql.connector.connect(
            host=host_name,
            user=user_name,
            passwd=user_password
        )
        print("Conexión a la base de datos ejecutada")
    except Error as e:
        print(f"Ha ocurrido el error '{e}'")

    return connection

connection = create_connection("localhost", "root", "")

def create_database(connection, query):
    cursor = connection.cursor()
    try:
        cursor.execute(query)
        print("Base de datos creada")
    except Error as e:
        print(f"Ha ocurrido el error '{e}'")

create_database_query = "CREATE DATABASE sm_app"
create_database(connection, create_database_query)
```

Primero necesitamos conectar al servidor MySQL. Por otro lado, se crea la base de datos.

Una vez creada la base de datos, tendremos que pasar la conexión del server a la base de datos concreta, y ya crear el objeto cursor y programar las queries.

Mirar fichero *MySQL_sm_app* para ver el resto



Unión Europea
Fondo Social Europeo
El FSE invierte en tu futuro

EOI Escuela de
organización
industrial

Módulo 2: el core de Python

Para aprender más...

DataCamp

COMMUNITY

[News](#) BETA

[Tutorials](#)

[Cheat Sheets](#)

[Open Courses](#)

[Podcast - DataFramed](#)

[Chat](#) NEW

DATA CAMP

[Official Blog](#)

[Resource Center](#)

[Upcoming Events](#)

[Subscribe to RSS](#)

[f](#) [t](#) [in](#) [v](#)

[About](#) [Terms](#) [Privacy](#)

[Log in](#) [Create Free Account](#)

[Back to Tutorials](#)

Tutorials

Hafeezul Kareem Shaik
December 9th, 2019

SQL +2

Getting Started with MySQL in Python

In this tutorial, you'll learn how to install MySQL, create databases, and create tables, along with much more.

15

28

Prerequisites

- Python Data Types, Control Structures, Loops, etc., If you're new to Python learn Datacamp's free [Intro to Python for Data Science](#) course.
- SQL Basics. If you don't know SQL, take the Datacamp's free [SQL](#) course.



Unión Europea
Fondo Social Europeo
El FSE invierte en tu futuro

EOI Escuela de
organización
industrial

Módulo 2: el core de Python

PostgreSQL

Es una base de datos relacional, también Open Source, pero más avanzada que MySQL. También requiere conectar a través de un servidor donde se crean y almacenan las bases de datos. El driver más popular es psycopg2 pero requiere tener instaladas las librerías cliente de PostgreSQL.

psycopg2 2.8.5

```
pip install psycopg2
```





Módulo 2: el core de Python

```
import psycopg2
from psycopg2 import OperationalError

# Define la función que conecta con el servidor PostgreSQL
def create_connection(db_name, db_user, db_password, db_host, db_port):
    connection = None
    try:
        connection = psycopg2.connect(
            database=db_name,
            user=db_user,
            password=db_password,
            host=db_host,
            port=db_port,
        )
        print("La conexión se ha ejecutado")
    except OperationalError as e:
        print(f"Ha ocurrido el error '{e}'")
    return connection

# Conecta con el servidor llamando a la función create_connection
connection = create_connection(
    "postgres", "postgres", "abc123", "127.0.0.1", "5432"
)

# Define la función para crear la base de datos en el servidor
def create_database(connection, query):
    connection.autocommit = True
    cursor = connection.cursor()
    try:
        cursor.execute(query)
        print("Consulta ejecutada con éxito")
    except OperationalError as e:
        print(f"Ha ocurrido el error '{e}'")

# Crea la base de datos
create_database_query = "CREATE DATABASE sm_app"
create_database(connection, create_database_query)
```

Primero necesitamos conectar al servidor PostgreSQL. Por otro lado, se crea la base de datos.

Una vez creada la base de datos, tendremos que pasar la conexión del server a la base de datos concreta, y ya crear el objeto cursor y programar las queries.

Mirar fichero *PostgreSQL_sm_app* para ver el resto



Unión Europea
Fondo Social Europeo
El FSE invierte en tu futuro

EOI Escuela de
organización
industrial

Módulo 2: el core de Python

SQLAlchemy

Es la librería de SQL para Python más utilizada. Permite trabajar con todas las bases de datos vistas: SQLite, MySQL, PostgreSQL,....

Para instalarla: **\$ pip install sqlalchemy**

SQLAlchemy se utiliza a uno de estos 3 niveles:

- Engine room
- SQL expression language
- ORM (Object Relational Model)





Unión Europea
Fondo Social Europeo
El FSE invierte en tu futuro

EOI Escuela de
organización
industrial

Módulo 2: el core de Python

SQLAlchemy

SQLAlchemy trabaja con los drivers que hemos visto, pero no es necesario importarlos. Al inicio, se le pasa esta cadena:

```
dialect + driver :// user : password @ host : port / dbname
```

Lo más común es trabajar con ORM, pues se evita la complejidad de SQL, pero si algo no sale como se espera, es necesario bajar de nivel para volver a SQL.

[Tutorial SQLAlchemy ORM](#)





Unión Europea
Fondo Social Europeo
El FSE invierte en tu futuro

EOI Escuela de
organización
industrial

Módulo 2: el core de Python

Datasets y Dataframes

Para aplicaciones de analítica de datos y machine learning, lo normal es generar un dataset, que es la estructura más simple orientada a crear un dataframe o conjunto ordenado de datos que alimentarán un algoritmo de machine learning. Se tratan los datos en formato estadístico, y no se necesita realizar las consultas como en los casos anteriores.

Ejemplo BigML





Unión Europea
Fondo Social Europeo
El FSE invierte en tu futuro

EOI Escuela de
organización
industrial

Módulo 2: el core de Python

carmenbvg@gmail.com

```
1  def gratitude():  
2      print("Thank you.")  
3
```

