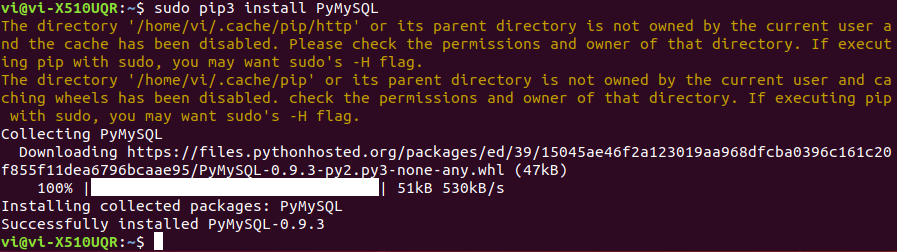
**Python으로 MariaDB연결 및 DML작업**

1. **PyMySQL 다운받기**

먼저 Python으로 MariaDB연결하는 과정에서 가장 먼저 Mariadb 서버와 통신을 할 수 있게 해주는 라이브러리 인 PyMySQL을 다운 받아야 한다.

|  |
| --- |
| :~$ sudo pip3 install PyMySQL |



Python 에서 사용하기 위해서는 import를 사용한다.

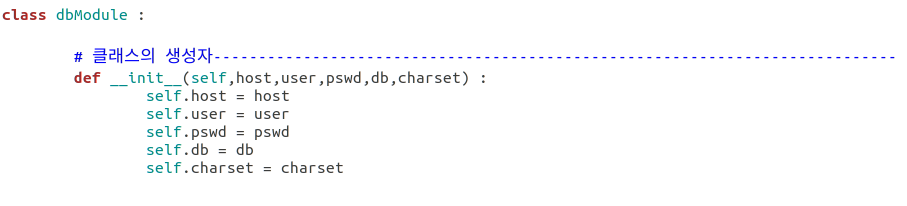


1. **사용함수 설명**

우리는 DB연결부터 ~ 종료 및 DML작업을 하나의 클래스와 7개의 함수를 만들었다. 각 함수의

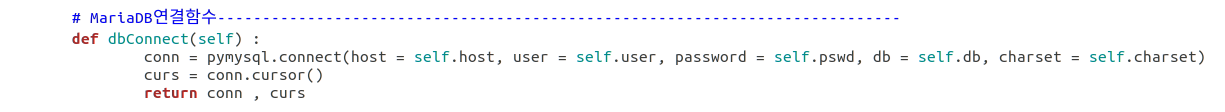
이름과 기능은 다음과 같다.

1. **\_\_init\_\_()**



DB와 연결할 때 host, user, password, db, charset을 파라미터로 지정하는데 매개변수를 인자로 받아서 객체를 생성할 수 있도록 하기 위해 사용

1. **dbConnect()**



Pymysql.connect() 메서드를 사용하여 MariaDB에 연결 여기서 사용하는 주요 파라미터

Host : 접속할 server 주소

User : ID

Password : ID의 암호

Db : 접속할 데이터베이스

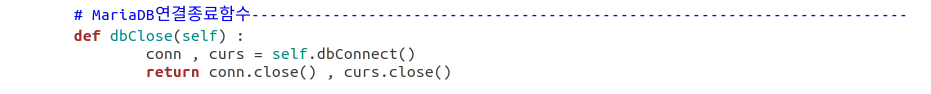
Charset : select하여 데이터를 가져올 때 한글이 깨질 수 있으므로 설정 해야한다.

DB와 연결이 성공하면 cursor() 메서드를 호출하여 cursor 객체를 가져온다.

Cursor 객체의 executeI() 메서드를 사용하여 SQL 문장을 DB서버에 전송하기 위해서다.

\*\*\* 해당 함수를 따로 만든 이유는 트랜잭션 시작할 때 마다 DB연결을 하기 위해서

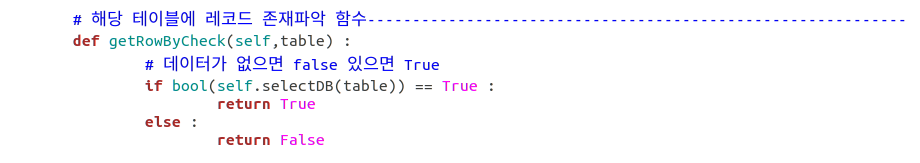
1. **dbClose()**



DB연결함수인 dbConnect()를 호출하고 close() 메서드를 사용하여 DB연결종료와 Cursor종료를 한다.

\*\*\* 해당 함수를 따로 만든 이유는 트랜잭션 끝날 때 마다 DB연결종료를 하기 위해서

1. **getRowByCheck()**

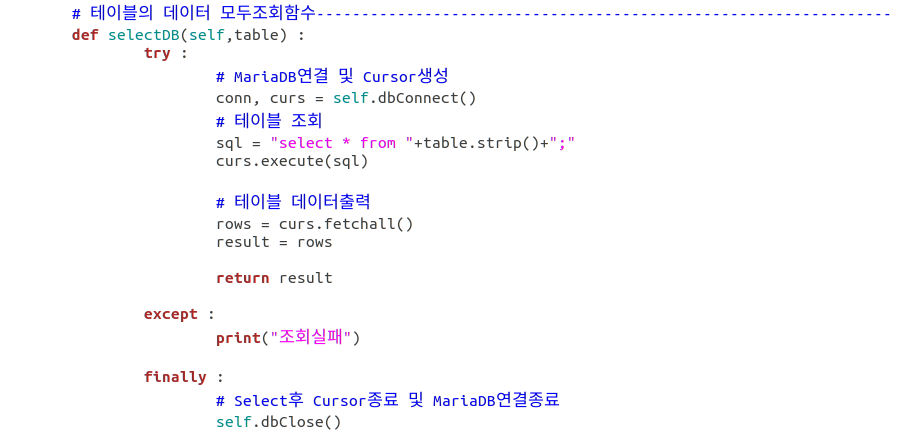


Data를 삽입하기 전에 해당 테이블의 Row존재 여부를 파악하기 위해 bool대수를 사용한다.

조회한 테이블의 Data가 없으면 False로 Data가 있으면 True로 return 한다.

\*\*\* 해당 함수는 트윗을 가져와 Data를 삽입하기 전에 기존의 Data가 있으면 삭제하기 위해서

1. **selectDB()**



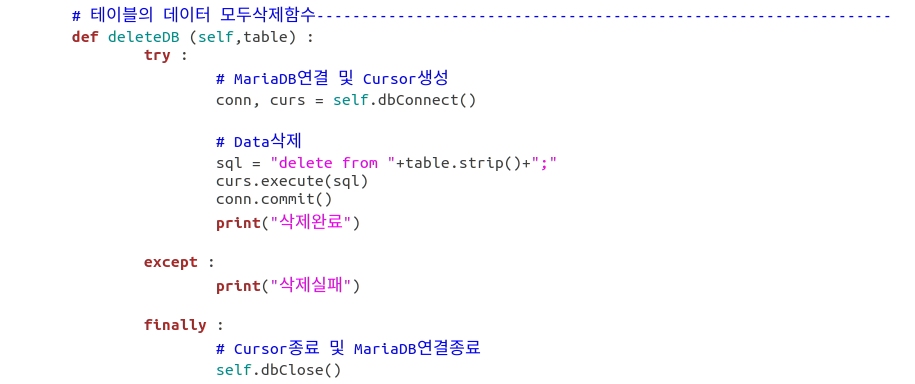
db.Connect() 함수를 호출해 DB연결 Cursor객체를 가져온다. Cursor 객체

의 execute 메서드를 사용하여 Data를 select하고 fetchall() 메서드를 사용해 DB서버로부터

Data를 가져온다. 결과값을 return 한다. 모든작업이 끝나면 dbClose() 함수를 호출해

DB연결종료와 Cursor를 종료한다.

1. **deletedDB()**

****

db.Connect() 함수를 호출해 DB연결 Cursor객체를 가져온다. Cursor 객체

의 execute 메서드를 사용하여 Data를 delete하고 모든작업이 끝나면 dbClose() 함수를 호출해

DB연결종료와 Cursor를 종료한다.

1. **insertDB()**

****

db.Connect() 함수를 호출해 DB연결 Cursor객체를 가져온다. Cursor 객체

의 execute 메서드를 사용하여 Data를 insert하고 모든작업이 끝나면 dbClose() 함수를 호출해

DB연결 종료와 Cursor를 종료한다.

1. **Python으로 작성한 코드**

|  |
| --- |
| '''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''  dbModule.py  Title: MariaDB연결,종료 및 DML 작업  Author: Bae InGyu  Create\_at: 2019.04.02.  '''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''  import pymysql  class dbModule :  # 클래스의 생성자----------------------------------------------------------------------------  def \_\_init\_\_(self,host,user,pswd,db,charset) :  self.host = host  self.user = user  self.pswd = pswd  self.db = db  self.charset = charset  # MariaDB연결함수----------------------------------------------------------------------------  def dbConnect(self) :  conn = pymysql.connect(host = self.host, user = self.user, password = self.pswd, db = self.db, charset = self.charset)  curs = conn.cursor()  return conn , curs  # MariaDB연결종료함수-----------------------------------------------------------------------  def dbClose(self) :  conn , curs = self.dbConnect()  return conn.close() , curs.close()    # 해당 테이블에 레코드 존재파악 함수-----------------------------------------------------  def getRowByCheck(self,table) :  # 데이터가 없으면 false 있으면 True  if bool(self.selectDB(table)) == True :  return True  else :  return False    # 테이블의 데이터 모두조회함수------------------------------------------------------------  def selectDB(self,table) :  try :  # MariaDB연결 및 Cursor생성  conn, curs = self.dbConnect()  # 테이블 조회  sql = "select \* from "+table.strip()+";"  curs.execute(sql)  # 테이블 데이터출력  rows = curs.fetchall()  result = rows  return result  except :  print("조회실패")    finally :  # Select후 Cursor종료 및 MariaDB연결종료  self.dbClose()      # 테이블의 데이터 모두삭제함수------------------------------------------------------------  def deleteDB (self,table) :  try :  # MariaDB연결 및 Cursor생성  conn, curs = self.dbConnect()  # Data삭제  sql = "delete from "+table.strip()+";"  curs.execute(sql)  conn.commit()  print("삭제완료")  except :  print("삭제실패")  finally :  # Cursor종료 및 MariaDB연결종료  self.dbClose()    # 테이블의 데이터 삽입함수-----------------------------------------------------------------  def insertDB (self,table,values) :  try :  # MariaDB연결 및 Cursor생성  conn, curs = self.dbConnect()  # Data삽입  self.values = values  sql = "insert into "+table.strip()+values  curs.execute(sql)  conn.commit()  print("삽입완료")  except :  print("삽입실패")  finally :  # Cursor종료 및 MariaDB연결종료  self.dbClose() |