Lab 09.02 Python and Databases

Using databases

Overview

Calling the sql commands in python

Before the lab

```
pip install mysql-connector
```

Install the python package

You will need to have your mysql server up and running,

I would usually create the database and tables on the server and not through python.

In this lab I show you how to create the table and then perform the crud operations.

NOTE: The user name and password for your database in WAMP the default is root and blank,

I show you how to change it in the lecture.

You should make a new file for each of these tasks.

NOTE: I origianlly had the student table as id, name and address.

I changed it to id name and age to match the flask project we diid last week

Not usually done (create database and tables)

1. Create a database called datarepresentation using a python script

```
import mysql.connector

mydb = mysql.connector.connect(
  host="localhost",
  user="root",
  password=""
)

mycursor = mydb.cursor()

mycursor.execute("CREATE DATABASE datarepresentation")
```

2. Create the table in the database with the command

```
import mysql.connector

mydb = mysql.connector.connect(
  host="localhost",
  user="root",
  password="",
  database="datarepresentation"
)

mycursor = mydb.cursor()
sql="CREATE TABLE student (id INT AUTO_INCREMENT PRIMARY KEY, name VARCHAR(255), age INT)"

mycursor.execute(sql)
```

CRUD operations on a table, this is what you would normally do from an application

3. Insert data

```
import mysql.connector

db = mysql.connector.connect(
  host="localhost",
  user="root",
  password="",
  database="datarepresentation"
)

cursor = db.cursor()
sql="insert into student (name, age) values (%s,%s)"
values = ("Mary",21)

cursor.execute(sql, values)

db.commit()
print("1 record inserted, ID:", cursor.lastrowid)
```

```
import mysql.connector

db = mysql.connector.connect(
  host="localhost",
  user="root",
  password="",
  database="datarepresentation"
)

cursor = db.cursor()
sql="select * from student where id = %s"
values = (1,)

cursor.execute(sql, values)
result = cursor.fetchall()
for x in result:
  print(x)
```

4. View data

5. Update data

```
import mysql.connector

db = mysql.connector.connect(
  host="localhost",
  user="root",
  password="",
  #user="datarep",  # this is the user name on my mac
  #passwd="password" # for my mac
  database="datarepresentation"
)

cursor = db.cursor()
sql="update student set name= %s, age=%s  where id = %s"
values = ("Joe",33, 1)

cursor.execute(sql, values)

db.commit()
print("update done")
```

```
import mysql.connector

db = mysql.connector.connect(
  host="localhost",
  user="root",
  password="",
  #user="datarep",  # this is the user name on my mac
  #passwd="password" # for my mac
  database="datarepresentation"
)

cursor = db.cursor()
sql="delete from student where id = %s"
values = (1,)

cursor.execute(sql, values)

db.commit()
print("delete done")
```

6. Delete

Put it into a file that can be used from another file (eg from your flask

```
import mysql.connector
class StudentDAO:
    db=""
    def __init__(self):
        self.db = mysql.connector.connect(
        host="localhost",
        user="root",
        password="",
        #user="datarep",  # this is the user name on my mac
        #passwd="password" # for my mac
        database="datarepresentation"
        )
    def create(self, values):
        cursor = self.db.cursor()
        sql="insert into student (name, age) values (%s,%s)"
        cursor.execute(sql, values)

        self.db.commit()
        return cursor.lastrowid

    def getAll(self):
        cursor = self.db.cursor()
        sql="select * from student"
        cursor.execute(sql)
        result = cursor.fetchall()
        return result

    def findByID(self, id):
        cursor = self.db.cursor()
        sql="select * from student where id = %s"
        values = (id,)

        cursor.execute(sql, values)
        result = cursor.fetchone()
        return result
    def update(self, values):
        cursor = self.db.cursor()
        sql="update student set name= %s, age=%s  where id = %s"
        cursor.execute(sql, values)
        self.db.commit()
    def delete(self, id):
        cursor = self.db.cursor()
        sql="delete from student where id = %s"
        values = (id,)

        cursor.execute(sql, values)

        self.db.commit()
        print("delete done")

studentDAO = StudentDAO()
```

app)

Test it

```
from zstudentDAO import studentDAO

#create
latestid = studentDAO.create(('mark', 45))
# find by id
result = studentDAO.findByID(latestid);
print (result)

#update
studentDAO.update(('Fred',21,latestid))
result = studentDAO.findByID(latestid);
print (result)

# get all
allStudents = studentDAO.getAll()
for student in allStudents:
  print(student)

# delete
studentDAO.delete(latestid)
```