

# บทที่ 9

## ไพทอนกับระบบการจัดการฐานข้อมูล

Chapter 9 Python and Database Management System



ตอนที่ 1 ระบบการจัดการฐานข้อมูล (DMBS) ด้วย MySQL

Asst.Prof.Dr.Nutthapat Kaewrattanapat  
Suan Sunandha Rajabhat University

Course: Computer Programming and Developing Applications for Education  
Department of Digital Technology for Education  
Faculty of Education, Suan Sunandha Rajabhat University

# อาจารย์บรรยาย



## ผู้ช่วยศาสตราจารย์ ดร.ณัฐภัทร แก้วรัตนภัทร

การศึกษา: มหาวิทยาลัยเทคโนโลยีพระจอมเกล้าพระนครเหนือ

- 2565 ปริญญาเอก ปรัชญาดุสิตบัณฑิต สาขาวิชาเทคโนโลยีสารสนเทศและการสื่อสารเพื่อการศึกษา (GPA. 4.00)
- 2551 ปริญญาโท วิทยาศาสตรมหาบัณฑิต สาขาวิชาระบบสารสนเทศเพื่อการจัดการ (GPA. 3.58)
- 2549 ปริญญาตรี วิทยาศาสตรบัณฑิต เกียรตินิยมอันดับ 1 สาขาวิชาวิทยาการคอมพิวเตอร์ (GPA. 3.66)

## คุณวุฒิวิชาชีพและประกาศนียบัตร

- คุณวุฒิวิชาชีพ จาก สถาบันคุณวุฒิวิชาชีพ (องค์การมหาชน) สาขาวิชาชีพอุตสาหกรรมดิจิทัล สาขาธุรกิจดิจิทัลและพาณิชย์อิเล็กทรอนิกส์ อาชีพนักจัดการเทคโนโลยีสารสนเทศสำหรับธุรกิจ ระดับ 6 เลขที่หนังสือรับรอง PQCN-ICT-ECM-0-251100-B-64/000029
- วิทยาศาสตรข้อมูลด้วยภาษาไพทอน, มหาวิทยาลัยธรรมศาสตร์
- การโปรแกรมสำหรับนักภาษาศาสตร์, มหาวิทยาลัยมหิดล
- การโปรแกรมภาษาไพทอน, มหาวิทยาลัยเพนซิลวาเนีย, สหรัฐอเมริกา
- การโปรแกรมสำหรับทุกคน, มหาวิทยาลัยมิชิแกน, สหรัฐอเมริกา

ติดต่อ: nutthapat.ke@ssru.ac.th

# Course Description

หลักการ ทฤษฎี ที่เกี่ยวข้องกับการเขียนโปรแกรมและพัฒนาแอปพลิเคชัน หลักการ พัฒนาโปรแกรมคอมพิวเตอร์ คุณสมบัติของโปรแกรมภาษาชนิดต่าง ๆ หลักการ เบื้องต้นเกี่ยวกับองค์ประกอบ ลักษณะคำสั่ง การเขียนโปรแกรม ขั้นตอนวิธี การ วิเคราะห์ การออกแบบ แอปพลิเคชันเพื่อการศึกษา การประเมินซอฟต์แวร์ สามารถพัฒนาแอปพลิเคชันเพื่อการศึกษา

Principles, theories associated with computer programming and development applications, computer programming principles, computer language, elements of computer language, syntax, computer programming, algorithms, analysis and design application for education, software evaluation, candidate teachers able to developing applications for education.

Algorithm Design

Python Programming Language

System Development Life Cycle: SDLC

Reference: <https://edu.ssru.ac.th/useruploads/files/20230724/1772131ed638786bc8d19918b37249af72c36be4.pdf>

# Course Outline

- Chapter 1 – Computer Programming
- Chapter 2 – Introduction to Python Programming
- Chapter 3 – Conditional Statement
- Chapter 4 – Iteration Statement
- Chapter 5 – Strings
- Chapter 6 – Immutable and Mutable Data Types
- Chapter 7 – Functions
- Chapter 8 – Object-Oriented Programming: OOP
- Chapter 9 – Python and Database Management System
- Chapter 10 – System Development Life Cycle: SDLC

# Measurement and Evaluation

## การวัดและประเมินผล

### 1. ระหว่างการจัดการเรียนรู้

- สอบ Pre-test
- การมอบหมายงาน
- สอบ Post-test
- การมีส่วนร่วมในชั้นเรียน

0%  
24%  
12%  
4%

### 2. การสอบกลางภาค (Midterm Examination)

- ปรนัย 35 ข้อ (35 คะแนน) อัตนัย 1 ข้อ (5 คะแนน)

20%

### 3. โครงการประจำภาคเรียน (Term Project)

- โครงการและการนำเสนอ

20%

### 4. การสอบปลายภาค (Final Examination)

- ปรนัย 35 ข้อ (35 คะแนน) อัตนัย 1 ข้อ (5 คะแนน)

20%

| ร้อยละ   | ระดับผลการเรียน | ความหมาย            |
|----------|-----------------|---------------------|
| 86 – 100 | A               | ดีเยี่ยม            |
| 82 – 85  | A-              | ดีเยี่ยม            |
| 78 – 81  | B+              | ดีมาก               |
| 74 – 77  | B               | ดี                  |
| 70 – 73  | B-              | ค่อนข้างดี          |
| 66 – 69  | C+              | ปานกลางค่อนข้างดี   |
| 62 – 65  | C               | ปานกลาง             |
| 58 – 61  | C-              | ปานกลางค่อนข้างอ่อน |
| 54 – 57  | D+              | ค่อนข้างอ่อน        |
| 50 – 53  | D               | อ่อน                |
| 46 – 49  | D-              | อ่อนมาก             |
| 0 – 45   | F               | ตก                  |

# Measurement and Evaluation

| ครั้งที่/วันที่         | บทเรียน/หัวข้อ                                                                               | มอบหมายงาน (24%) | สอบ Post-test (12%) | การมีส่วนร่วมในชั้นเรียน (4%) |
|-------------------------|----------------------------------------------------------------------------------------------|------------------|---------------------|-------------------------------|
| 1 พท 7 ธค 66            | แนะนำรายวิชา การวัดและการประเมินผล หัวข้อเรียนรู้ บทที่ 1 การโปรแกรมคอมพิวเตอร์              | -                | 1 คะแนน             | ขาด/ไม่ทันเช็คชื่อ -1%        |
| 2 พท 14 ธค 66           | บทที่ 2 พื้นฐานการโปรแกรมภาษาไพทอน                                                           | 2 คะแนน          | 1 คะแนน             | ขาด/ไม่ทันเช็คชื่อ -1%        |
| 3 พท 21 ธค 66           | บทที่ 3 การโปรแกรมแบบเงื่อนไข (Conditional Statement)                                        | -                | 1 คะแนน             | ขาด/ไม่ทันเช็คชื่อ -1%        |
| 4 พท 28 ธค 66<br>Online | บทที่ 3 การโปรแกรมแบบเงื่อนไข (Conditional Statement) (ต่อ)                                  | 3 คะแนน          | 1 คะแนน             | ขาด/ไม่ทันเช็คชื่อ -1%        |
| 5 พท 4 มค 67            | บทที่ 4 การโปรแกรมแบบทำซ้ำ (Iteration Statement)                                             | 3 คะแนน          | 1 คะแนน             | ขาด/ไม่ทันเช็คชื่อ -1%        |
| 6 พท 11 มค 67           | บทที่ 5 การโปรแกรมสายอักขระ (String)                                                         | -                | 1 คะแนน             | ขาด/ไม่ทันเช็คชื่อ -1%        |
| 7 จ 22 มค 67            | สอบกลางภาค ปรนัย 35 ข้อ (35 คะแนน) และอัตนัย 1 ข้อ (5 คะแนน) 20%                             |                  |                     |                               |
| 8 พท 1 กพ 67            | บทที่ 6 ประเภทข้อมูลแบบลำดับ (Immutable and Mutable Data Types) : Tuple, List และ Dictionary | 6 คะแนน          | 1 คะแนน             | ขาด/ไม่ทันเช็คชื่อ -1%        |

# Measurement and Evaluation

| ครั้งที่/วันที่         | บทเรียน/หัวข้อ                                                                             | มอบหมายงาน<br>(24%) | สอบ Post-test<br>(12%) | การมีส่วนร่วมในชั้นเรียน<br>(4%) |
|-------------------------|--------------------------------------------------------------------------------------------|---------------------|------------------------|----------------------------------|
| 9 พฤษ 8 กพ 67           | บทที่ 7 ฟังก์ชัน                                                                           | 2 คะแนน             | 1 คะแนน                | ขาด/ไม่ทันเช็คชื่อ -1%           |
| 10 พฤษ 15 กพ 67         | บทที่ 8 การโปรแกรมเชิงวัตถุ                                                                | 2 คะแนน             | 1 คะแนน                | ขาด/ไม่ทันเช็คชื่อ -1%           |
| 12 พฤษ 22 กพ 67         | บทที่ 9 ภาษาไพทอนกับระบบการจัดการฐานข้อมูล<br>ระบบการจัดการฐานข้อมูล (DBMS) MySQL          | 2 คะแนน             | 1 คะแนน                | ขาด/ไม่ทันเช็คชื่อ -1%           |
| 13 พฤษ 29 กพ 67         | บทที่ 9 ภาษาไพทอนกับระบบการจัดการฐานข้อมูล ต่อ<br>การเขียนโปรแกรมภาษา Python ร่วมกับ MySQL | 2 คะแนน             | 1 คะแนน                | ขาด/ไม่ทันเช็คชื่อ -1%           |
| 14 จ 4 มีค 67<br>Online | บทที่ 10 วงจรการพัฒนาระบบ                                                                  | 2 คะแนน             | 1 คะแนน                | ขาด/ไม่ทันเช็คชื่อ -1%           |
| 15 พฤษ 14 มีค 67        | สอบปลายภาค ปรนัย 35 ข้อ (35 คะแนน) และอัตนัย 1 ข้อ (5 คะแนน) 20%                           |                     |                        |                                  |
| 16 พฤษ 21 มีค 67        | ส่งโครงงาน และนำเสนอ 20%                                                                   |                     |                        |                                  |

# การติดตั้งตัวเชื่อมต่อกับระบบการจัดการฐานข้อมูล MySQL

ติดตั้งตัวเชื่อมต่อกับระบบการจัดการฐานข้อมูล MySQL

1. เปิด Console (Windows) หรือ Terminal (Mac)
2. ป้อนคำสั่ง

**python -m pip install mysql-connector-python** (Windows)

**python3 -m pip install mysql-connector-python** (Mac)

```
nuth@Nutthapats-MacBook-Pro ~ % python3 -m pip install mysql-connector-python
Collecting mysql-connector-python
  Downloading mysql_connector_python-8.3.0-cp312-cp312-macosx_13_0_arm64.whl.metadata (1.9 kB)
  Downloading mysql_connector_python-8.3.0-cp312-cp312-macosx_13_0_arm64.whl (15.3 MB)
  _____ 15.3/15.3 MB 8.9 MB/s eta 0:00:00
Installing collected packages: mysql-connector-python
Successfully installed mysql-connector-python-8.3.0
```

# การติดตั้งตัวเชื่อมต่อกับระบบการจัดการฐานข้อมูล MySQL

ตรวจสอบว่าในเครื่องคอมพิวเตอร์มีตัวเชื่อมต่อกับระบบการจัดการฐานข้อมูล MySQL

1. เปิด Console (Windows) หรือ Terminal (Mac)
2. ป้อนคำสั่ง `pip freeze` (Windows) หรือ `pip3 freeze` (Mac) และกด Enter
3. ตรวจสอบว่ามีคำว่า  
**`mysql-connector-python==8.3.0`**

```
[nuth@Nutthapats-MacBook-Pro ~ % pip3 freeze
appnope==0.1.4
asttokens==2.4.1
comm==0.2.1
debugpy==1.8.1
decorator==5.1.1
executing==2.0.1
ipykernel==6.29.2
ipython==8.21.0
jedi==0.19.1
jupyter_client==8.6.0
jupyter_core==5.7.1
matplotlib-inline==0.1.6
mysql-connector-python==8.3.0
nest-asyncio==1.6.0
packaging==23.2
parso==0.8.3
```

# การติดตั้งตัวแสดงผลแบบตาราง (prettytable)

ติดตั้งตัวแสดงผลแบบตารางด้วย prettytable

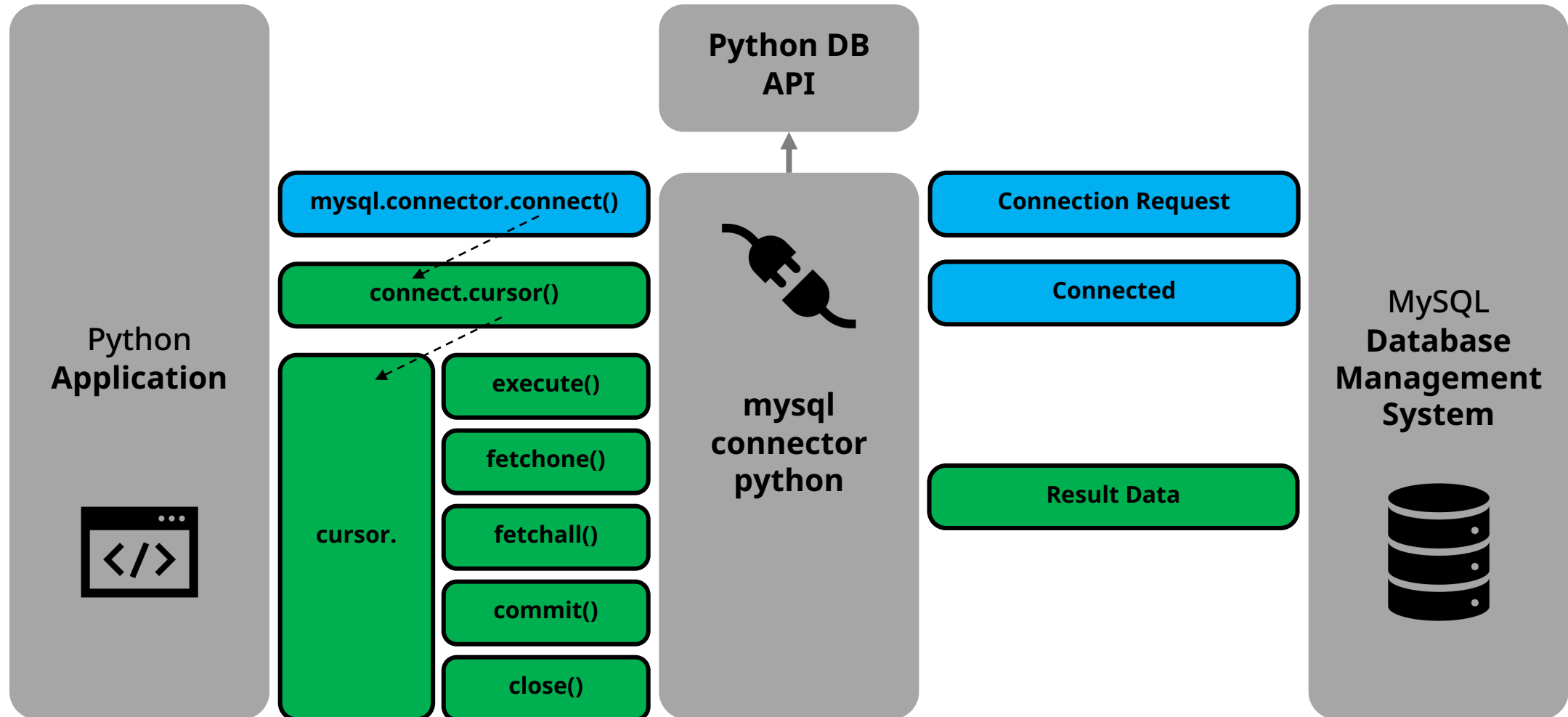
1. เปิด Console (Windows) หรือ Terminal (Mac)
2. ป้อนคำสั่ง

**python -m pip install prettytable** (Windows)

**python3 -m pip install prettytable** (Mac)

```
lnuth@Nutthapats-MacBook-Pro ~ % python3 -m pip install prettytable
Collecting prettytable
  Downloading prettytable-3.10.0-py3-none-any.whl.metadata (30 kB)
Requirement already satisfied: wcwidth in ./Library/Python/3.12/lib/python/site-packages (from prettytable) (0.2.13)
Downloading prettytable-3.10.0-py3-none-any.whl (28 kB)
Installing collected packages: prettytable
Successfully installed prettytable-3.10.0
```

# ภาพรวมของการเชื่อมต่อและจัดการระบบฐานข้อมูลด้วยภาษา Python



# สรุปขั้นตอนการเขียนโปรแกรมภาษา Python เพื่อจัดการระบบฐานข้อมูล

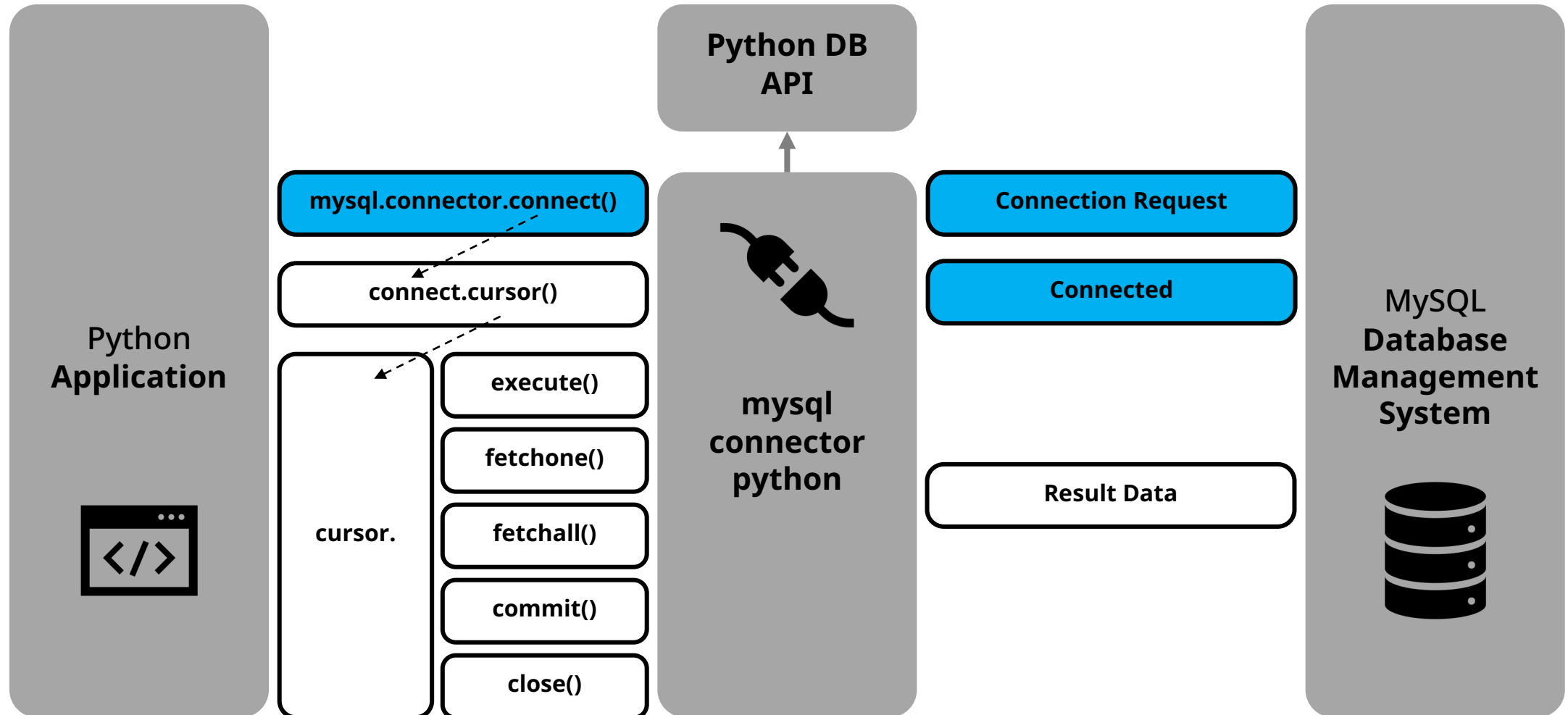
## ส่วนของระบบการจัดการฐานข้อมูล (Database)

- สร้างฐานข้อมูล (Database) ด้วย MySQL หรือ MariaDB ตามที่ได้ออกแบบไว้
- สร้างตารางข้อมูล (Table) ภายในฐานข้อมูล
- อาจเพิ่มข้อมูลสำคัญไว้ (Insert)

## ส่วนของการโปรแกรมเพื่อจัดการระบบฐานข้อมูล (Python)

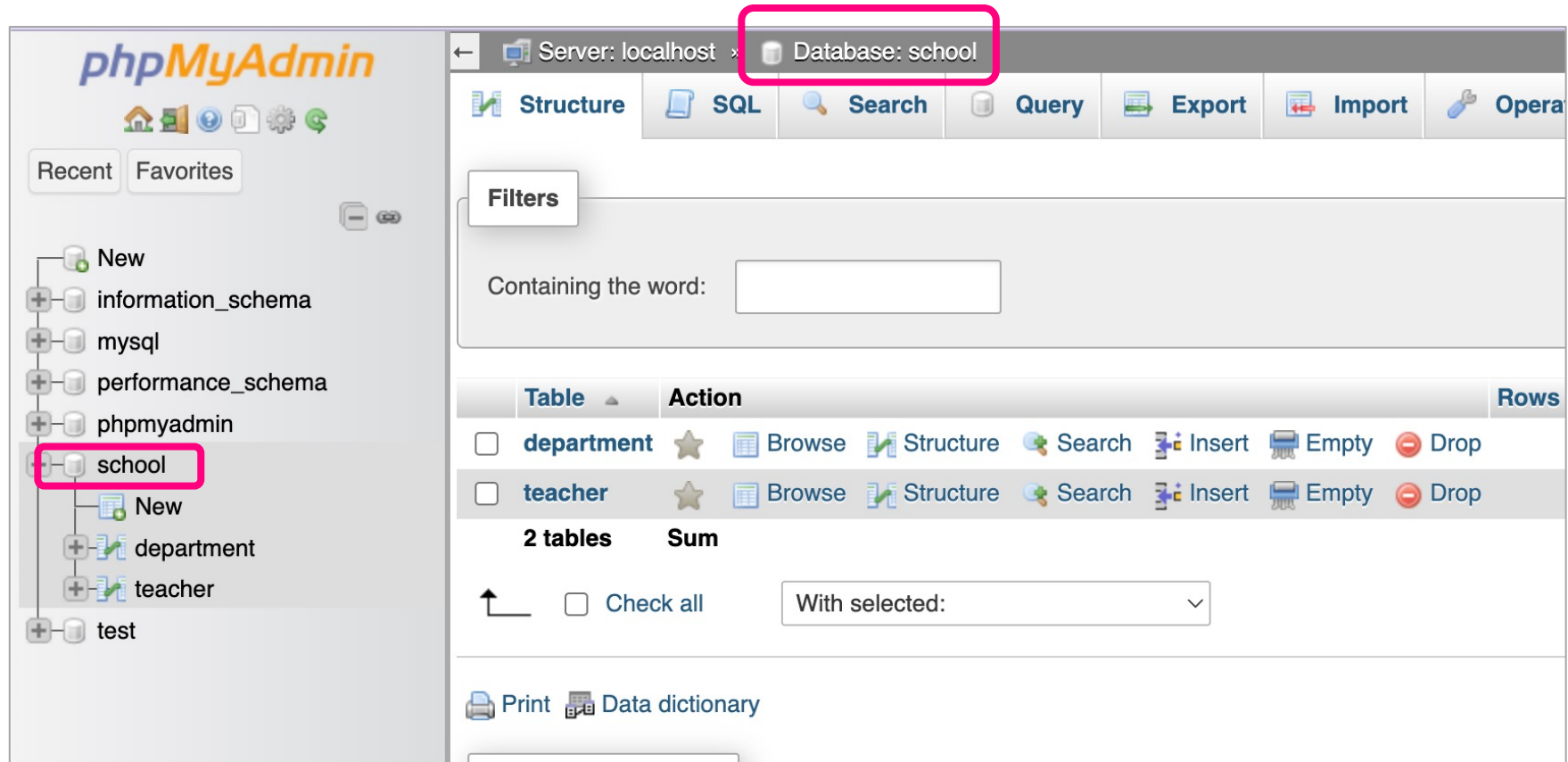
- ตรวจสอบว่าติดตั้งตัวเชื่อมต่อบนระบบฐานข้อมูลให้พร้อมใช้หรือยัง?
- นำเข้า `import mysql.connector` และเชื่อมต่อฐานข้อมูลผ่าน Code ภาษา Python
- เขียนโปรแกรมเพื่อประมวลผลคำสั่ง SQL ผ่าน Code ภาษา Python
- รับค่าข้อมูลจากฐานข้อมูลและแสดงผลให้ผู้ใช้งาน

# 1. การเขียนโปรแกรมเชื่อมต่อระบบการจัดการฐานข้อมูล



# 1. การเขียนโปรแกรมเชื่อมต่อระบบการจัดการฐานข้อมูล

ขั้นแรกได้สร้างฐานข้อมูล ชื่อว่า school ไว้



# 1. การเขียนโปรแกรมเชื่อมต่อระบบการจัดการฐานข้อมูล

ขั้นแรกได้สร้างฐานข้อมูล ชื่อว่า school ไว้ จากนั้นจึงเขียนโปรแกรมเพื่อเชื่อมต่อ ดังนี้

```
#การเชื่อมต่อระบบการจัดการฐานข้อมูล (MySQL) ชื่อว่า school
import mysql.connector
mydb = mysql.connector.connect(
    host="localhost",
    user="root",
    password="",
    database="school"
)
```

```
#cursor คือ ตัวสำหรับการจัดการระบบฐานข้อมูล
mycursor = mydb.cursor()
```

```
print(mydb)
```

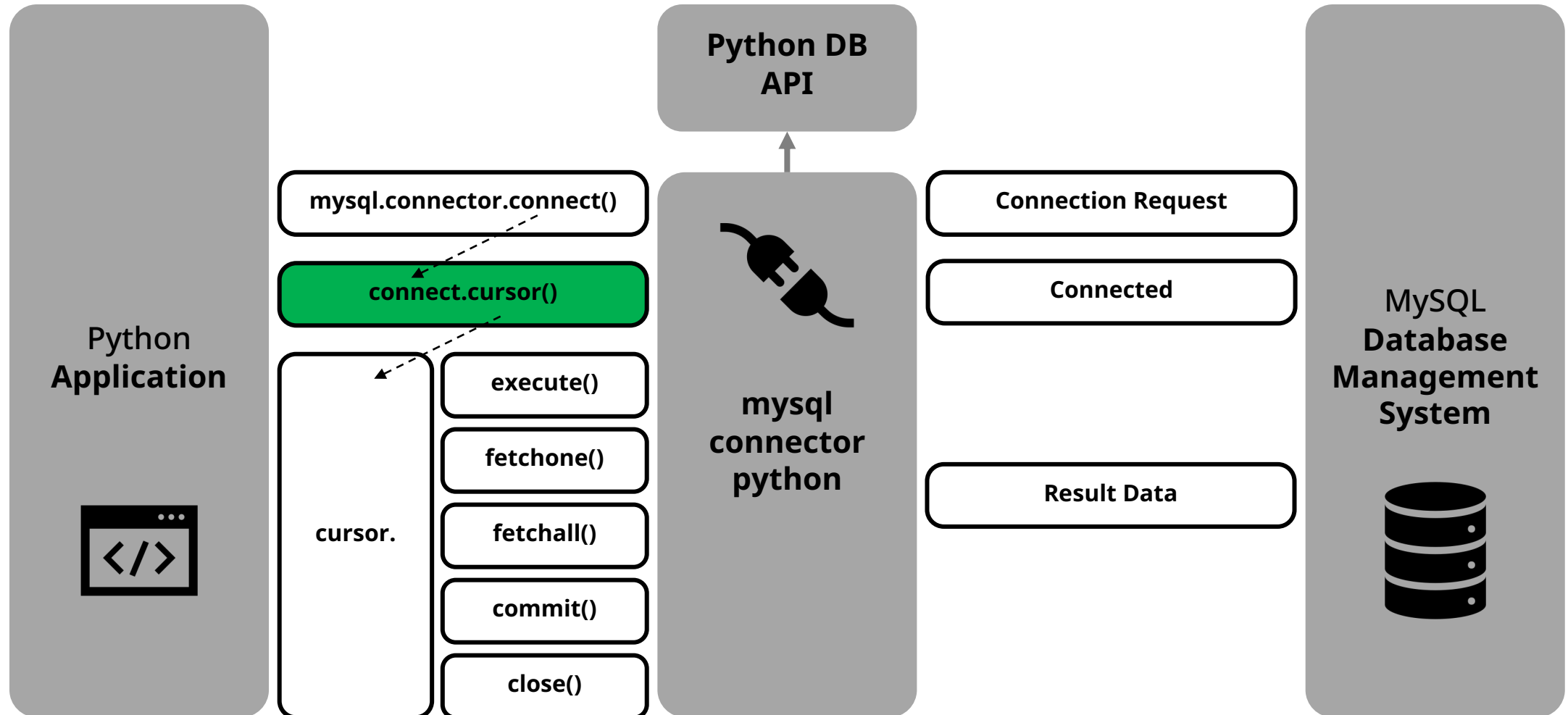
เชื่อมต่อสำเร็จ

```
school_system.py > ...
1  #การเชื่อมต่อระบบการจัดการฐานข้อมูล (MySQL) ชื่อว่า school
2  import mysql.connector
3  mydb = mysql.connector.connect(
4      host="localhost",
5      user="root",
6      password="",
7      database="school"
8  )
9  #cursor คือ ตัวสำหรับการจัดการระบบฐานข้อมูล
10 mycursor = mydb.cursor()
11 print(mydb)
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

```
● nuth@Nutthapats-MacBook-Pro TEST % /usr/local/bin/python3 /Users/nuth/Library/Caches/Python/Python39-64/Python39-64.nutthapats-MacBook-Pro/PYTHON/TEST/school_system.py
<mysql.connector.connection_cext.CMySQLConnection object at 0x1043b60c0>
```

# 1. การเขียนโปรแกรมเชื่อมต่อระบบการจัดการฐานข้อมูล



## 2. การเขียนโปรแกรมเพื่อดึงข้อมูลจากฐานข้อมูลออกมาแสดงผล

The screenshot shows the phpMyAdmin interface. On the left, the database structure is visible with 'school' and 'teacher' tables highlighted. The main panel displays a SQL query: `SELECT * FROM `teacher``. Below the query, there are options for 'Show all', 'Number of rows' (set to 25), 'Filter rows' (Search this table), and 'Sort by key' (None). The results table shows three rows of data, with the first two rows highlighted. The table has columns: t\_id, t\_name, t\_lastname, t\_email, and dep\_id.

| t_id  | t_name  | t_lastname    | t_email                 | dep_id |
|-------|---------|---------------|-------------------------|--------|
| 23771 | มงคลไชย | เทียนสุตินนท์ | mongkolchai@ru.ac.th    | 004    |
| 39154 | ณัฐภัทร | แก้วรัตนภัทร์ | nutthapat.ke@sstu.ac.th | 002    |
| 45545 | วิชุดา  | ขุนหนู        | wichuda.ku@ssru.ac.th   | 004    |

## 2. การเขียนโปรแกรมเพื่อดึงข้อมูลจากฐานข้อมูลออกมาแสดงผล

การเขียนโปรแกรมเพื่อดึงข้อมูลทั้งหมดในตาราง (table) ออกมาแสดงจะใช้ฟังก์ชันชื่อว่า `fetchall()` โดยที่คำว่า fetch แปลว่า ดึง และ all แปลว่า ทั้งหมด

#ฟังก์ชันสำหรับการดึงข้อมูลครูทั้งหมด (fetch all) ออกจากระบบฐานข้อมูลมาแสดงผล

```
def get_teacher():  
    mycursor.execute("SELECT * FROM teacher")  
    myresult = mycursor.fetchall()
```

```
    for datarow in myresult:  
        print(datarow)
```

---

`SELECT * FROM teacher`  
คือ คำสั่ง SQL ที่ใช้สำหรับการเลือกข้อมูลในทุกๆ column ออกมาจากตาราง teacher

คือ การวนซ้ำเพื่อนำค่าออกมาจากผลลัพธ์ที่ดึงออกมาจากฐานข้อมูล

```
get_teacher() #เรียกฟังก์ชันให้ทำงาน
```

## 2. การเขียนโปรแกรมเพื่อดึงข้อมูลจากฐานข้อมูลออกมาแสดงผล

school\_system.py > ...

```
1  #การเชื่อมต่อระบบการจัดการฐานข้อมูล (MySQL) ชื่อว่า school
2  import mysql.connector
3  mydb = mysql.connector.connect(
4      host="localhost",
5      user="root",
6      password="",
7      database="school"
8  )
```

9 #cursor คือ ตัวสำหรับการจัดการระบบฐานข้อมูล

```
10 mycursor = mydb.cursor()
```

```
11 print(mydb)
```

12

13 #ฟังก์ชันสำหรับการดึงข้อมูลครุทั้งหมด (fetch all) ออกจากระบบฐานข้อมูลมาแสดงผล

```
14 def get_teacher():
```

```
15     mycursor.execute("SELECT * FROM teacher")
```

```
16     myresult = mycursor.fetchall()
```

```
17
```

```
18     for datarow in myresult:
```

```
19         print(datarow)
```

```
20
```

```
21 get_teacher()
```

PROBLEMS

OUTPUT

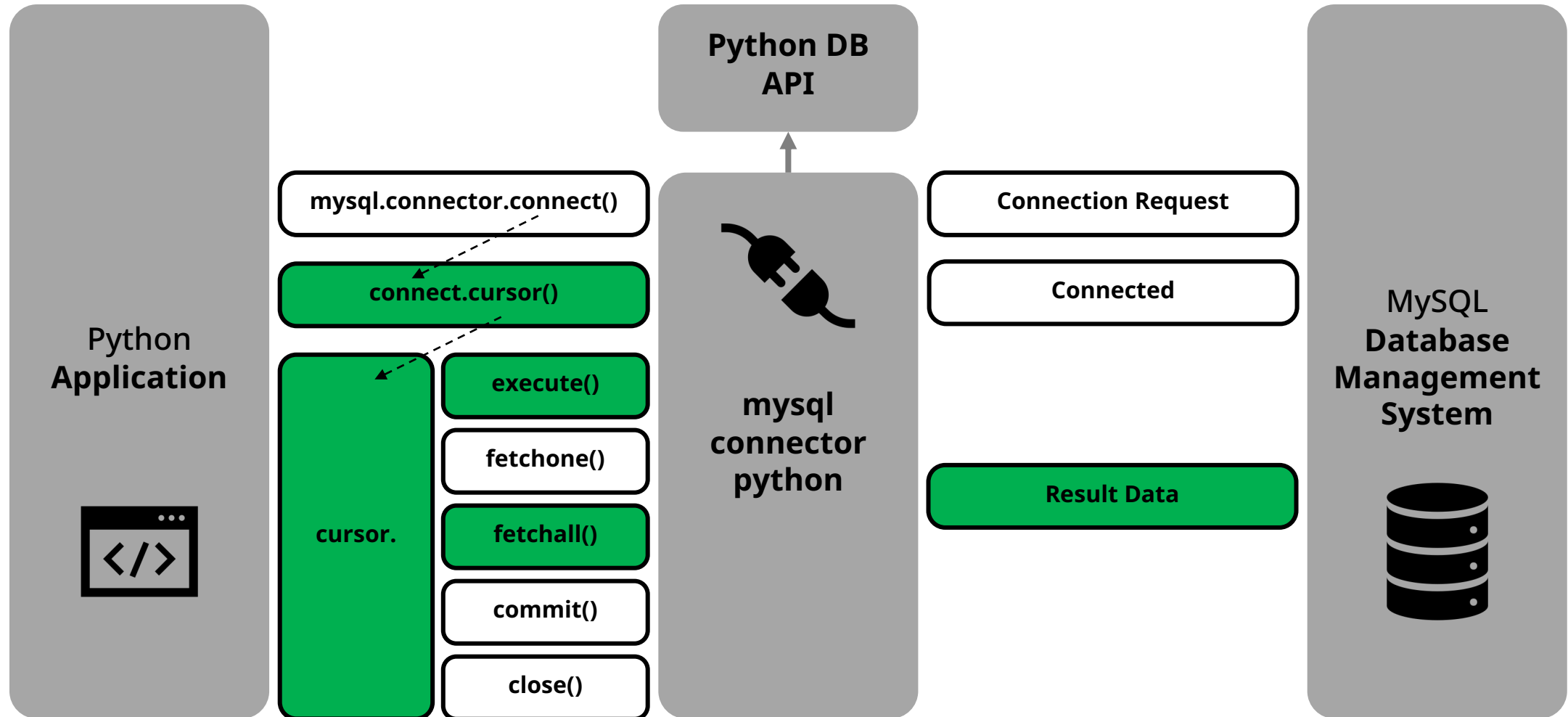
DEBUG CONSOLE

TERMINAL

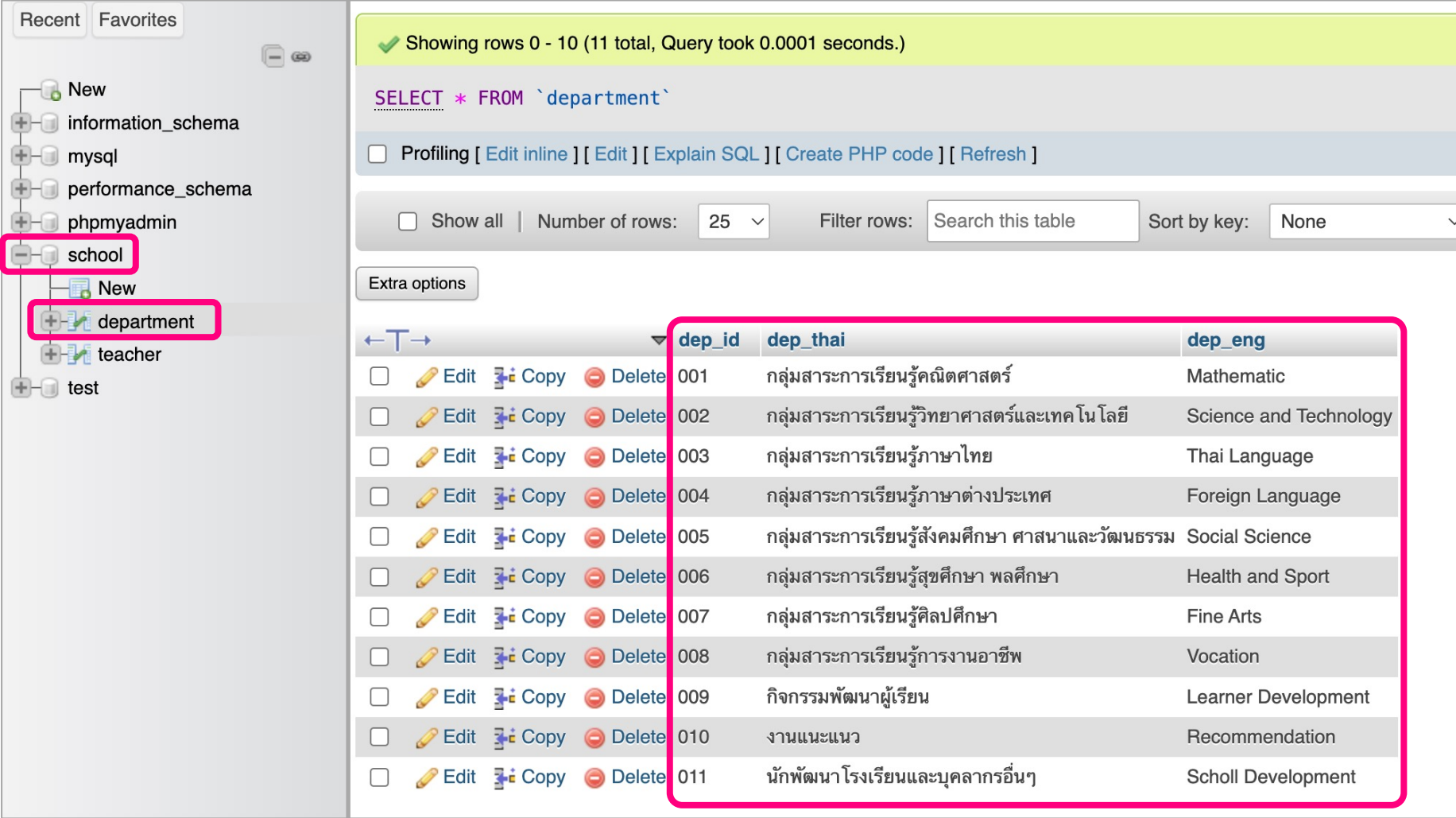
PORTS

```
• nuth@Nutthapats-MacBook-Pro TEST % /usr/local/bin/python3 /Users/nuth/Libra
ndhaRajabhatUniversity/PYTHON/TEST/school_system.py
mysql.connector.connection_cext.CMySQLConnection object at 0x102cf63f0>
('23771', 'มงคลไชย', 'เทียนสุดินนท์', 'mongkolchai@ru.ac.th', '004')
('39154', 'ณัฐภัทร', 'แก้วรัตนภัทร', 'nutthapat.ke@sstu.ac.th', '002')
('45545', 'วิชุดา', 'ชนหน', 'wichuda.ku@ssru.ac.th', '004')
o nuth@Nutthapats-MacBook-Pro TEST %
```

## 2. การเขียนโปรแกรมเพื่อดึงข้อมูลจากฐานข้อมูลออกมาแสดงผล



### 3. การเขียนโปรแกรมเพื่อดึงข้อมูลจากฐานข้อมูลออกมาแสดงผล



The screenshot shows the phpMyAdmin interface. On the left, the database structure is listed, with 'school' and its 'department' table highlighted. The main panel displays a query result for the 'department' table. The query is `SELECT * FROM `department``. The result shows 11 rows of data. The table has three columns: `dep_id`, `dep_thai`, and `dep_eng`. The data is as follows:

| dep_id | dep_thai                                        | dep_eng                |
|--------|-------------------------------------------------|------------------------|
| 001    | กลุ่มสาระการเรียนรู้คณิตศาสตร์                  | Mathematic             |
| 002    | กลุ่มสาระการเรียนรู้วิทยาศาสตร์และเทคโนโลยี     | Science and Technology |
| 003    | กลุ่มสาระการเรียนรู้ภาษาไทย                     | Thai Language          |
| 004    | กลุ่มสาระการเรียนรู้ภาษาต่างประเทศ              | Foreign Language       |
| 005    | กลุ่มสาระการเรียนรู้สังคมศึกษา ศาสนาและวัฒนธรรม | Social Science         |
| 006    | กลุ่มสาระการเรียนรู้สุขศึกษา พลศึกษา            | Health and Sport       |
| 007    | กลุ่มสาระการเรียนรู้ศิลปศึกษา                   | Fine Arts              |
| 008    | กลุ่มสาระการเรียนรู้การงานอาชีพ                 | Vocation               |
| 009    | กิจกรรมพัฒนาผู้เรียน                            | Learner Development    |
| 010    | งานแนะแนว                                       | Recommendation         |
| 011    | นักพัฒนา โรงเรียนและบุคลากรอื่นๆ                | Scholl Development     |

### 3. การเขียนโปรแกรมเพื่อดึงข้อมูลจากฐานข้อมูลออกมาแสดงผล

#ฟังก์ชันสำหรับการดึงข้อมูลกลุ่มสาระการเรียนรู้ทั้งหมด (fetch all) ออกจากระบบฐานข้อมูลมาแสดงผล

```
def get_department() :  
    mycursor.execute("SELECT * FROM department")  
    myresult = mycursor.fetchall()  
  
    for datarow in myresult:  
        print(datarow)
```

SELECT \* FROM department  
คือ คำสั่ง SQL ที่ใช้สำหรับการเลือกข้อมูลใน  
ทุกๆ columnk ออกมาจากตาราง  
department

คือ การวนซ้ำเพื่อนำค่าออกมาจาก  
ผลลัพธ์ที่ดึงออกมาจากฐานข้อมูล

---

```
get_department()
```

### 3. 8

```
10 mycursor = mydb.cursor()
11 print(mydb)
12
13 #ฟังก์ชันสำหรับการดึงข้อมูลครูทั้งหมด (fetch all) ออกจากระบบฐานข้อมูลมา
14 def get_teacher():
15     mycursor.execute("SELECT * FROM teacher")
16     myresult = mycursor.fetchall()
17
18     for datarow in myresult:
19         print(datarow)
```

```
#ฟังก์ชันสำหรับการดึงข้อมูลกลุ่มสาระการเรียนรู้ทั้งหมด (fetch all) ออกจากฐานข้อมูลมาแสดงผล
def get_department():
    mycursor.execute("SELECT * FROM department")
    myresult = mycursor.fetchall()

    for datarow in myresult:
        print(datarow)

get_department()
```

```

● nuth@nutthapats-MacBook-Pro TEST % /usr/local/bin/python3 /Users/nuth/Library/
ndhaRajabhatUniversity/PYTHON/TEST/school_system.py
mysql.connector.connection_cext.CMySQLConnection object at 0x1003bc630
('001', 'กลุ่มสาระการเรียนรู้คณิตศาสตร์', 'Mathematic')
('002', 'กลุ่มสาระการเรียนรู้วิทยาศาสตร์และเทคโนโลยี', 'Science and Technology')
('003', 'กลุ่มสาระการเรียนรู้ภาษาไทย', 'Thai Language')
('004', 'กลุ่มสาระการเรียนรู้ภาษาต่างประเทศ', 'Foreign Language')
('005', 'กลุ่มสาระการเรียนรู้สังคมศึกษา ศาสนาและวัฒนธรรม', 'Social Science')
('006', 'กลุ่มสาระการเรียนรู้สุขศึกษา พลศึกษา', 'Health and Sport')
('007', 'กลุ่มสาระการเรียนรู้ศิลปศึกษา', 'Fine Arts')
('008', 'กลุ่มสาระการเรียนรู้การงานอาชีพ', 'Vocation')
('009', 'กิจกรรมพัฒนาผู้เรียน', 'Learner Development')
('010', 'งานแนะแนว', 'Recommendation')
('011', 'นักพัฒนาโรงเรียนและบุคลากรอื่นๆ', 'Scholl Development')
● nuth@nutthapats-MacBook-Pro TEST % █

```

## 4. การเขียนโปรแกรมเพื่อดึงข้อมูลจากฐานข้อมูลออกมาแสดงผลแบบมีเงื่อนไข

การดึงข้อมูลจากฐานข้อมูลออกมา หลายครั้ง เราต้องการเฉพาะข้อมูลของครูบางคน เท่านั้น ไม่ได้ต้องการข้อมูลจากครูทุกคน หรือ ต้องการค้นหาข้อมูลครู 1 คนเท่านั้น

The screenshot shows the phpMyAdmin interface. On the left, the database structure is listed, with 'school' and 'teacher' tables highlighted. The main panel displays the 'teacher' table data. A yellow banner at the top indicates 'Showing rows 0 - 2 (3 total, Query took 0.0001 seconds.)'. Below this, the SQL query 'SELECT \* FROM `teacher`' is shown. The table data is as follows:

|                          | t_id  | t_name  | t_lastname    | t_email                 | dep_id |
|--------------------------|-------|---------|---------------|-------------------------|--------|
| <input type="checkbox"/> | 23771 | มงคลไชย | เทียนสุตินนท์ | mongkolchai@ru.ac.th    | 004    |
| <input type="checkbox"/> | 39154 | ณัฐภัทร | แก้วรัตนภัทร์ | nutthapat.ke@sstu.ac.th | 002    |
| <input type="checkbox"/> | 45545 | วิชุดา  | ขุนหนู        | wichuda.ku@ssru.ac.th   | 004    |

## 4. การเขียนโปรแกรมเพื่อดึงข้อมูลจากฐานข้อมูลออกมาแสดงผลแบบมีเงื่อนไข

#ฟังก์ชันสำหรับการค้นหาข้อมูลครูด้วยชื่อ

```
def search_teacher_by_name():  
    name = input("ป้อนชื่อครูเพื่อทำการค้นหาด้วยชื่อ: ")  
    mycursor.execute("SELECT * FROM teacher where t_name like %s", (name,))  
    myresult = mycursor.fetchall()  
  
    for datarow in myresult:  
        print(datarow)
```

ที่ซึ่ง t\_name เหมือนกับ



คือ คำสั่ง SQL ที่ใช้สำหรับการเลือกข้อมูลในทุกๆ column ออกมาจากตาราง teacher โดยที่ t\_name เหมือนกับ ค่าในตัวแปร name

---

`search_teacher_by_name()`

## 4. การเขียนโปรแกรมเพื่อดึงข้อมูลจากฐานข้อมูลออกมาแสดงผลแบบมีเงื่อนไข

```
29  #ฟังก์ชันสำหรับการค้นหาข้อมูลครูด้วยชื่อ
30  def search_teacher_by_name():
31      name = input("ป้อนชื่อครูเพื่อทำการค้นหาด้วยชื่อ: ")
32      mycursor.execute("SELECT * FROM teacher where t_name like %s",(name,))
33
34      myresult = mycursor.fetchall()
35
36      for datarow in myresult:
37          print(datarow)
38
39  search_teacher_by_name()
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

```
nuth@Nutthapats-MacBook-Pro TEST % /usr/local/bin/python3 /Users/nuth/Library/CloudStorage/ndhaRajabhatUniversity/PYTHON/TEST/school_system.py
<mysql.connector.connection_cext.CMySQLConnection object at 0x100e128d0>
ป้อนชื่อครูเพื่อทำการค้นหาด้วยชื่อ: ณัฐภัทร
('39154', 'ณัฐภัทร', 'แก้วรัตนภัทร', 'nutthapat.ke@sstu.ac.th', '002')
nuth@Nutthapats-MacBook-Pro TEST %
```

## 4. การเขียนโปรแกรมเพื่อดึงข้อมูลจากฐานข้อมูลออกมาแสดงผลแบบมีเงื่อนไข และการใช้ wildcard

```
29 #ฟังก์ชันสำหรับการค้นหาข้อมูลครด้วยชื่อ
30 def search_teacher_by_name():
31     name = input("ป้อนชื่อครูเพื่อทำการค้นหาด้วยชื่อ: ")
32     mycursor.execute("SELECT * FROM teacher where t_name like %s", ("%"+name+"%",))
33
34     myresult = mycursor.fetchall()
35
36     for datarow in myresult:
37         print(datarow)
38
39 search_teacher_by_name()
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

```
nuth@Nutthapats-MacBook-Pro TEST % /usr/local/bin/python3 /Users/nuth/Library/CloudStorage/One
ndhaRajabhatUniversity/PYTHON/TEST/school_system.py
<mysql.connector.connection_cext.CMySQLConnection object at 0x104d9a930>
ป้อนชื่อครูเพื่อทำการค้นหาด้วยชื่อ: ญั
('39154', 'ญัฏฐภัทร', 'แก้วรัตนภัทร', 'nutthapat.ke@sstu.ac.th', '002')
```

## 4. การเขียนโปรแกรมเพื่อดึงข้อมูลจากฐานข้อมูลออกมาแสดงผลแบบมีเงื่อนไข และการใช้ wildcard (Select)

#ฟังก์ชันสำหรับการค้นหาข้อมูลกลุ่มสาระการเรียนรู้ด้วยชื่อกลุ่มสาระ ภาษาไทย

```
def search_department_by_thai():  
    text = input("ป้อนชื่อกลุ่มสาระการเรียนรู้ (ภาษาไทย) เพื่อทำการค้นหา: ")  
    mycursor.execute("SELECT * FROM department where dep_thai like %s", ("%"+text+"%",))  
    myresult = mycursor.fetchall()  
  
    for datarow in myresult:  
        print(datarow)
```

---

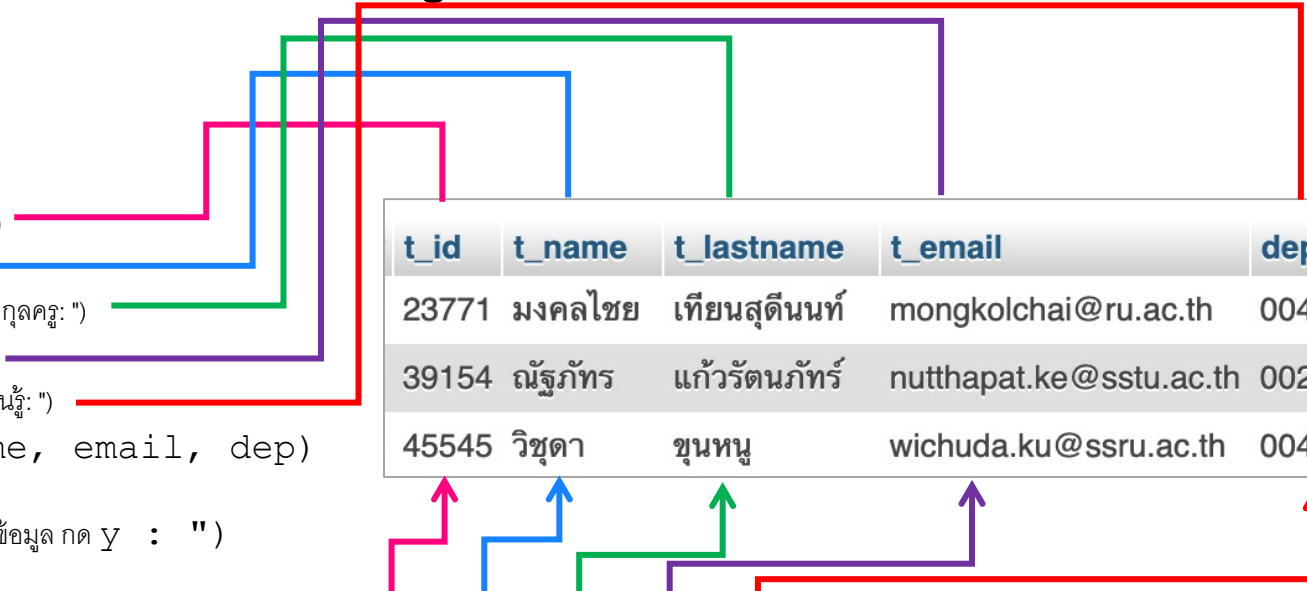
```
search_department_by_thai()
```

## 5. การเพิ่มข้อมูลเข้าไปในระบบฐานข้อมูล (Insert)

#ฟังก์ชันสำหรับการเพิ่มข้อมูลครู

```
def insert_teacher():
    print("ระบบการเพิ่มข้อมูลครู")
    id = input("ป้อนรหัสประจำตัวครู:")
    name = input("ป้อนชื่อครู:")
    lastname = input("ป้อนนามสกุลครู:")
    email = input("ป้อนอีเมลครู:")
    dep = input("ป้อนรหัสสาระการเรียนรู้:")
    print(id, name, lastname, email, dep)

    confirm = input("ต้องการเพิ่มข้อมูล กด y : ")
    if confirm=='y':
        sql = "INSERT INTO teacher VALUES (%s, %s, %s, %s, %s)"
        val = (id, name, lastname, email, dep)
        mycursor.execute(sql, val)
        mydb.commit()
        print("เพิ่มข้อมูลครูเรียบร้อยแล้ว")
        get_teacher()
    else:
        print("ยกเลิกการเพิ่มข้อมูลครู")
```



| t_id  | t_name  | t_lastname    | t_email                 | dep_id |
|-------|---------|---------------|-------------------------|--------|
| 23771 | มงคลไชย | เทียนสุดินนท์ | mongkolchai@ru.ac.th    | 004    |
| 39154 | ณัฐภัทร | แก้วรัตนภัทร์ | nutthapat.ke@sstu.ac.th | 002    |
| 45545 | วิชุดา  | ขุนหนู        | wichuda.ku@ssru.ac.th   | 004    |

**INSERT INTO teacher VALUES**  
คือคำสั่ง SQL ที่ใช้ในการแทรกเพิ่มข้อมูลเข้าไปยังตารางที่กำหนด  
ในที่นี้คือตารางชื่อ teacher เมื่อ execute แล้ว ต้องเพิ่มคำสั่ง  
commit() ทุกครั้ง เพื่อกำการเพิ่มข้อมูลลงไปยังฐานข้อมูล

**insert\_teacher()**

## 5. การเพิ่มข้อมูลเข้าไปในระบบฐานข้อมูล (Insert)

```
48
49 #ฟังก์ชันสำหรับการเพิ่มข้อมูลครู
50 def insert_teacher():
51     print("ระบบการเพิ่มข้อมูลครู")
52     id = input("ป้อนรหัสประจำตัวครู: ")
53     name = input("ป้อนชื่อครู: ")
54     lastname = input("ป้อนนามสกุลครู: ")
55     email = input("ป้อนอีเมลครู: ")
56     dep = input("ป้อนรหัสสาระการเรียนรู้: ")
57     print(id, name, lastname, email, dep)
58     confirm = input("ต้องการเพิ่มข้อมูล กด y : ")
59     if confirm=='y':
60         sql = "INSERT INTO teacher VALUES (%s, %s, %s, %s, %s)"
61         val = (id, name, lastname, email, dep)
62         mycursor.execute(sql, val)
63         mydb.commit()
64         print("เพิ่มข้อมูลครูเรียบร้อยแล้ว")
65         get_teacher()
66     else:
67         print("ยกเลิกการเพิ่มข้อมูลครู")
68
69 insert_teacher()
```

```
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

<mysql.connector.connection_cext.CMySQLConnection object at 0x104d07560>
ระบบการเพิ่มข้อมูลครู
ป้อนรหัสประจำตัวครู: 22222
ป้อนชื่อครู: ภาษา
ป้อนนามสกุลครู: เชี่ยวชาญ
ป้อนอีเมลครู: phachaya.ch@ssru.ac.th
ป้อนรหัสสาระการเรียนรู้: 002
22222 ภาษา เชี่ยวชาญ phachaya.ch@ssru.ac.th 002
ต้องการเพิ่มข้อมูล กด y : y
เพิ่มข้อมูลครูเรียบร้อยแล้ว
('11111', 'จารุมณ', 'หนูคง', 'jarumon.no@ssru.ac.th', '002')
('22222', 'ภาษา', 'เชี่ยวชาญ', 'phachaya.ch@ssru.ac.th', '002')
('23771', 'มงคลไชย', 'เทียนสุตินนท์', 'mongkolchai@ru.ac.th', '004')
('39154', 'ณัฐภัทร', 'แก้วรัตนภัทร', 'nutthapat.ke@sstu.ac.th', '002')
('45545', 'วิชุดา', 'ขุนทด', 'wichuda.ku@ssru.ac.th', '004')
nuth@Nutthapats-MacBook-Pro TEST %
```

## 6. การปรับปรุงข้อมูลในฐานข้อมูล (Update)

สรุปขั้นตอน

1. แสดงข้อมูลครูทั้งหมดก่อน เพื่อให้ผู้ใช้ได้อ่านและทราบว่าต้องเรียกรหัสครูคนใดมาปรับปรุงข้อมูล
2. รับค่ารหัสประจำตัวครู เพื่อนำไปค้นหาและทำให้ทราบว่าต้องการปรับปรุงข้อมูลครูคนใด
3. รับค่าข้อมูลครู ประกอบไปด้วย t\_name, t\_lastname, t\_email, dep\_id
4. ยืนยัน และปรับปรุงข้อมูลข้อมูลครูเข้าไปยังฐานข้อมูล

## 6. การปรับปรุงข้อมูลในฐานข้อมูล (Update)

#ฟังก์ชันสำหรับการปรับปรุงข้อมูลครู

```
def update_teacher():
    print("-"*25)
    print("ระบบการปรับปรุงข้อมูลครู")
    print("-"*25)
    get_teacher() #แสดงข้อมูลครูก่อน เพื่อดูก่อนการเลือกข้อมูลครูขึ้นมาปรับปรุง
    print("-"*25)
```

```
id = input("ป้อนรหัสข้อมูลครู ที่ต้องการปรับปรุงข้อมูล: ")
```

```
mycursor.execute("SELECT * FROM teacher where
t_id = %s", (id,))
myresult = mycursor.fetchall()
```

```
for datarow in myresult:
    print(datarow)
```

```
print("-"*25)
name = input("ป้อนชื่อครู:")
lastname = input("ป้อนนามสกุลครู:")
email = input("ป้อนอีเมลครู:")
dep = input("ป้อนรหัสสาระการเรียนรู้:")
print("-"*25)
print(id, name, lastname, email, dep)
```

```
confirm = input("ต้องการเพิ่มข้อมูล กดย : ")
```

```
if confirm=='y':
    sql = "UPDATE teacher SET t_name = %s,
t_lastname=%s, t_email=%s, dep_id=%s WHERE
t_id = %s "
    val = (name, lastname, email, dep,
myresult[0][0])
    mycursor.execute(sql, val)
    mydb.commit()
    print("ปรับข้อมูลครูเรียบร้อยแล้ว")
    get_teacher()
else:
    print("ยกเลิกการปรับปรุงข้อมูลครู")
```

update\_teacher()

## 6. การปรับปรุงข้อมูลในฐานข้อมูล (Update)

#ฟังก์ชันสำหรับการปรับปรุงข้อมูลครู

```
def update_teacher():
```

```
    print("-"*25)
```

```
    print("ระบบการปรับปรุงข้อมูลครู")
```

```
    print("-"*25)
```

```
    get_teacher() #แสดงข้อมูลครูก่อน เพื่อใช้ดูก่อนการเลือกข้อมูลครูขึ้นมาปรับปรุง
```

```
    print("-"*25)
```

```
    id = input("ป้อนรหัสข้อมูลครู ที่ต้องการปรับปรุงข้อมูล: ")
```

```
    mycursor.execute("SELECT * FROM teacher where t_id = %s",(id,))
```

```
    myresult = mycursor.fetchall()
```

```
    for datarow in myresult:
```

```
        print(datarow)
```

```
    print("-"*25)
```

```
    name = input("ป้อนชื่อครู: ")
```

```
    lastname = input("ป้อนนามสกุลครู: ")
```

```
    email = input("ป้อนอีเมลครู: ")
```

```
    dep = input("ป้อนรหัสสาระการเรียนรู้: ")
```

```
    print("-"*25)
```

```
    print(id, name, lastname, email, dep)
```

```
    confirm = input("ต้องการเพิ่มข้อมูล กด y : ")
```

```
    if confirm=='y':
```

```
        sql = "UPDATE teacher SET t_name = %s, t_lastname=%s, t_email=%s, dep_id=%s WHERE t_id = %s "
```

```
        val = (name, lastname, email, dep, myresult[0][0])
```

```
        mycursor.execute(sql, val)
```

```
        mydb.commit()
```

```
        print("ปรับข้อมูลครูเรียบร้อยแล้ว")
```

```
        get_teacher()
```

```
    else:
```

```
        print("ยกเลิกการปรับปรุงข้อมูลครู")
```

```
update_teacher()
```

```
nuth@Nutthapats-MacBook-Pro TEST % /usr/local/bin/python3 /Users/nuth/Library/Python/TEST/school_system.py  
<mysql.connector.connection_cext.CMySQLConnection object at 0x10078c2c0>
```

ระบบการปรับปรุงข้อมูลครู

```
(('11111', 'j', 'n', 'j@mail', '002'))  
(('22222', 'ภาชญา', 'เชียวชาญ', 'phachaya.ch@ssru.ac.th', '002'))  
(('23771', 'มงคลไชย', 'เทียนสุตินนท์', 'mongkolchai@ru.ac.th', '004'))  
(('39154', 'ณัฐภัทร', 'แก้วรัตนภัทร', 'nutthapat.ke@sstu.ac.th', '002'))  
(('45545', 'วิชุดา', 'ขุนหนู', 'wichuda.ku@ssru.ac.th', '004'))
```

ป้อนรหัสข้อมูลครู ที่ต้องการปรับปรุงข้อมูล: 11111  
(('11111', 'j', 'n', 'j@mail', '002'))

ป้อนชื่อครู: Jarumon  
ป้อนนามสกุลครู: Nookhong  
ป้อนอีเมลครู: jarumon.no@ssru.ac.th  
ป้อนรหัสสาระการเรียนรู้: 002

11111 Jarumon Nookhong jarumon.no@ssru.ac.th 002

ต้องการเพิ่มข้อมูล กด y : y

ปรับข้อมูลครูเรียบร้อยแล้ว

```
(('11111', 'Jarumon', 'Nookhong', 'jarumon.no@ssru.ac.th', '002'))  
(('22222', 'ภาชญา', 'เชียวชาญ', 'phachaya.ch@ssru.ac.th', '002'))  
(('23771', 'มงคลไชย', 'เทียนสุตินนท์', 'mongkolchai@ru.ac.th', '004'))  
(('39154', 'ณัฐภัทร', 'แก้วรัตนภัทร', 'nutthapat.ke@sstu.ac.th', '002'))  
(('45545', 'วิชุดา', 'ขุนหนู', 'wichuda.ku@ssru.ac.th', '004'))
```

```
nuth@Nutthapats-MacBook-Pro TEST %
```

## 7. การลบข้อมูลในฐานข้อมูล (Delete)

#ฟังก์ชันสำหรับการลบข้อมูลครู

```
def delete_teacher():
    print("-"*25)
    print("ระบบการลบข้อมูลครู")
    print("-"*25)
    get_teacher() # แสดงข้อมูลครูก่อน เพื่อดูก่อนการเลือกข้อมูลครูขึ้นมา
    ลบ
    print("-"*25)

    id = input("ป้อนรหัสข้อมูลครู ที่ต้องการลบข้อมูล:")
    mycursor.execute("SELECT * FROM teacher
where t_id = %s", (id,))
    myresult = mycursor.fetchall()

    for datarow in myresult:
        print(datarow)
    print("-"*25)
```

```
confirm = input("ต้องการลบข้อมูล กด y : ")
if confirm=='y':
    sql = "DELETE FROM teacher WHERE t_id
= %s"
    val = (id,)
    mycursor.execute(sql, val)
    mydb.commit()
    print("ลบข้อมูลครูเรียบร้อยแล้ว")
    get_teacher()
else:
    print("ยกเลิกการลบข้อมูลครู")
```

---

**delete\_teacher()**

## 7. การลบข้อมูลในฐานข้อมูล (Delete)

#ฟังก์ชันสำหรับการลบข้อมูลครู

def delete\_teacher():

print("-"\*25)

print("ระบบการลบข้อมูลครู")

print("-"\*25)

get\_teacher() #แสดงข้อมูลครูก่อน เพื่อให้ดูก่อนการเลือกข้อมูลครูขึ้นมอลบ

print("-"\*25)

id = input("ป้อนรหัสข้อมูลครู ที่ต้องการลบข้อมูล: ")

mycursor.execute("SELECT \* FROM teacher where t\_id = %s", (id,))

myresult = mycursor.fetchall()

for datarow in myresult:

print(datarow)

print("-"\*25)

confirm = input("ต้องการลบข้อมูล กด y : ")

if confirm=='y':

sql = "DELETE FROM teacher WHERE t\_id = %s"

val = (id,)

mycursor.execute(sql, val)

mydb.commit()

print("ลบข้อมูลครูเรียบร้อยแล้ว")

get\_teacher()

else:

print("ยกเลิกการลบข้อมูลครู")

delete\_teacher()

nuth@Nutthapats-MacBook-Pro TEST % /usr/local/bin/python3 /Users/nuth/L  
<mysql.connector.connection\_cext.CMySQLConnection object at 0x102a1e5d0>

ระบบการลบข้อมูลครู

-----  
( '11111', 'Jarumon', 'Nookhong', 'jarumon.no@ssru.ac.th', '002' )  
( '22222', 'ภาชญา', 'เชี้ยวชาญ', 'phachaya.ch@ssru.ac.th', '002' )  
( '23771', 'มงคลไชย', 'เทียนสุตินนท์', 'mongkolchai@ru.ac.th', '004' )  
( '39154', 'ณัฐภัทร', 'แก้วรัตนภัทร', 'nutthapat.ke@sstu.ac.th', '002' )  
( '45545', 'วิชุดา', 'ขุนหนู', 'wichuda.ku@ssru.ac.th', '004' )  
-----

ป้อนรหัสข้อมูลครู ที่ต้องการลบข้อมูล: 11111

( '11111', 'Jarumon', 'Nookhong', 'jarumon.no@ssru.ac.th', '002' )

ต้องการลบข้อมูล กด y : y

ลบข้อมูลครูเรียบร้อยแล้ว

-----  
( '22222', 'ภาชญา', 'เชี้ยวชาญ', 'phachaya.ch@ssru.ac.th', '002' )  
( '23771', 'มงคลไชย', 'เทียนสุตินนท์', 'mongkolchai@ru.ac.th', '004' )  
( '39154', 'ณัฐภัทร', 'แก้วรัตนภัทร', 'nutthapat.ke@sstu.ac.th', '002' )  
( '45545', 'วิชุดา', 'ขุนหนู', 'wichuda.ku@ssru.ac.th', '004' )  
-----

nuth@Nutthapats-MacBook-Pro TEST %

## 8. การดึงข้อมูลผสมกันหลายตาราง

ในการเรียกดูหรือค้นคืนข้อมูลหลายๆ ครั้ง จะมีการดึงข้อมูลจากหลายตารางร่วมกัน เช่น หากต้องการข้อมูลครูที่อยู่ในกลุ่มสาระการเรียนรู้ วิทยาศาสตร์และเทคโนโลยี เพื่อแสดงผล ดังนี้

```
-----  
School Management System  
-----
```

```
ข้อมูลครู
```

```
1 : แสดงรายชื่อครูทั้งหมด
```

```
2 : ค้นหารายชื่อครู
```

```
3 : เพิ่มข้อมูลครู
```

```
4 : ปรับปรุงข้อมูลครู
```

```
5 : ลบข้อมูลครู
```

```
กลุ่มสาระการเรียนรู้
```

```
6 : แสดงกลุ่มสาระการเรียนรู้ทั้งหมด
```

```
7 : ค้นหากลุ่มสาระการเรียนรู้
```

```
8 : ค้นหาข้อมูลครูจากกลุ่มสาระการเรียนรู้
```

```
ออกจากระบบ กด 0
```

```
-----  
ป้อนเมนูที่ต้องการ: 8
```

```
ป้อนชื่อกลุ่มสาระการเรียนรู้ (ภาษาไทย) เพื่อทำการแสดงรายชื่อครู: วิท
```

```
('22222', 'ภาษา', 'เขียวชาญ', 'phachaya.ch@ssru.ac.th', '002', 'กลุ่มสาระการเรียนรู้วิทยาศาสตร์และเทคโนโลยี')
```

```
('39154', 'ณัฐภัทร', 'แก้วรัตนภัทร', 'nutthapat.ke@sstu.ac.th', '002', 'กลุ่มสาระการเรียนรู้วิทยาศาสตร์และเทคโนโลยี')
```

## 8. การดึงข้อมูลผสมกันหลายตาราง

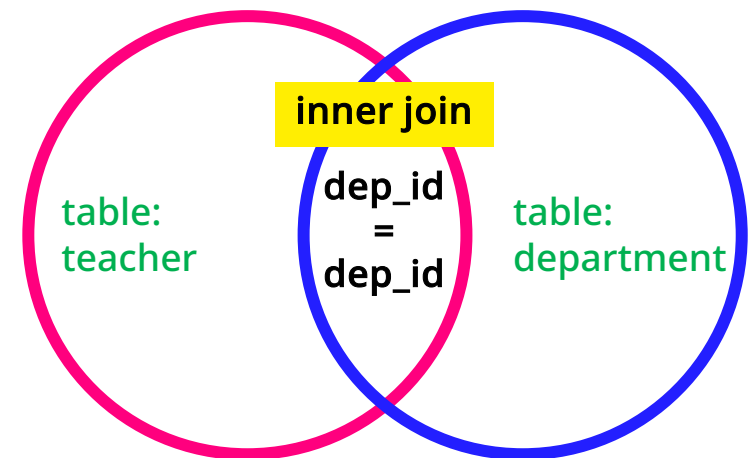
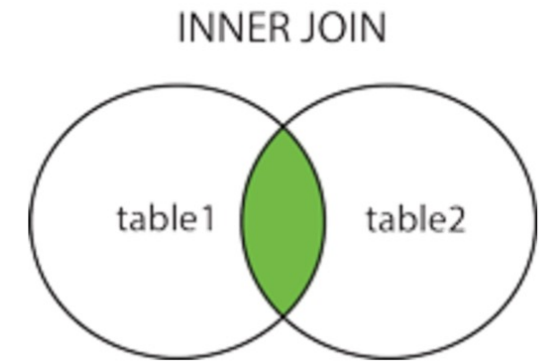
#ฟังก์ชันสำหรับการแสดงข้อมูลครูตามสาระการเรียนรู้ที่ต้องการ

```
def search_teacher_by_department():  
    text = input("ป้อนชื่อกลุ่มสาระการเรียนรู้ (ภาษาไทย) เพื่อทำการแสดงรายชื่อครู:")
```

```
mycursor.execute("SELECT teacher.*, department.dep_thai FROM teacher  
INNER JOIN department ON teacher.dep_id = department.dep_id  
WHERE department.dep_thai like %s", ("%"+text+"%",))
```

```
myresult = mycursor.fetchall()
```

```
for datarow in myresult:  
    print(datarow)
```



## 8. การดึงข้อมูลผสมกันหลายตาราง

```
52 #ฟังก์ชันสำหรับการแสดงข้อมูลครูตามสาระการเรียนรู้ที่ต้องการ
53 def search_teacher_by_department():
54     text = input("ป้อนชื่อกลุ่มสาระการเรียนรู้ (ภาษาไทย) เพื่อทำการแสดงรายชื่อครู: ")
55     mycursor.execute("SELECT teacher.*, department.dep_thai FROM teacher \
56                       INNER JOIN department ON teacher.dep_id = department.dep_id \
57                       WHERE department.dep_thai like %s", ("%"+text+"%",))
58     #เครื่องหมาย \ ใช้กรณีที่ code ยาวไปแล้วต้องการเขียน code ให้สั้นบรรทัดใหม่
59     myresult = mycursor.fetchall()
60
61     for datarow in myresult:
62         print(datarow)
63
64
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

ข้อมูลครู

1 : แสดงรายชื่อครูทั้งหมด

2 : ค้นหารายชื่อครู

3 : เพิ่มข้อมูลครู

4 : ปรับปรุงข้อมูลครู

5 : ลบข้อมูลครู

กลุ่มสาระการเรียนรู้

6 : แสดงกลุ่มสาระการเรียนรู้ทั้งหมด

7 : ค้นหากลุ่มสาระการเรียนรู้

8 : ค้นหาข้อมูลครูจากกลุ่มสาระการเรียนรู้

ออกจากระบบ กด 0

ป้อนเมนูที่ต้องการ: 8

ป้อนชื่อกลุ่มสาระการเรียนรู้ (ภาษาไทย) เพื่อทำการแสดงรายชื่อครู: วิทย์

('22222', 'ภาษา', 'เชี่ยวชาญ', 'phachaya.ch@ssru.ac.th', '002', 'กลุ่มสาระการเรียนรู้วิทยาศาสตร์และเทคโนโลยี')

('39154', 'คณิต', 'แก้วรัตนภัทร์', 'nutthapat.ke@ssu.ac.th', '002', 'กลุ่มสาระการเรียนรู้วิทยาศาสตร์และเทคโนโลยี')

## 9. การสร้างเมนูเพื่อการใช้งานที่สะดวก

```
def menu() :  
    print("-"*25)  
    print("School Management System")  
    print("-"*25)  
    print("ข้อมูลครู")  
    print("1 : แสดงรายชื่อครูทั้งหมด")  
    print("2 : ค้นหารายชื่อครู")  
    print("3 : เพิ่มข้อมูลครู")  
    print("4 : ปรับปรุงข้อมูลครู")  
    print("5 : ลบข้อมูลครู")  
    print("กลุ่มสาระการเรียนรู้")  
    print("6 : แสดงกลุ่มสาระการเรียนรู้ทั้งหมด")  
    print("7 : ค้นหากลุ่มสาระการเรียนรู้")  
    print("ออกจากระบบ กด 0")  
    print("-"*25)
```

```
choice = input("ป้อนเมนูที่ต้องการ: ")  
if choice=="1":  
    get_teacher() เรียกแต่ละฟังก์ชันที่ได้สร้างขึ้น  
elif choice=="2":  
    search_teacher_by_name() เรียกแต่ละฟังก์ชันที่ได้สร้างขึ้น  
elif choice=="3":  
    insert_teacher() เรียกแต่ละฟังก์ชันที่ได้สร้างขึ้น  
elif choice=="4":  
    update_teacher() เรียกแต่ละฟังก์ชันที่ได้สร้างขึ้น  
elif choice=="5":  
    delete_teacher() เรียกแต่ละฟังก์ชันที่ได้สร้างขึ้น  
else:  
    exit() #ออกจากโปรแกรม
```

---

```
while True:  
    menu() สั่งให้โปรแกรมวนลูปแสดงเมนูไปเรื่อยๆ  
จนกว่าจะได้รับคำสั่ง exit()
```

## 10. หากต้องการแสดงผลให้อยู่ในรูปแบบตารางสามารถใช้ prettytable

```
import mysql.connector
from prettytable import PrettyTable
```

#ฟังก์ชันสำหรับการดึงข้อมูลครู่ทั้งหมด (fetch all) ออกจากระบบฐานข้อมูลมาแสดงผลในรูปแบบตาราง prettytable

```
def get_table_teacher():
    mycursor = mydb.cursor()
    mycursor.execute("SELECT * FROM teacher")
    myresult = mycursor.fetchall()
```

```
columns = []
```

สร้างตัวแปร list สำหรับการเก็บชื่อ column

```
for col in mycursor.description:
    columns.append(col[0])
```

วนลูปเพื่อดึงค่าชื่อ column ออกมาใส่ต่อๆ กัน

```
tb = PrettyTable(columns)
```

สร้างตารางโดยให้มีหัว column ตามที่ได้ดึงออกมา

```
for row in myresult:
    tb.add_row(row)
```

วนลูปเพื่อดึงค่าแถวข้อมูลมาใส่ในตารางที่สร้างขึ้น

```
print(tb)
```

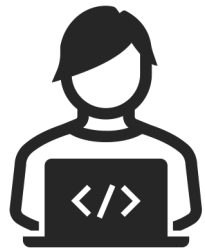
สั่งแสดงผลตาราง

```
del tb
```

ลบตารางเมื่อใช้งานเสร็จแล้ว

| t_id  | t_name  | t_lastname    | t_email                 | dep_id |
|-------|---------|---------------|-------------------------|--------|
| 22222 | ภาชญา   | เขียวชาญ      | phachaya.ch@ssru.ac.th  | 002    |
| 23771 | มงคลไชย | เทียนสุตินนท์ | mongkolchai@ru.ac.th    | 004    |
| 39154 | ณัฐภัทร | แก้วรัตนภัทร์ | nutthapat.ke@sstu.ac.th | 002    |
| 45545 | วิชุดา  | ขุนหนู        | wichuda.ku@ssru.ac.th   | 004    |

# ไฟล์สำหรับการเรียนรู้



[https://cutt.ly/python\\_ajnuith](https://cutt.ly/python_ajnuith)

1. ไฟล์นามสกุล .py คือ ไฟล์โปรแกรมภาษา Python
2. ไฟล์นามสกุล .sql คือ ไฟล์โปรแกรมระบบฐานข้อมูลที่น่าไปใช้ใน MySQL / MariaDB โดยให้ทำการสร้างฐานข้อมูลชื่อ school ก่อนการ import ไฟล์ sql เข้าไป