



รายงาน MiniProject

เรื่อง Student management system

นาย รตน จำปาสา รหัสนักศึกษา 60070080

นาย ศุภมิตร บัวศรีแก้ว รหัสนักศึกษา 60070100

นาย ไอศูรย์ ทิมศรี รหัสนักศึกษา 60070121

เสนอ

ผศ.ดร. ธนิตา นุ่มนนท์

รายวิชา 06016317 OBJECT-ORIENTED PROGRAMMING

สาขาวิชาเทคโนโลยีสารสนเทศ(ภาคปกติ)

คณะเทคโนโลยีสารสนเทศ

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

บทนำ

ในปัจจุบันเทคโนโลยีและคอมพิวเตอร์เข้ามามีบทบาทสำคัญในการดำเนินชีวิต ไม่ว่าจะเป็นการเรียนหรือการทำงาน

โรงเรียนมัธยมศึกษาหลายๆแห่งภายในประเทศไทยมีระบบทรัพยากรบุคคลสำหรับนักเรียนและคณาจารย์ภายในโรงเรียนซึ่งเป็นระบบทรัพยากรบุคคลที่ครอบคลุมข้อมูลส่วนตัวนักเรียน, ข้อมูลส่วนตัวอาจารย์, ข้อมูลคะแนนของนักเรียน, คะแนนความประพฤติและคะแนนอื่นๆ ซึ่งในโครงการนี้กลุ่มเราได้นำระบบทรัพยากรบุคคลของโรงเรียนมัศึกษามาดำเนินต่อ โดยเราได้สร้างโปรแกรมซึ่งจัดเก็บข้อมูลที่จำเป็นสำหรับทรัพยากรบุคคลสำหรับนักเรียนเพื่อเพิ่มความสะดวกแก่คณาจารย์, นักเรียนและผู้ปกครองในการจัดการและตรวจสอบข้อมูลต่างๆของนักเรียน

วัตถุประสงค์และเป้าหมาย

1. เพื่อพัฒนาระบบทรัพยากรบุคคลของนักเรียนภายในโรงเรียนมัธยมศึกษา
2. เพื่อเพิ่มความสะดวกในการทำงานในระบบจัดการนักเรียน
3. เพื่อเพิ่มความสะดวกของอาจารย์ในการดูแลนักเรียน
4. เพื่อเพิ่มความสะดวกของนักเรียนในการเข้าถึงข้อมูลที่จำเป็นและเกี่ยวข้องกับตนเอง
5. เพื่อให้ได้ซอฟต์แวร์ต้นแบบที่ซึ่งสามารถนำไปประยุกต์ใช้ได้จริง

ประโยชน์ที่คาดว่าจะได้รับและการนำไปใช้งาน

- 1.พัฒนาทักษะการเขียนโปรแกรมเชิงวัตถุ
- 2.พัฒนาทักษะการออกแบบโปรแกรมเชิงวัตถุ
- 3.สถานศึกษามีเทคโนโลยีเข้ามาช่วยในการจัดการ
- 4.ช่วยให้การสื่อสารกับผู้เรียนได้สะดวกขึ้น

รายละเอียดโปรแกรมที่จะพัฒนาเชิงเทคนิค

เป็นโปรแกรมที่ทำให้ อาจารย์ภายในโรงเรียนสามารถ เข้าถึงคะแนนของนักเรียนได้ผ่านทางโปรแกรมที่เราได้สร้างขึ้น และนักเรียนสามารถเรียกดูข้อมูลของตนเองและคะแนนของตนเองในรายวิชาได้ผ่านโปรแกรม โดยโปรแกรมจะมี ผู้ใช้งาน3กลุ่มหลัก

1.Admin โดยแอดมินนำข้อมูลของนักเรียนและครูอาจารย์ลงระบบ โดยจัดเก็บ โดย database ให้ง่ายต่อการจัดการข้อมูล

2.อาจารย์ โดยอาจารย์จะมีหน้าที่ดูข้อมูลของนักเรียนที่ตนได้สอนในรายวิชานั้นและสามารถกรอกคะแนนเก็บของนักเรียนคนนั้นๆ ได้ผ่านทางโปรแกรม

3.นักเรียน โดยนักเรียนจะทำการเข้าสู่ข้อมูลส่วนตัว และ คะแนน ของตนเองผ่านทางโปรแกรมโดย User และ Password จะได้รับผ่านทาง admin

Input/Output/Specification

Input : รหัสนักเรียน, รหัสอาจารย์, คะแนนที่กรอกให้นักเรียน

Output: ข้อมูลของนักเรียน/อาจารย์, คะแนนเก็บ, เกรดที่คำนวณ ณ ปัจจุบัน

Functional Specification

จะแบ่งออกเป็น3ส่วน ส่วนที่1 ส่วนของ Admin โดยที่หน้าที่ นำข้อมูลของ นักเรียน และอาจารย์เพิ่มลงไปยัง ฐานข้อมูล ส่วนที่2 ส่วนของ อาจารย์ โดยมีหน้าที่ ดูข้อมูลของนักเรียนตามห้องที่

สอนตามรายวิชา และ กรอกคะแนน โดยคะแนนจะถูกจัดเก็บในฐานข้อมูล ส่วนที่3 ส่วนของ นักเรียน โดยมีหน้าที่ ดูข้อมูลประวัติส่วนตัวของตนเอง และ ข้อมูลคะแนนเก็บ และ เกรด ตามแต่ละรายวิชา

ตารางรับผิดชอบการทำงานและระยะเวลาการทำงาน

รายชื่อสมาชิก	ส่วนที่รับผิดชอบ	ระยะเวลา
60070080 นาย รตน จำปาสา	ออกแบบภาพรวมของโปรแกรม, ออกแบบUI ทำระบบฐานข้อมูล	23/10/62 - 3/11/62 21/11/62 – 27/11/62
60070100 นาย สุกุมิตร์ บัวศรีแก้ว	เขียนโค้ดตามแบบแปลนของโปรแกรม ทำระบบlogin	3/11/62 – 15/11/62 28/11/62 - 30/11/62
60070121 นาย ไชยสุรย์ ทิมศรี	เขียนโค้ดออกแบบGUI เขียนหน้าต่างผู้ใช้งาน	6/11/62 – 17/11/62 1/12/62 – 4/12/62
-	ทดสอบระบบ	5/12/62 – 7/12/62
-	แก้ไขข้อผิดพลาด	8/12/62 – 15/12/62

ขอบเขตและข้อจำกัดของโปรแกรมที่จะพัฒนา

- ใช้ได้สำหรับเฉพาะนักเรียนภายในโรงเรียนและอาจารย์
- ในอนาคตจะมีการพัฒนา คะแนนในส่วนอื่นๆนอกจากคะแนนในห้องเรียน

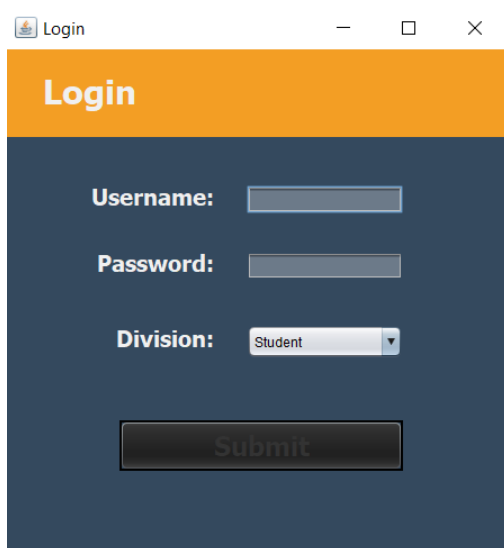
Student management system คือโปรแกรมที่ใช้สำหรับจัดการข้อมูลและคะแนนของนักเรียนให้อยู่ในโปรแกรมโปรแกรมให้เพื่อให้สะดวกต่อการใช้งานของ อาจารย์ และ นักเรียนภายในโรงเรียน โดยใช้ Database เข้ามาช่วยในการจัดเก็บข้อมูล

ภาษาที่ใช้



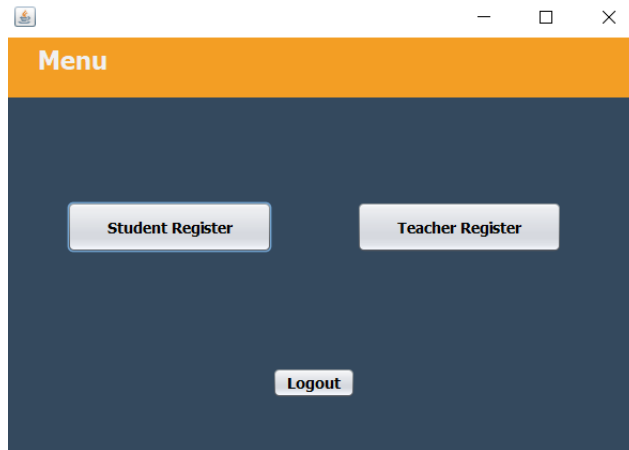
ขั้นตอนการทำงานของโปรแกรม

1. login เมื่อเปิดโปรแกรมมา จะมีหน้าต่าง login โดยผู้ใช้งานต้องเลือกสถานะของตัวเองก่อน โดยแบ่งออกเป็น 3 กลุ่ม คือ admin, อาจารย์ และ นักเรียน

A screenshot of a web application login window. The window has a title bar with a small icon and the text "Login". The main content area has a dark blue background. At the top, there is an orange header bar with the word "Login" in white. Below the header, there are three input fields: "Username:" with a text input, "Password:" with a text input, and "Division:" with a dropdown menu showing "Student". At the bottom, there is a "Submit" button.

2 ส่วนของหน้าต่างการใช้งาน

2.1) Admin หลังจากlogin เข้ามาแล้วโดยเลือกสถานะ admin โดยจะมีหน้าที่ในการสร้าง ID และPass ให้กับนักเรียนและอาจารย์ โดยจะทำการเลือกว่าจะใส่ข้อมูลให้นักเรียนหรืออาจารย์ดังนี้



หลังจากนั้นเพิ่มข้อมูลของนักเรียนและอาจารย์ไปยังฐานข้อมูล ตาม UI ดังนี้

A screenshot of a web application window titled "Register". The window has a dark blue background with an orange header bar. The form contains the following fields: ID, Password, Sex, First Name, Last Name, Citizen Identification, Date Of Birth, Mobile Phone, Address (with a text area), and class. At the bottom, there are two buttons: "Cancel" (red) and "Submit" (blue).A screenshot of a web application window titled "Register". The window has a dark blue background with an orange header bar. The form contains the following fields: ID, Password, Subject, First Name, Last Name, E-mail, Date Of Birth, and Mobile Phone. At the bottom, there are two buttons: "Cancel" (red) and "Submit" (blue).

2.2)อาจารย์ หลังจากlogin จะเข้าสู่หน้าของผู้ใช้งาน โดยหน้าแรกจะเป็นข้อมูลของอาจารย์

Personal 3/1 3/2 3/3

Personal Detail

ID: 55555 Password: password

Firstname: somkrit Lastname: suza

Email: aaaa@gmail.com

Subject: math

ส่วนหน้าที่เหลือจะเป็นส่วนของห้องเรียนที่ตนเองสอนในรายวิชานั้น โดยจะเห็นข้อมูลของทุกคนในห้องนั้นและสามารถให้คะแนนนักเรียนได้ตามขั้นตอนต่อไปนี้

1. นำเมาส์ไปคลิกยังนักเรียนที่ต้องการจะเพิ่มคะแนน

Personal 3/1 3/2 3/3

Class: 3/1 Subject : Math

ID : 60070666

Name : Kacaya

Lastname : komutbu

Score : 0

Update score :

id	ชื่อ	นามสกุล	คะแนน
60070666	Kacaya	komutbu	0
60070555	pitawas	choo	0
60070444	pimtapat	yoo	0

2. ข้อมูลของนักเรียนคนนั้นจะมาอยู่ทางด้านซ้าย

Class: 3/1 Subject : Math

ID :

Name :

Lastname :

Score :

Update score :

รหัส	ชื่อ	นามสกุล	คะแนน
60070666	Kazuya	komutzu	0
60070555	pittawas	choo	0
60070444	pimlapat	yoo	0

3. กรอกคะแนนที่ต้องการเพิ่มตามช่องตามภาพ(หากต้องการลบคะแนนให้เครื่องหมาย-ตามด้วยคะแนน เช่น -5)

Class: 3/1 Subject : Math

ID :

Name :

Lastname :

Score :

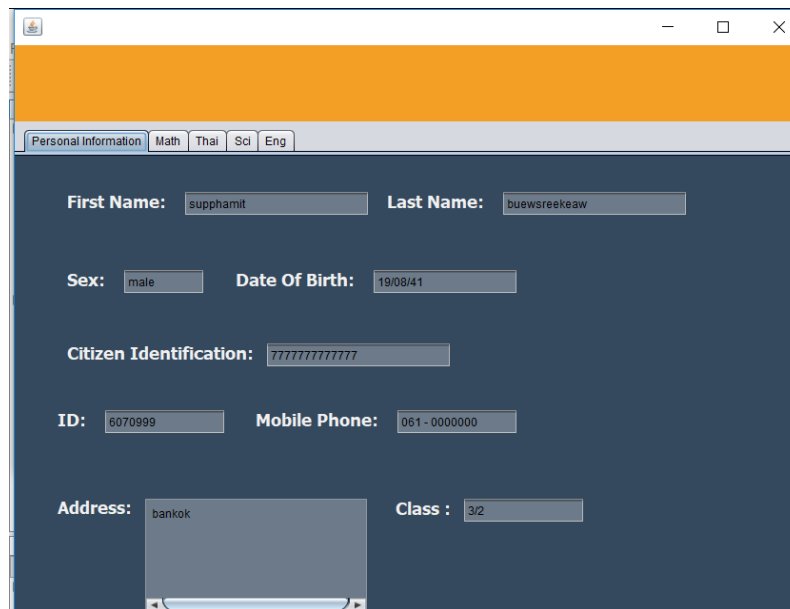
Update score :

รหัส	ชื่อ	นามสกุล	คะแนน
60070666	Kazuya	komutzu	30
60070555	pittawas	choo	0
60070444	pimlapat	yoo	0

Message

Data updated Successfully

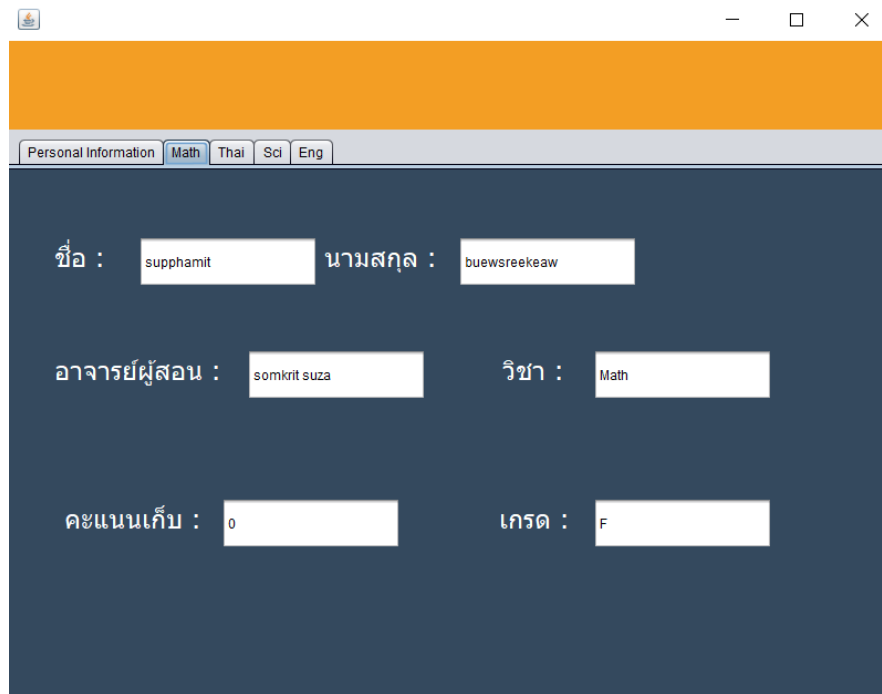
2.3)นักเรียน หลังจาก login เข้ามาหน้าแรกจะเป็นข้อมูลส่วนตัวของนักเรียน



A screenshot of a web application window. The window has a title bar with a minimize, maximize, and close button. Below the title bar is a navigation bar with tabs: "Personal Information", "Math", "Thai", "Sci", and "Eng". The "Personal Information" tab is selected. The main content area is a form with the following fields:

First Name:	supphamit	Last Name:	buewsreekeaw	
Sex:	male	Date Of Birth:	19/08/41	
Citizen Identification:	777777777777			
ID:	6070999	Mobile Phone:	061-0000000	
Address:	bankok		Class :	3/2

ส่วนหน้าถัดไปจะเป็นรายวิชาที่ตนเองเรียน จะมีชื่อ ของอาจารย์ ผู้สอน วิชาที่เรียน คะแนนเก็บ เกรดที่คำนวณ ณ ปัจจุบัน



A screenshot of a web application window. The window has a title bar with a minimize, maximize, and close button. Below the title bar is a navigation bar with tabs: "Personal Information", "Math", "Thai", "Sci", and "Eng". The "Math" tab is selected. The main content area is a form with the following fields:

ชื่อ :	supphamit	นามสกุล :	buewsreekeaw
อาจารย์ผู้สอน :	somkrit suza	วิชา :	Math
คะแนนเก็บ :	0	เกรด :	F

การสร้าง Database

โดยจะสร้าง database แยกออกเป็น แอดมิน, นักเรียนและจารย์

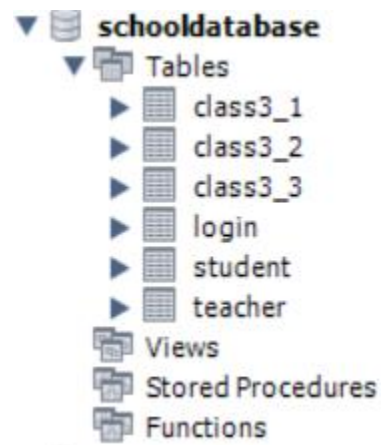
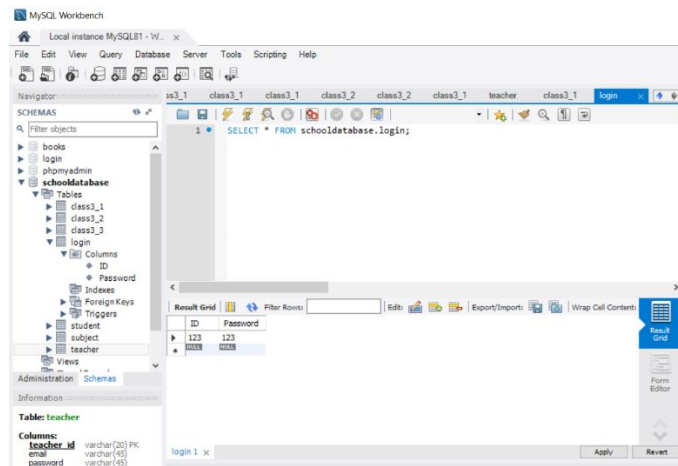


Table admin จะประกอบด้วยcolumnดังนี้

-ID

-Password



โดยนักเรียนจะมี 2 table คือ

เก็บข้อมูลของนักเรียนทั้งโรงเรียน โดย column จะประกอบด้วย

- ID
- Password
- FName
- LName
- CitizenID
- Address
- Mobilephone
- DateofBirth
- Class

MySQL Workbench

Local instance MySQL81 - W...

File Edit View Query Database Server Tools Scripting Help

Navigator: ss3_1 class3_1 class3_2 class3_2 class3_1 teacher class3_1 login student

SCHEMAS

Filter objects

- class3_2
- class3_3
- login
- student
 - Columns
 - ID
 - Password
 - FName
 - LName
 - CitizenID
 - Sex
 - Address
 - MobilePhone
 - DateofBirth
 - Class
 - Indexes
 - Foreign Keys
 - Triggers
- teacher

Administration Schemas

Information

Table: student

Columns:
ID varchar(10) PK
Password varchar(50)

1 • SELECT * FROM schooldatabase.student;

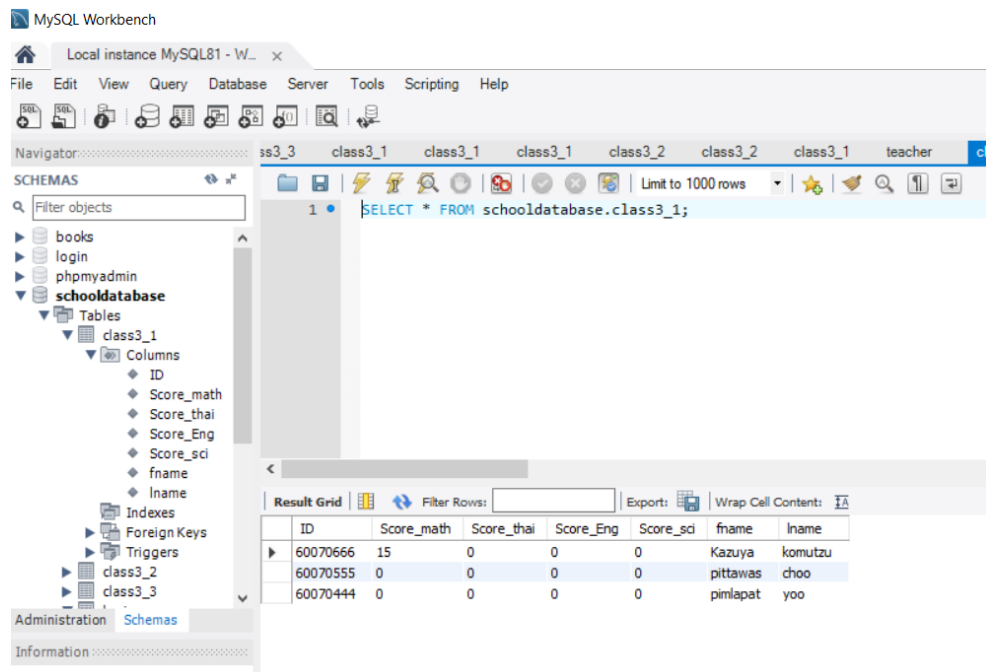
Result Grid

ID	Password	FName	LName	CitizenID	Sex	Address	MobilePhone
60070111	999	first	last	1795465214453	female	LKB BKK	0989542114
60070222	999	firstname	lastname	1200154875641	male	acbdefg	0954756214
60070333	999	narapitch	sea	1200154784562	male	acbdef	084587441
60070444	999	pimlapat	yoo	7954315452145	female	acbde	084562541
60070555	999	pittawas	choo	7854621547851	female	acbd	0324595654
60070666	999	Kazuza	komutzu	4236879548756	male	LKB	0245987565
6070777	password	rattana	jumpasa	5555555555555	male	bankok ladkrabank	0974561234
6070888	password	alison	timsri	4444444444444	male	bankok ladkrabank	087654456

student 1 x Apply Revert

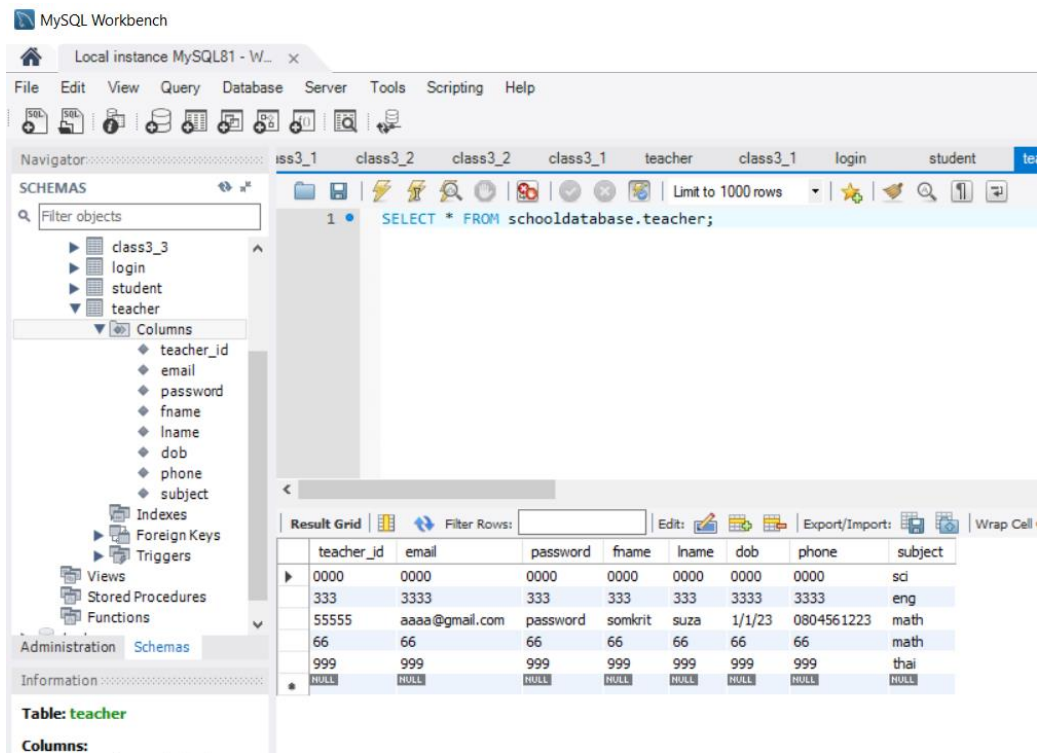
และเก็บข้อมูลของนักเรียนเป็นห้อง โดยcolumnจะประกอบด้วย

- ID
- Score_math
- Score_thai
- Score_Eng
- Score_sci
- fname
- lname



อาจารย์ ประกอบด้วย column

- teacher_id
- email
- password
- fname
- lname
- dob
- phone
- subject



ตัวอย่าง code การนำข้อมูลนักเรียนจากฐานข้อมูล มาแสดงบน UI ของอาจารย์

```
public ArrayList<AddScore> getUsersList3_2()
{
    ArrayList<AddScore> usersList = new ArrayList<AddScore>();
    Connection connection = getConnection();

    String query = "SELECT * FROM class3_2";
    Statement st;
    ResultSet rs;

    try {
        st = connection.createStatement();
        rs = st.executeQuery(query);

        AddScore user;

        while(rs.next())
        {
            user = new AddScore(rs.getInt("ID"),rs.getString("fname"),rs.getString("lname"),rs.getInt("Score_math"));
            usersList.add(user);
        }

    }
    catch (Exception e) {
        e.printStackTrace();
    }
    return usersList;
}
```

```

public void Show_Student_math_In_Table3_2(){
    ArrayList<AddScore> list = getUsersList3_2();
    DefaultTableModel model = (DefaultTableModel)jTable3.getModel();
    Object[] row = new Object[4];
    for (int i = 0; i < list.size() ; i++) {
        row[0] = list.get(i).getId();
        row[1] = list.get(i).getFirstName();
        row[2] = list.get(i).getLastName();
        row[3] = list.get(i).getScore();
        model.addRow(row);
    }

}

public void executeSQLQuery3_2(String query, String message)
{
    Connection con = getConnection();
    Statement st;
    try{
        st = con.createStatement();
        if((st.executeUpdate(query)) == 1)
        {
            // refresh jtable data
            DefaultTableModel model = (DefaultTableModel)jTable3.getModel();
            model.setRowCount(0);
            Show_Student_math_In_Table3_2();

            JOptionPane.showMessageDialog(null, "Data "+message+" Succesfully");
        }else{
            JOptionPane.showMessageDialog(null, "Data Not "+message);
        }
    }catch(Exception ex){
        ex.printStackTrace();
    }
}

```