



# Database

Ths. Trần Tuấn Dũng – [dungtran@uit.edu.vn](mailto:dungtran@uit.edu.vn)



# Relational vs Non – Relational

	Relational Database	Non-Relational Database
<b>Data Integrity</b>	Enforces data integrity through relationships and constraints	Flexible schema allows for easy data updates
<b>Structured Queries</b>	Supports complex queries using SQL	May offer simpler, faster query performance
<b>Scalability</b>	May not scale as well for large datasets	Highly scalable and can handle big data and high traffic
<b>ACID Transactions</b>	Provides ACID (Atomicity, Consistency, Isolation, Durability) transactions	May sacrifice ACID properties for improved performance or scalability
<b>Consistency</b>	Ensures strong data consistency	May offer eventual consistency or tunable consistency
<b>Schema Evolution</b>	Schema changes can be complex and time-consuming	Allows for easy schema changes and flexibility in data model
<b>Examples</b>	MySQL, PostgreSQL, Oracle	MongoDB, Cassandra, Redis, DynamoDB



# Relational Database

This lecture assume you already pass IT004 and have knowledge of SQL Queries

See [SQL Tutorial \(w3schools.com\)](http://w3schools.com) if you need revision

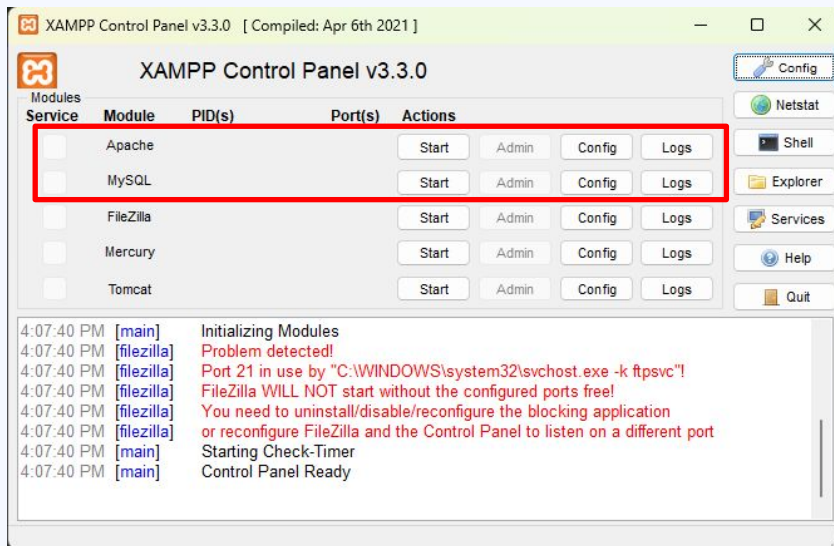


# MySQL

This course will use MySQL as Database Server and PHPMyadmin as DBMS

Install Xampp: [XAMPP Installers and Downloads for Apache Friends](#)

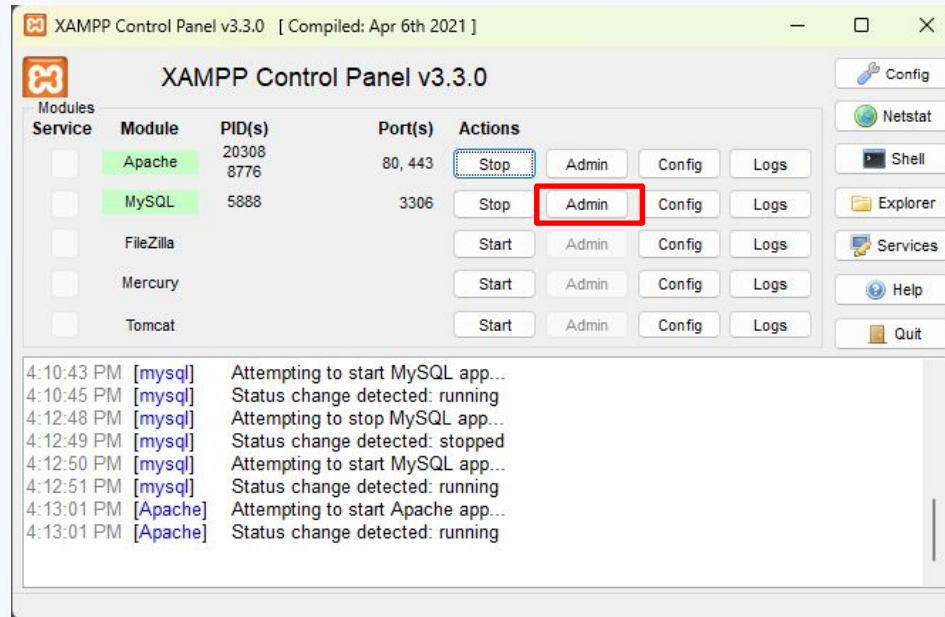
Start both PHP and MySQL Server and you're ready to go





# MySQL

Click on Admin to open phpMyAdmin DBMS site on your local machine





# MySQL

## Trường Đại học Công nghệ Thông tin – Đại học Quốc gia Hồ Chí Minh Bộ môn An toàn thông tin



The screenshot displays the phpMyAdmin web interface for a MySQL server. The top navigation bar includes tabs for Databases, SQL, Status, User accounts, Export, Import, Settings, Replication, Variables, Charsets, Engines, and Plugins. The left sidebar shows a tree view of databases, including information\_schema, mysql, performance\_schema, phpmyadmin, pv, test, and wordpress\_enhance\_ui. The main content area is divided into several sections:

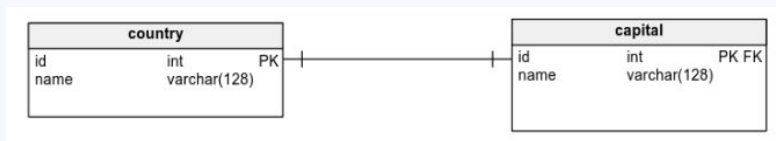
- General settings:** Shows the server connection collation set to 'utf8mb4\_unicode\_ci' and a link to 'More settings'.
- Appearance settings:** Shows the language set to 'English' and the theme set to 'pmahomme'.
- Database server:** Lists server details: Server: 127.0.0.1 via TCP/IP, Server type: MariaDB, Server connection: SSL is not being used, Server version: 10.4.27-MariaDB - mariadb.org binary distribution, Protocol version: 10, User: root@localhost, and Server charset: UTF-8 Unicode (utf8mb4).
- Web server:** Lists web server details: Apache/2.4.54 (Win64) OpenSSL/1.1.1p PHP/8.2.0, Database client version: libmysql - mysqlnd 8.2.0, PHP extension: mysqli, curl, mbstring, and PHP version: 8.2.0.
- phpMyAdmin:** Lists version information: Version information: 5.2.0, latest stable version: 5.2.1, and links to Documentation, Official Homepage, Contribute, Get support, List of changes, and License.

A notification at the bottom states: "A newer version of phpMyAdmin is available and you should consider upgrading. The newest version is 5.2.1, released on 2023-02-08."



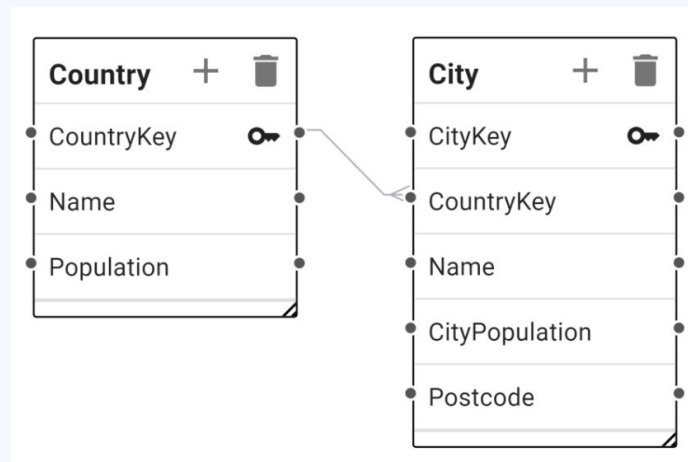
# 1 – 1, 1 – n, n – n Relationship

Primary key ref Primary key



1 – 1 relationship

Foreign key ref Primary key

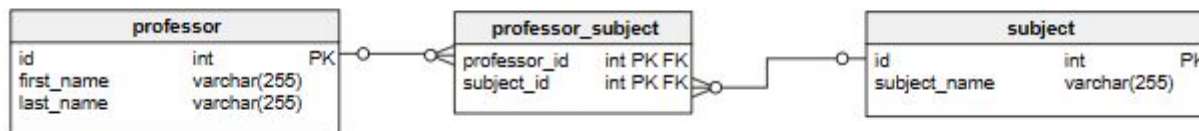


1 – n relationship



# 1 – 1, 1 – n, n – n Relationship

Use intermediate table



n – n relationship





# Object Relational Mapping (ORM)

Instead of writing raw queries, we can map a **table** with a **model**

Most frameworks now support ORM, allow we to interact with database via Object

ORM provide simpler syntax, more readable then raw queries



## Example: PHP Laravel

```
class User extends Model {  
  
    protected $table = 'my_users';  
  
}
```

Laravel already cover all the setup we need in the **Model** class

Now we can interact with `my\_users` table via **User** class

Eloquent ORM - Laravel 5.0 - The PHP Framework For Web Artisans



## Example: PHP Laravel

```
$users = User::all(); // get all records  
$user = User::find(1); //find record with primary  
key value is 1  
$model = User::where('votes', '>',  
100)->firstOrFail(); // complex queries  
$user->save(); // update
```

[Overview of Entity Framework Core - EF Core | Microsoft Learn](#)

[Sequelize | Feature-rich ORM for modern TypeScript & JavaScript](#)

[Eloquent ORM - Laravel 5.0 - The PHP Framework For Web Artisans](#)



# Non – Relational Database (NoSQL Database)

Storing data as **Collections** and **Documents** (just like **tables** and **records**)

More Flexible Schema than Relational Database

Syntax similar to ORM



# MongoDB

Use Free Service to get start MongoDB Atlas | MongoDB

**Database Deployments**

Find a database deployment...

Edit Config + Create

Cluster0 Connect View Monitoring Browse Collections ... FREE SHARED

Your cluster has been automatically paused due to prolonged inactivity.  
Resume your cluster to connect to it and to gain access to your data.

Resume

VERSION	REGION	CLUSTER TIER	TYPE	BACKUPS	LINKED APP SERVICES	ATLAS SEARCH
7.0.6	AWS / Singapore (ap-southeast-1)	M0 Sandbox (General)	Replica Set - 3 nodes	Inactive	None Linked	Create Index

+ Add Tag

New On Atlas 2

Goto



# MongoDB

Get Connection string and connect with nodejs

[Node.js MongoDB Create Database \(w3schools.com\)](#)

## Connecting with MongoDB Driver

### 1. Select your driver and version

We recommend installing and using the latest driver version.

Driver

Version

Node.js

5.5 or later

### 2. Install your driver

Run the following on the command line

```
npm install mongodb
```

[View MongoDB Node.js Driver installation instructions.](#)

### 3. Add your connection string into your application code

☐ View full code sample

```
mongodb+srv://fkmddev:<password>@cluster0.vnqljo6.mongodb.net/?  
retryWrites=true&w=majority&appName=Cluster0
```



# Firestore

[Get started with Cloud Firestore | Firebase \(google.com\)](#)