

Phát Triển Phần Mềm Mã Nguồn Mở

LẬP TRÌNH PHP & MYSQL

LÊ THỊ BÍCH HẰNG
NGUYỄN HẢI TRIỀU

Bộ môn Kỹ thuật phần mềm
Khoa Công nghệ thông tin
ĐH Nha Trang

Mục lục

- 1 Tổng quan về MySQL
- 2 Thao tác CSDL trong PHP

- 1 Tổng quan về MySQL
- 2 Thao tác CSDL trong PHP

MySQL là một hệ thống quản trị cơ sở dữ liệu mã nguồn mở (Relational Database Management System, viết tắt là **RDBMS**) hoạt động theo mô hình **client-server**.

- MySQL là một trong số các phần mềm RDBMS
- MySQL là CSDL không những chứa dữ liệu mà nó còn cho phép người dùng **tìm kiếm, xử lý, sắp xếp** ... dữ liệu
- MySQL hỗ trợ nhiều người dùng, với nhiều quyền khác nhau
- MySQL sử dụng các **truy vấn theo chuẩn SQL**

SQL (Structured Query Language) là ngôn ngữ **truy vấn có cấu trúc** chung. MySQL, PostgreSQL, Microsoft SQL server ... đều **sử dụng syntax của SQL**.

Cách vận hành chính trong môi trường MySQL:

- MySQL **tạo ra bảng** để lưu trữ dữ liệu, **định nghĩa sự liên quan** giữa các bảng đó.
- Client sẽ gửi yêu cầu SQL bằng một lệnh đặc biệt trên MySQL.
- Ứng dụng trên server sẽ phản hồi thông tin và trả về kết quả trên máy client.

Một vài định nghĩa trong SQL

- **NULL/NOT NULL**: Giá trị cho phép được **rỗng/không rỗng**
- **AUTO_INCREMENT**: Cho phép giá trị **tăng dần (tự động)**
- **UNSIGNED** : Phải là **số nguyên dương**
- **PRIMARY KEY**: Cho phép nó là **khóa chính** trong bảng
- **FOREIGN KEY**: Cho phép nó là **khóa ngoại** trong bảng, dùng để **tham chiếu các bảng với nhau** thông qua **PRIMARY KEY**.

Trong **MySQL** chúng ta có 3 kiểu dữ liệu chính: **chuỗi**, **số học**, **ngày giờ**. Các kiểu dữ liệu chi tiết:

Kiểu dữ liệu	Mô tả
char (length)	Chuỗi , tối đa 255 ký tự, chiều dài cố định = length
varchar (length)	Chuỗi , tối đa 255 ký tự, chiều dài động ≤ length
text	Chuỗi , tối đa 65536 ký tự
int (length)	Số nguyên , từ -2.147.483.648 đến +2.147.483.647
decimal (length,dec)	Số thập phân, tối đa length chữ số trong đó có dec chữ số thập phân
float	Số thập phân có chiều dài nhỏ
double	Số thập phân có chiều dài lớn
date	kiểu dữ liệu ngày theo format: YYYY-MM-DD
datetime	kết hợp ngày và giờ theo format: YYYY-MM-DD hh:mm:ss
enum ("option1", , "option2",...)	Tập hợp tự định nghĩa, tối đa 65.535 giá trị

Lệnh	Mô tả
CREATE	tạo CSDL hoặc bảng
ALTER	thay đổi cấu trúc bảng có sẵn
SELECT	chọn dữ liệu từ bảng
DELETE	xóa dữ liệu khỏi bảng
DESCRIBE	xem thông tin mô tả về cấu trúc bảng
INSERT INTO	ghi giá trị vào bảng
UPDATE	cập nhật dữ liệu đã có trong bảng
DROP	xóa bảng hay toàn bộ CSDL
WHERE	giới hạn phạm vi tìm kiếm
JOIN	Nối 2 bảng hay nhiều bảng với nhau để có một thông tin đầy đủ

Tạo CSDL

Cú pháp

```
CREATE DATABASE databasename;
```

Ví dụ

```
CREATE DATABASE testDB  
DEFAULT CHARACTER SET utf8  
COLLATE utf8_general_ci;
```


Tạo Bảng

Cú pháp

```
CREATE TABLE table_name (  
    column1 datatype [NULL|NOT NULL] [CONSTRAINTS],  
    column2 datatype [NULL|NOT NULL] [CONSTRAINTS],  
    column3 datatype [NULL|NOT NULL] [CONSTRAINTS],  
    ...  
);
```

Ví dụ

```
CREATE TABLE Persons (  
    PersonID int not null primary key,  
    LastName varchar(255),  
    FirstName varchar(255),  
    Address varchar(255),  
    City varchar(255)  
);
```

Lệnh Create

- Tạo indexes trong bảng

```
CREATE INDEX index_name  
ON table_name (column1, column2, ...);
```

- Tạo procedure

```
CREATE PROCEDURE procedure_name([parameters]) BEGIN ... END;
```

- Tạo user

```
CREATE USER user_name[@hostname] [IDENTIFIED BY [PASSWORD]  
'password'];
```

- Tạo view

```
CREATE [OR REPLACE] VIEW view_name AS SELECT ...;
```

Câu lệnh **ALTER TABLE** được sử dụng để **thêm, xóa hoặc sửa đổi các cột trong bảng** hiện có. Ngoài ra nó cũng được sử dụng để **thêm và xóa các CONSTRAINTS** trên một bảng hiện có.

Cú pháp

```
ALTER TABLE table_name(  
  ADD column datatype [NULL | NOT NULL] [CONSTRAINTS],  
  CHANGE column columns datatype [ NULL | NOT NULL]  
  [CONSTRAINTS], DROP column, ...  
);
```

Ví dụ

```
ALTER TABLE Customers  
ADD Email varchar(255) NOT NULL;
```

Lệnh **SELECT** được sử dụng để chọn data từ database. Cú pháp

```
SELECT column_name, ...  
FROM table_name, ...  
[WHERE ...]  
[UNION ...]  
[GROUP BY ...]  
[HAVING ...]  
[ORDER BY ...];
```

Trong đó

- **WHERE**: giới hạn phạm vi tìm kiếm
- **UNION**: toán tử này được sử dụng để kết hợp tập kết quả của hai hay nhiều câu lệnh WHERE
- **ORDER BY**: là từ khóa được sử dụng để sắp xếp tập kết quả theo thứ tự tăng dần hoặc giảm dần.

Ví dụ

```
SELECT * FROM Customers  
WHERE CustomerID=1;
```

Kết quả

CustomerID	CustomerName	ContactName	Address	City	PostalCode	Country
1	Alfreds Futterkiste	Maria Anders	Obere Str. 57	Berlin	12209	Germany

Hình 1: kết quả cho điều kiện **WHERE CustomerID=1;**

CustomerID	CustomerName	ContactName	Address	City	PostalCode	Country
1	Alfreds Futterkiste	Maria Anders	Obere Str. 57	Berlin	12209	Germany
2	Ana Trujillo Emparedados y helados	Ana Trujillo	Avda. de la Constitución 2222	México D.F.	05021	Mexico
3	Antonio Moreno Taquería	Antonio Moreno	Mataderos 2312	México D.F.	05023	Mexico
4	Around the Horn	Thomas Hardy	120 Hanover Sq.	London	WA1 1DP	UK
5	Berglunds snabbköp	Christina Berglund	Berguvsvägen 8	Luleå	S-958 22	Sweden

Hình 2: Bảng customers **WHERE CustomerID<=5;**(https://www.w3schools.com/sql/trysql.asp?filename=trysql_select_where)

Lệnh **DELETE** được dùng để xóa dữ liệu đã được ghi trong bảng.

```
DELETE FROM table_name
[WHERE ...];
```

Ví dụ

```
DELETE FROM Customers
WHERE CustomerName='Alfreds Futterkiste';
```

CustomerID	CustomerName	ContactName	Address	City	PostalCode	Country
1	Alfreds Futterkiste	Maria Anders	Obere Str. 57	Berlin	12209	Germany
2	Ana Trujillo Emparedados y helados	Ana Trujillo	Avda. de la Constitución 2222	México D.F.	05021	Mexico
3	Antonio Moreno Taquería	Antonio Moreno	Mataderos 2312	México D.F.	05023	Mexico

CustomerID	CustomerName	ContactName	Address
2	Ana Trujillo Emparedados y helados	Ana Trujillo	Avda. de la Constitución 2222
3	Antonio Moreno Taquería	Antonio Moreno	Mataderos 2312

Để xóa dữ liệu của tất cả các hàng trong bảng mà không phải xóa bảng, ta sử dụng

```
DELETE FROM table_name;
```

Lệnh **INSERT INTO** dùng để ghi giá trị mới vào bảng. Cú pháp

```
INSERT INTO table_name (column1, column2, column3, ...)
VALUES (value1, value2, value3, ...);
```

Ví dụ:

```
INSERT INTO Customers (CustomerName, ContactName, Address,
City, PostalCode, Country)
VALUES ('Cardinal', 'Tom B. Erichsen', 'Skagen 21',
'Stavanger', '4006', 'Norway');
```

CustomerID	CustomerName	ContactName	Address	City	PostalCode	Country
89	White Clover Markets	Karl Jablonski	305 - 14th Ave. S. Suite 3B	Seattle	98128	USA
90	Wilman Kala	Matti Karttunen	Keskuskatu 45	Helsinki	21240	Finland
91	Wolski	Zbyszek	ul. Filtrowa 68	Walla	01-012	Poland

CustomerID	CustomerName	ContactName	Address	City
89	White Clover Markets	Karl Jablonski	305 - 14th Ave. S. Suite 3B	Seattle
90	Wilman Kala	Matti Karttunen	Keskuskatu 45	Helsinki
91	Wolski	Zbyszek	ul. Filtrowa 68	Walla
92	Cardinal	Tom B. Erichsen	Skagen 21	Stavanger

Lệnh **UPDATE** dùng để sửa đổi các bản ghi hiện có trong bảng. Cú pháp

```
UPDATE table_name
SET column1 = value1, column2 = value2, ...
WHERE condition;
```

Ví dụ

```
UPDATE Customers
SET ContactName = 'Alfred_Schmidt', City= 'Frankfurt'
WHERE CustomerID = 1;
```


Xóa CSDL, sử dụng cú pháp sau

```
DROP DATABASE databasename;
```

Áp dụng xóa tương tự đối với các thành phần khác như bảng, chỉ mục trigger

```
DROP INDEX | PROCEDURE | TABLE | TRIGGER |  
USER | VIEW itemname;
```

Ví dụ

```
DROP DATABASE testDB;  
DROP TABLE Customers;  
USER | VIEW itemname;
```

Để xóa quan hệ giữa các bảng (**foreign keys**) chúng ta có thể sử dụng các lệnh sau

```
ALTER TABLE yourtable DROP CONSTRAINT yourforeignkeyname  
  
SET FOREIGN_KEY_CHECKS=0; DROP TABLE yourtable;
```

Khóa Chính

- Khóa chính **PRIMARY KEY** là **duy nhất** trong một bảng và không được **NULL**
- Mỗi bảng chỉ có được **một khóa chính**

Ví dụ

```
CREATE TABLE Persons (  
    ID int NOT NULL AUTO_INCREMENT,  
    LastName varchar(255) NOT NULL,  
    FirstName varchar(255),  
    Age int,  
    PRIMARY KEY (ID)  
);
```

Để loại bỏ khóa chính, chúng ta có thể sử dụng các lệnh:

- Đối với MySQL

```
ALTER TABLE Persons  
DROP PRIMARY KEY;
```

- Đối với SQL Server / Oracle / MS Access:

```
ALTER TABLE Persons  
DROP CONSTRAINT PK_Person;
```

- Khóa ngoại **FOREIGN KEY** dùng để liên kết hai bảng lại với nhau
- là một cột hoặc tập hợp các cột liên kết đến khóa chính **PRIMARY KEY** của bảng khác
- bảng chứa **FOREIGN KEY** được gọi là bảng con (child table)

Ví dụ

```
CREATE TABLE Persons (  
    ID int NOT NULL,  
    LastName varchar(255) NOT NULL,  
    FirstName varchar(255),  
    Age int,  
    PRIMARY KEY (ID)  
);  
CREATE TABLE Address (  
    addressID int NOT NULL,  
    street varchar(254) NOT NULL,  
    PersonID int,  
    FOREIGN KEY (PersonID) REFERENCES Persons(ID)  
);
```

DB cho Flight Search & Booking

Flight Search & Booking là database mô phỏng dữ liệu của tìm kiếm chuyến bay và đặt vé. Mô tả danh sách bảng như sau:

Tên bảng	Ý nghĩa
countries	bảng lưu trữ định danh quốc gia. (nơi đến và nơi đi)
city	lưu trữ thành phố, bang/tỉnh
travel_class	hạng vé đặt: Economy or Business
airline	lưu trữ thông tin hãng hàng không
airplane	lưu trữ các thông tin về máy bay
airplane_capacity	mô tả các thông tin về sức chứa của máy bay
flight_schedule	các thông tin tìm kiếm vé máy bay
seats_booked	thông tin vị trí đã đặt
user	thông tin người tìm kiếm vé máy bay
Booking	thông tin đặt vé của khách hàng

Cấu trúc của bảng được biểu diễn như sau:

Field	DataType	Description	Constraints
id	INT	id của quốc gia	Primary Key
name	Varchar(25)	tên của quốc gia	NOT NULL

Bảng 1: countries

```
create table countries(  
    id int primary key AUTO_INCREMENT,  
    name varchar(25) not null  
);  
  
insert into countries values(1, 'VietNam');  
insert into countries values(2, 'Japan');
```

Field	Data Type	Description	Constraints
id	INT	id của thành phố	Primary Key
name	Varchar(25)	tên của thành phố	NOT NULL
state_name	Varchar(25)	tên của bang/tỉnh	
country	INT	PK từ countries table	Foreign Key

Bảng 2: city

```

create table city(
    id int primary key AUTO_INCREMENT,
    name varchar(25) not null,
    state varchar(25) not null,
    country int,
    constraint _FK_city_countries foreign key (country)
        references countries(id)
);
-- thanh pho: 1. id; 2. ten; 3. bang/ tinh
insert into city values(1, 'Nha_Trang', 'Khanh_Hoa', 1);
insert into city values(2, 'Tokyo', 'Tokyo', 2);

```

Thực hiện tương tự cho các bảng sau

Field	Data Type	Description	Constraints
id	INT	id của hạng vé	Primary Key
name	Varchar(10)	tên của hạng vé	NOT NULL

Bảng 3: travel_class

Field	Data Type	Description	Constraints
id	INT	id của hãng hàng không	Primary Key
name	varchar(30)	tên đầy đủ của hãng hàng không	
short_name	char(2)	ký hiệu hãng hàng không	NOT NULL
logo	BLOB	logo của hãng hàng không	NOT NULL

Bảng 4: airline

Field	Data Type	Description	Constraints
id	INT	id của máy bay	Primary Key
airline_id	INT	PK từ airline table	Foreign Key

Bảng 5: airplane

Field	Data Type	Description	Constraints
airplane_id	INT	PK từ airplane table	Foreign Key
class_id	INT	PK từ travel_class table	Foreign Key
capacity	INT	sức chứa của máy bay	NOT NULL

Bảng 6: airplane_capacity

Field	Data Type	Description	Constraints
id	INT	id duy nhất để xác định chuyến bay cần tìm	Primary Key
fnumber	INT	số hiệu chuyến bay	NOT NULL
airplane_id	INT	PK từ airplane table	Foreign Key
d_city	INT	thành phố khởi hành, PK từ city table	Foreign Key
d_date_time	datetime	giờ khởi hành	NOT NULL
a_city	INT	thành phố đến, PK từ city table	Foreign Key
a_date_time	datetime	giờ đến	NOT NULL
price	numeric(8,2)	giá vé	NOT NULL
free_meals	BOOLEAN	phục vụ ăn miễn phí	NOT NULL
refundable	BOOLEAN	khả năng hoàn trả vé	NOT NULL

Bảng 7: flight_schedule

Field	Data Type	Description	Constraints
flight_id	INT	PK từ flight_schedule table	Foreign Key
class_id	INT	PL từ travel_class table	Foreign Key
seats_booked	INT	ghế đã đặt	NOT NULL

Bảng 8: seats_booked

Field	Data Type	Description	Constraints
id	INT	id cho booking	Primary Key
booking_time	datetime	lưu thời gian booking	NOT NULL
flight_id	INT	PK từ flight_schedule table	Foreign Key
user_id	INT	PK từ user table	Foreign Key
status	char(3)	trạng thái book phòng	NOT NULL

Bảng 9: Booking

Field	Data Type	Description	Constraints
id	INT	id của user	Primary Key
password_hash	char(32)	mật khẩu đăng nhập	
fname	varchar(20)	tên đầy đủ	NOT NULL
mname	varchar(20)	tên đệm	
lname	varchar(20)	tên	NOT NULL
sex	char(1)	giới tính	NOT NULL
address	varchar(50)	địa chỉ của user	NOT NULL
email	varchar(30)	email của user	NOT NULL

Bảng 10: user

```
create table travel_class(  
    id int not null primary key AUTO_INCREMENT,  
    name varchar(10) not null, UNIQUE (name)  
);  
  
insert into travel_class (name) values('Economy');  
insert into travel_class (name) values('Business');  
  
create table airline(  
    id int PRIMARY KEY AUTO_INCREMENT ,  
    name varchar(30) NOT NULL,  
    short_name char(2) NOT NULL,  
    logo BLOB  
);  
  
Create table airplane (  
    id int primary key AUTO_INCREMENT,  
    airline_id int not null,  
    CONSTRAINT _FK_airplane_airline foreign key(  
        airline_id)  
        references airline(id)  
);
```

```
create table airplane_capacity(  
    airplane_id int not null,  
    class_id int not null,  
    capacity int not null,  
    CONSTRAINT _FK_airplane_capacity_airplane  
        foreign key(airplane_id) references airplane(id),  
    CONSTRAINT _FK_airplane_capacity_travel_class  
        foreign key(class_id) references travel_class(id)  
);  
Create table flight(  
    id int primary key AUTO_INCREMENT,  
    fnumber int not null,  
    airplane_id int not null,  
    d_city int not null,  
    d_date_time datetime not null,  
    a_city int not null,  
    a_date_time datetime not null,  
    price numeric(8,2) not null,  
    free_meals BOOLEAN not null,  
    refundable BOOLEAN not null,  
    CONSTRAINT _FK_flight_airplane  
        Foreign key(airplane_id) REFERENCES airplane(id),  
    CONSTRAINT _FK_flight_city_dcity  
        Foreign key(d_city) references city(id),  
    CONSTRAINT _FK_flight_city_acity
```

```
create table seats_booked(  
    flight_id int not null,  
    class_id int not null,  
    seats_booked int not null,  
    CONSTRAINT _FK_seats_booked_travel_class  
        Foreign key(class_id) references travel_class(id),  
    CONSTRAINT _FK_seats_booked_flight  
        Foreign key(flight_id) references flight(id),  
    PRIMARY KEY(flight_id,class_id)  
);  
  
CREATE TABLE user(  
    id int(11) NOT NULL AUTO_INCREMENT,  
    password_hash char(32) NOT NULL,  
    fname varchar(20) NOT NULL,  
    mname varchar(10) DEFAULT NULL,  
    lname varchar(10) NOT NULL,  
    sex char(1) DEFAULT NULL,  
    address varchar(50) DEFAULT NULL,  
    email varchar(30) DEFAULT NULL,  
    PRIMARY KEY (id),  
    CONSTRAINT check_sex check(sex='M' OR sex='F')  
);
```

```
Create table Booking(  
    id int primary key AUTO_INCREMENT,  
    booking_time datetime DEFAULT CURRENT_TIMESTAMP,  
    flight_id int not null,  
    user_id int not null,  
    noOfTravellers int not null,  
    class_id int not null,  
    status char(3) DEFAULT NULL,  
    CONSTRAINT _FK_booking_user  
        Foreign key(user_id) references user(id),  
    CONSTRAINT _FK_booking_flight  
        Foreign key(flight_id) references flight(id),  
    CONSTRAINT _FK_booking_travel_class  
        Foreign Key(class_id) references travel_class(id)  
);
```

- 1 Tổng quan về MySQL
- 2 Thao tác CSDL trong PHP

- Kết nối mysql server

```
mysql [-h hostname] [-P portnumber] -u username -p  
mysql [-h hostname] [-P portnumber] --user=user  
--password=pass
```

- Nhập lệnh sau dấu nhắc lệnh `mysql>`
- Mỗi lệnh SQL kết thúc bằng dấu `;`

Sử dụng **XAMPP** hoặc **LAMP** stack, truy cập vào địa chỉ <http://localhost/phpmyadmin>. Phiên bản phpMyAdmin đang sử dụng trong slide là **4.9.5deb2**

The screenshot displays the phpMyAdmin web interface in a browser window. The address bar shows the URL `localhost/phpmyadmin/index.php?db=&table=&token=334963296163602c4d6941592c3e347a&lang=en`. The interface includes a sidebar with a tree view of databases and a main content area with a top navigation bar and several panels.

phpMyAdmin

Recent | Favorites

- New
- abcxyz
- CSDL_thi
- information_schema
- LeDucTrungTin.sql
- myDB
- mysql
- performance_schema
- quanly_ban_sua
- Quan_ly_Ban_Hang
- sys
- TracuuSP_MSSV.php

Server: localhost:3306

Databases | SQL | Status | User accounts | Export | Import | Settings | More

Server connection: User is not being used

- Server version: 8.0.21-0ubuntu0.20.04.4 - (Ubuntu)
- Protocol version: 10
- User: root@localhost
- Server charset: UTF-8 Unicode (utf8mb4)

Appearance settings

Language: English

Theme: pmahomme

Font size: 82%

More settings

Web server

- Apache/2.4.41 (Ubuntu)
- Database client version: libmysql - mysqlnd 7.4.3
- PHP extension: mysqli curl mbstring
- PHP version: 7.4.3

phpMyAdmin

- Version information: 4.9.5deb2
- Documentation
- Official Homepage
- Contribute
- Get support

Một số thao tác với phpMyAdmin

Từ trang **Home**, để tạo mới một DB, chọn **New->Điền Tên DB->Bảng mã** (tiếng việt: `utf8_unicode_ci`; `utf8mb4_unicode_ci`)

The screenshot shows the phpMyAdmin interface with the 'Databases' tab selected. The 'Create database' section is highlighted with two red arrows pointing to the database name field (containing 'tao_moi_DB') and the collation dropdown (set to 'utf8_unicode_ci'). Below this is a 'Filters' section with a search box. At the bottom, a table lists existing databases:

Database	Collation	Master replication	Action
<input type="checkbox"/> abcxyz	utf8_unicode_ci	✓ Replicated	Check privileges
<input type="checkbox"/> CSDL_thi	utf8_unicode_ci	✓ Replicated	Check privileges

Một số thao tác với phpMyAdmin

Để tạo mới một bảng trong phpMyAdmin, trên giao diện điền **tên bảng**->**số lượng cột**-> **Go**

The screenshot shows the phpMyAdmin interface. On the left is a sidebar with a tree view of databases, including 'New', 'abcxyz', 'CSDL_thi', 'information_schema', 'LeDucTrungTin.sql', 'myDB', 'mysql', 'performance_schema', 'quanly_ban_sua', 'Quan_ly_Ban_Hang', 'sys', and 'tao_moi_DB'. The top navigation bar includes tabs for 'Structure', 'SQL', 'Search', 'Query', 'Export', and 'More'. The main content area displays a message: 'No tables found in database.' Below this is the 'Create table' form. The 'Name' field is filled with 'people' and the 'Number of columns' field is filled with '4'. A 'Go' button is located at the bottom right of the form. A red error message at the bottom of the interface reads: 'mysqli_real_connect(): (HY000/1045): Access denied for user 'phpmyadmin'@'localhost' (using password: YES)'. Blue arrows indicate the sequence of actions: clicking the 'Name' field, the 'Number of columns' field, and finally the 'Go' button.

Một số thao tác với phpMyAdmin

Tạo các cột trong bảng, cần điền các thông tin về Name, Type, Length, Index, A.I.

Server: localhost:3306 » Database: tao_moi_DB » Table: people

Table name: people Add 1 column(s) Go

Name	Type	Length/Values	Default	Collation	Attributes	Null	Index	A.I.
ID	INT		None			<input type="checkbox"/>	PRIMARY	<input checked="" type="checkbox"/>
Ten	VARCHAR		None			<input type="checkbox"/>	---	<input type="checkbox"/>
Ho	VARCHAR		None			<input type="checkbox"/>	---	<input type="checkbox"/>
Tuoi	INT		None			<input type="checkbox"/>	---	<input type="checkbox"/>

Table comments:

Collation:

Storage engine:

PARTITION definition:

Partition by: (Expression or column list)

Partitions:

Add index

Index name: PRIMARY

Index choice: PRIMARY

+ Advanced Options

Column	Size
ID [int]	11

Go Cancel

Một số thao tác với phpMyAdmin

Để kiểm tra bảng vừa tạo, chọn vào tab **Structure**

Server: localhost:3306 » Database: tao_moi_DB » Table: people

[Browse](#)
[Structure](#)
[SQL](#)
[Search](#)
[Insert](#)
[Export](#)
[Import](#)
[Privileges](#)
[Operations](#)

[Table structure](#)
[Relation view](#)

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
<input type="checkbox"/> 1	ID	int			No	None		AUTO_INCREMENT	Change Drop More
<input type="checkbox"/> 2	Ten	varchar(30)	utf8_unicode_ci		No	None			Change Drop More
<input type="checkbox"/> 3	Ho	varchar(30)	utf8_unicode_ci		No	None			Change Drop More
<input type="checkbox"/> 4	Tuoi	int			No	None			Change Drop More

☐ Check all
 With selected: [Browse](#) Change Drop Primary Unique Index Fulltext

Print Move columns Normalize

Add column(s)

Indexes

Action	Keyname	Type	Unique	Packed	Column	Cardinality	Collation	Null	Comment
Edit Drop	PRIMARY	BTREE	Yes	No	ID	0	A	No	

Create an index on columns

Một số thao tác với phpMyAdmin

Để nhập dữ liệu vào bảng vừa tạo, chọn vào tab **Insert**

The screenshot shows the 'Insert' tab in phpMyAdmin. It displays a table with four columns: ID (int), Ten (varchar(30)), Ho (varchar(30)), and Tuổi (int). The 'Value' column shows the input values: 1 for ID, Trieu for Ten, Nguyen for Ho, and 20 for Tuổi. A blue box highlights the input fields, and a blue arrow points to the 'Go' button at the bottom right.

Column	Type	Function	Null	Value
ID	int			1
Ten	varchar(30)			Trieu
Ho	varchar(30)			Nguyen
Tuoi	int			20

Khi chọn **Go** phpMyAdmin sẽ tự động sinh ra code SQL trong thẻ **SQL**

The screenshot shows the 'SQL' tab in phpMyAdmin. It displays the generated SQL code for the insert operation. A blue box highlights the message '1 row inserted.' and the SQL code. Below the code, there is a section titled 'Run SQL query/queries on table tao_moi_DB.people:'.

```
1 row inserted.  
INSERT INTO 'people' ('ID', 'Ten', 'Ho', 'Tuoi') VALUES ('2', 'Trieu', 'Nguyen', '20');
```

Run SQL query/queries on table tao_moi_DB.people:

```
1 INSERT INTO 'people' ('ID', 'Ten', 'Ho', 'Tuoi') VALUES ('2', 'Trieu', 'Nguyen', '20');
```

Một số thao tác với phpMyAdmin

Ngoài ra, có thể sử dụng trực tiếp các lệnh **SQL** ở tab **SQL** mà không cần phải thông qua giao diện

Run SQL query/queries on table tao_moi_DB.people:

```
1 INSERT INTO `people` (`ID`, `Ten`, `Ho`, `Tuoi`) VALUES ('2','TRIEU1','NGUYEN','20')
```

Columns

- ID
- Ten
- Ho
- Tuoi

SELECT * SELECT INSERT UPDATE DELETE Clear Format

Get auto-saved query

☐ Bind parameters

[Delimiter :] ☒ Show this query here again ☐ Retain query box ☐ Rollback when finished ☒ Enable foreign key checks Go

Một số thao tác với phpMyAdmin

Để kiểm tra các giá trị vừa nhập vào các cột ở bảng, chọn vào tab **Browse**

The screenshot displays the phpMyAdmin interface. On the left is a tree view of the database structure, including 'mysql', 'performance_schema', 'quanly_ban_sua', 'Quan_ly_Ban_Ha', 'sys', 'tao_moi_DB', and a 'New' database. The 'people' table is selected under 'tao_moi_DB'. The main area shows the 'Browse' tab for the 'people' table. It includes a toolbar with 'Options', 'Edit', 'Copy', and 'Delete' icons. Below this is a table with the following data:

	ID	Ten	Ho	Tuoi
<input type="checkbox"/>	1	Trieu	Nguyen	20

Below the table, there are controls for 'Check all', 'With selected:', 'Edit', 'Copy', 'Delete', and 'Export'. At the bottom, there are options to 'Show all', 'Number of rows: 25', and a 'Filter rows: Search this table' input field.

Một số thao tác với phpMyAdmin

Để tạo **quan hệ** giữa các bảng, chúng ta cần vào bảng có chứa **Foreign Key constraints**, chọn **Structure->Relation view**. Trong ví dụ này, bảng *address* chứa khóa ngoại liên kết với khóa chính ở bảng *people*

The screenshot shows the phpMyAdmin interface for a database named 'tao_moi_DB'. The 'Structure' tab is selected for the 'address' table. The 'Relation view' sub-tab is active, showing the 'Foreign key constraints' section. The 'Constraint name' field is empty. The 'ON DELETE' and 'ON UPDATE' actions are set to 'RESTRICT'. The 'Indexes' section at the bottom shows a primary index on the 'id_diachi' column.

Actions	Constraint properties	Column	Foreign key constraint (INNODB)		
			Database	Table	Column
	Constraint name				
ON DELETE	RESTRICT				
ON UPDATE	RESTRICT				

Action	Keyname	Type	Unique	Packed	Column	Cardinality	Collation	Null	Comment
Edit	PRIMARY	BTREE	Yes	No	id_diachi	0	A	No	

Một số thao tác với phpMyAdmin

Sau khi chọn **Cột khóa ngoại**, tên DB, Bảng chứa khóa chính, **Cột khóa chính**, ta có được quan hệ giữa 2 bảng

[Browse](#)
[Structure](#)
[SQL](#)
[Search](#)
[Insert](#)
[Export](#)
[Import](#)
[Privileges](#)
[More](#)

Your SQL query has been executed successfully.

```
ALTER TABLE `address` ADD CONSTRAINT `_FK_people` FOREIGN KEY (`id_nguoi`) REFERENCES `people` (`ID`) ON DELETE RESTRICT ON UPDATE RESTRICT;
```

[\[Edit inline\]](#) [\[Edit\]](#) [\[Create PHP code\]](#)

[Table structure](#)
[Relation view](#)

Foreign key constraints

Actions	Constraint properties	Column	Foreign key constraint (INNODB)		
			Database	Table	Column
Drop	<input type="text" value="_FK_people"/> ON DELETE <input type="text" value="RESTRICT"/> ON UPDATE <input type="text" value="RESTRICT"/>	<input type="text" value="id_nguoi"/> + Add column	<input type="text" value="tao_moi_DB"/>	<input type="text" value="people"/>	<input type="text" value="ID"/>
	<input type="text" value="Constraint name"/> ON DELETE <input type="text" value="RESTRICT"/> ON UPDATE <input type="text" value="RESTRICT"/>	<input type="text"/> + Add column	<input type="text"/>	<input type="text"/>	<input type="text"/>

Một số thao tác với phpMyAdmin

Để xem quan hệ trực quan giữa hai bảng, chọn vào **DB Name** -> **More** -> **Designer**



Các bước thao tác CSDL trong PHP

- Mở kết nối đến CSDL
- Chọn CSDL
- Chọn bảng mã (nếu cần)
- Xây dựng câu truy vấn dữ liệu
- Thực thi câu truy vấn
- Xử lý kết quả trả về
- Dọn dẹp
- Đóng kết nối

Mở kết nối đến CSDL

Sử dụng thư viện cải tiến `mysqli`

```
1 // mysqli
2 mysqli_connect('hostname', 'username', 'password', 'dbname',
    'port', 'socket');
3 //mysqli: recommended
4 new mysqli($hostname, $username, $password, $dbname);
5 // mysql
6 mysql_connect('hostname', 'username', 'password');
```

- **hostname**: là địa chỉ host của bạn, thông thường là localhost
- **username**: là tên đăng nhập vào database
- **password**: là mật khẩu kết nối vào database
- **dbname**: tên database bạn chọn để xử lý
- **port**: cổng kết nối
- **socket**: phương thức socket kết nối

Ví dụ

```
1 $conn = mysqli_connect('localhost', 'root', '', 'demo') or
2 die('database'.mysqli_connect_error());
```

Xây dựng câu truy vấn

Xây dựng các câu truy vấn của SQL như:

- INSERT

```
1 $strSQL = "INSERT INTO Users  
2 (UserName, Password) VALUES ('admin','123')"
```

- DELETE

```
1 $strSQL = "DELETE FROM Users  
2 WHERE UserName = 'lvminh'"
```

- SELECT

```
1 $strSQL = "SELECT * FROM Users  
2 WHERE UserName = 'admin' "
```

- UPDATE

```
1 $strSQL = "UPDATE Users SET Password = 'abc'  
2 WHERE UserName = 'admin' "
```

Ví dụ

```
1 $sql='select * from city';
```

Thực thi câu truy vấn

```
1 // OOP mysqli
2 $result = mysqli->query("query");
3 // mysqli
4 $result = mysqli_query($conn, "query");
5 // mysql
6 $result = mysql_query("query", $conn)
```

- **\$conn**: biến kết nối CSDL
- **\$sql**: biến lưu trữ câu truy vấn

Ví dụ

```
<?php
$sql= "INSERT INTO Users (UserName , Password)
VALUES ('admin' , '123')";
$result = mysqli_query($conn, $sql);
if (!$result) die ('Câu truy vấn sai');
?>
```


Xử lý kết quả trả về

Hiển thị dữ liệu

- Sử dụng một trong các hàm:

- ▶ `$row=mysqli_fetch_assoc($result);`
- ▶ `$row=mysqli_fetch_row($result);`
- ▶ `$row=mysqli_fetch_array($result);`
- ▶ `$row=mysqli_fetch_object($result);`

- Đếm số lượng **FIELD**:

```
$num=mysqli_num_fields($result);
```

- Đếm số lượng dòng

```
$rows= mysqli_num_rows($result);
```

- Truy cập đến từng **FIELD**:

```
echo $row["UserName"];  
//or  
echo $row[0];
```

Xử lý kết quả trả về

Hiển thị dữ liệu

- Duyệt dữ liệu có các cách sau
 - ▶ Duyệt dữ liệu theo dạng mỗi mẫu tin là **một dòng**
`mysqli_fetch_row()`
 - ▶ Duyệt theo dạng mỗi mẫu tin là **một mảng**
`mysqli_fetch_array()`
 - ▶ Duyệt theo dạng mỗi mẫu tin là **một đối tượng**
`mysqli_fetch_object()`

Dưới dạng dòng: `mysqli_fetch_row`

- Duyệt dữ liệu theo dạng mỗi mẫu tin là **một dòng**, cú pháp: `mysqli_fetch_row($result)`
 - ▶ `$result`: là tham số bắt buộc xác định con trỏ dữ liệu. Con trỏ dữ liệu là **kết quả trả về của hàm `mysqli_query($conn,$sql)`**
 - ▶ Kết quả **trả về là một mảng** (có chỉ số là số) chứa giá trị của một dòng dữ liệu với **mỗi phần tử là nội dung của một cột** → truy cập bằng cách gọi từng phần tử của mảng `$row[0]`, `$row[1]`, `$row[2]`, ...

Dưới dạng dòng: `mysqli_fetch_row`

Ví dụ duyệt dữ liệu theo dạng mỗi mẫu tin là một **dòng**:

```
<?php
$conn = mysqli_connect('localhost', 'root', '', 'demo') or
    die ('Khong the ket noi toi database'.
        mysqli_connect_error());
$sql = "SELECT * FROM Custommers";
$result = mysqli_query($conn, $sql);
if(mysqli_num_rows($result)!=0){
    while($row = mysqli_fetch_row($result)){
        var_dump($row);
    }
}
?>
```

```
Array (
  [0] => kh001
  [1] => Khuất Thùy Phương
  [2] => 1
  [3] => A21 Nguyễn Oanh quận Gò Vấp
  [4] => 9874125
  [5] => ktphuong@hcmuns.edu.vn )
```

Dưới dạng mảng: `mysqli_fetch_array`

- Duyệt dữ liệu theo dạng mỗi mẫu tin là **một mảng**, cú pháp: `mysqli_fetch_array($result)`
 - ▶ Kết quả trả về là một **mảng (có chỉ số chuỗi)** chứa **giá trị của một dòng** dữ liệu với mỗi phần tử là nội dung của một cột.
 - ▶ Truy cập bằng cách gọi từng phần tử của mảng: `$row["tên cột 1"]`, `$row["tên cột 2"]`, `$row["tên cột 3"]`, ...

Dưới dạng mảng: `mysqli_fetch_array`

Ví dụ duyệt dữ liệu theo dạng mỗi mẫu tin là một **mảng**:

```
<?php
$conn = mysqli_connect('localhost', 'root', '', 'demo') or
die ('Khong the ket noi toi database'. mysqli_connect_error()
);
$sql = "SELECT * FROM Custommers";
$result = mysqli_query($conn, $sql);
if(mysqli_num_rows($result)!=0){
    while($row = mysqli_fetch_array($result)){
        var_dump($row);}
}
?>
```

```
Array( [Ma_KH]=>kh001
       [Ho_ten]=>Nguyễn Văn Minh
       [Gioi_tinh]=> 1
       [Dia_chi]=> 2 Trần Khánh Dư, NT
       [Dien_thoai]=>098712345
       [Email]=>nvminh@yahoo.com)
```

Câu truy vấn DELETE

Ví dụ:

```
<?php
$conn = mysqli_connect('localhost', 'root', '', 'demo') or
die ('Khong the ket noi toi database');
// Cau truy van
$sql = "DELETE FROM Customers WHERE id=1";
// DELETE
if (mysqli_query($conn, $sql)){
    echo 'xoa thanh cong';}
// Ngat ket noi
mysqli_close($conn);

?>
```

Câu truy vấn UPDATE

Ví dụ:

```
<?php
$conn = mysqli_connect('localhost', 'root', '', 'demo') or
die ('Khong the ket noi toi database');
// Cau truy van
$sql = "UPDATE Customers SET name='Le Minh An' where id=2
";
// DELETE
if (mysqli_query($conn, $sql)){
    echo 'Cap nhat thanh cong';
}
// Ngat ket noi
mysqli_close($conn);
?>
```


- Xóa kết quả khỏi bộ nhớ
 - ▶ `mysqli_free_result($result);`
- Sau khi thực thi xong thì ngắt kết nối database
 - ▶ `mysqli_close($conn);`

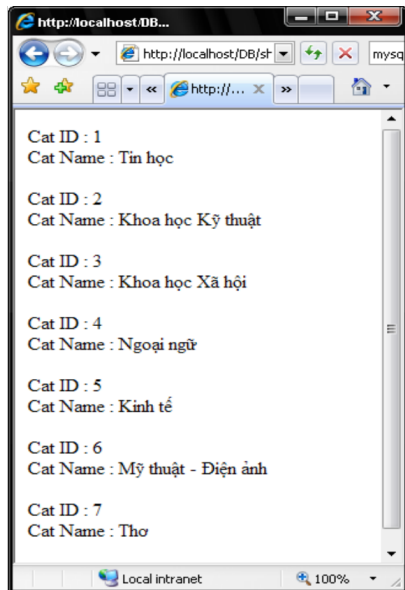
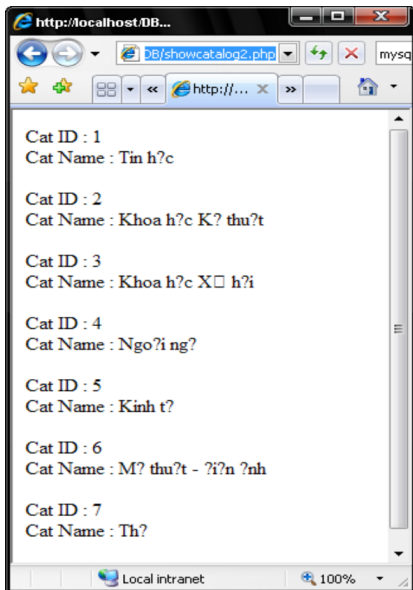
Tổng hợp các bước thao tác CSDL MySQL

```
<?php
// 1. Ket noi CSDL
$conn = mysqli_connect('localhost', 'root', '', 'qlbansua')
    or
die ('Khong the ket noi toi database'. mysqli_connect_error()
    );

// 2. Chuan bi cau truy van & 3. Thuc thi cau truy van
$sql = "SELECT * FROM Khach_hang";
$result = mysqli_query($conn, $sql);

// 4. Xu ly du lieu tra ve
if(mysqli_num_rows($result)!=0){
    while ($row = mysqli_fetch_array($result)){
        for ($i=0; $i<mysqli_num_fields($result); $i++){
            echo $row[$i] . " ";
        }
    }
}

// 5. Xoa ket qua khoi vung nho va Dong ket noi
mysqli_free_result($result);
mysqli_close($conn);
?>
```



Khắc phục lỗi font tiếng Việt

- Kiểm tra lại **Collation** của Table hoặc Database có là **utf8_*** hay chưa?
- Sử dụng hàm **mysqli_set_charset(\$conn, 'UTF8');**
- Ví dụ

```
$conn = mysqli_connect('localhost', 'root', '', 'demo') or
die ('Khong the ket noi toi database'.mysqli_connect_error());
;
//-----
mysqli_set_charset($conn, 'UTF8');
//-----
$sql = "SELECT * FROM Custommers";
$result = mysqli_query($conn, $sql);
if(mysqli_num_rows($result) != 0){
    while($row = mysqli_fetch_array($result)){
        var_dump($row);
    }
}
```

<?php

```
$hostName = "localhost";  
$databaseName = "bookstoreDB";  
$username = "root";  
$password = "root";
```

db.inc

?>

<?php

```
function showError( )  
{  
    die("Error " . mysql_errno( ) . " : " . mysql_error( ));  
}
```

error.inc

?>

<?php

```
include 'db.inc';  
include 'error.inc';
```

?>

Cách sử dụng:

- `include()` cũng có cách sử dụng tương tự như `require()`.
- Điểm khác nhau của `include()` và `require()`: khi nội dung bị lỗi thì dùng `require()` sẽ xuất hiện thông báo lỗi trong khi đó dùng `include()` sẽ chỉ xuất hiện cảnh báo.
- Trong những file có dùng `require()` thì không nên sử dụng các cấu trúc điều khiển vì sẽ không hiệu quả hoặc sử dụng `include()` thay cho `require()`.

Cú pháp:

```
include("path_to_PhpFile");
```

- Hiển thị dữ liệu không định dạng
- Hiển thị dữ liệu có yêu cầu
 - ▶ Định dạng hiển thị dữ liệu
 - ▶ Tạo các cột tùy biến
 - ▶ Phân trang
 - ▶ Liên kết trang có chuỗi tham số

Hiển thị dữ liệu không định dạng

- Cách đơn giản nhất là hiển thị dữ liệu theo **dạng bảng** hoặc **dạng text**.
- Ví dụ: hiển thị thông tin bảng khách hàng

THÔNG TIN KHÁCH HÀNG

Mã KH	Tên khách hàng	Giới tính	Địa chỉ	Số điện thoại
kh001	Khuất Thủy Phương	1	A21 Nguyễn Oanh quận Gò Vấp	9874125
kh002	Đỗ Lâm Thiên	0	357 Lê Hồng Phong Q.10	8351056
kh003	Phạm Thị Nhung	1	56 Đinh Tiên Hoàng quận 1	9745698
kh004	Nguyễn Khắc Thiện	0	12bis Đường 3-2 quận 10	8769128
kh005	Tô Trần Hồ Giăng	0	75 Nguyễn Kiệm quận Gò Vấp	5792564
kh006	Nguyễn Kiến Thi	1	357 Lê Hồng Phong Q.10	9874125
kh008	Nguyễn Anh Tuấn	0	1/2bis Nơ Trang Long Q.BT TP.HCM	8753159

Hiển thị dữ liệu có yêu cầu

- Yêu cầu đơn giản nhất của một trang Web là dữ liệu hiển thị phải được định dạng giúp cho người dùng dễ dàng đọc thông tin.
- Các định dạng thông thường là **định dạng cột, dòng, kích thước, màu sắc, kiểu chữ, ...** cho nội dung hiển thị.
- Để thực hiện được các yêu cầu về định dạng → cần phải xem xét và tính toán cho các nội dung cần hiển thị.

Hiển thị dữ liệu có yêu cầu

- Ví dụ hiển thị thông tin khách hàng với định dạng như sau:
 - ▶ Dòng **tiêu đề** của bảng sẽ được **canh giữa, chữ in đậm, màu sắc** phân biệt so với các dòng khác.
 - ▶ Các dòng thông tin của khách hàng: **dòng chẵn và dòng lẻ có màu nền khác nhau**, cột giới tính được **canh giữa**.

Dòng tiêu đề

THÔNG TIN KHÁCH HÀNG

Mã KH	Tên khách hàng	Giới tính	Địa chỉ	Số điện thoại
kh001	Khuất Thủy Phương	1	A21 Nguyễn Oanh quận Gò Vấp	9874125
kh002	Đỗ Lâm Thiên	0	357 Lê Hồng Phong Q.10	8351056
kh003	Phạm Thị Nhung	1	56 Đinh Tiên Hoàng quận 1	9745698
kh004	Nguyễn Khắc Thiện	0	12bis Đường 3-2 quận 10	8769128
kh005	Tô Trần Hồ Giảng	0	75 Nguyễn Kiệm quận Gò Vấp	5792564
kh006	Nguyễn Kiến Thi	1	357 Lê Hồng Phong Q.10	9874125
kh008	Nguyễn Anh Tuấn	0	1/2bis Nơ Trang Long Q.BT TP.HCM	8753159

Dòng lẻ

Dòng chẵn

Hiển thị dữ liệu có yêu cầu

Tạo các cột tùy biến

- Nhu cầu hiển thị dữ liệu một cách rõ ràng, chi tiết, dễ hiểu trở thành một yêu cầu tất yếu khi thiết kế trang Web.
- Một trong những cách hiển thị dữ liệu đáp ứng nhu cầu đó là tạo các cột tùy biến.
- Ví dụ: trong CSDL thường dùng kiểu dữ liệu boolean cho cột giới tính → cột này sẽ có một trong hai giá trị là 0 hoặc 1 và tùy theo quy ước để xác định 0 hoặc 1 thuộc giới tính gì. Nếu cột giới tính hiển thị 0, 1 và sau đó lại có dòng chú giải thì sẽ không thuận tiện → dùng hình ảnh hoặc checkbox,... để hiển thị giới tính. Cách hiển thị này vừa đẹp mắt, vừa dễ hiểu.

Hiển thị dữ liệu có yêu cầu

Ví dụ tạo các cột tùy biến

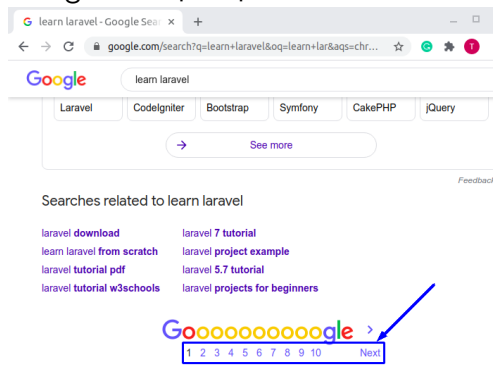
THÔNG TIN KHÁCH HÀNG

Mã KH	Tên khách hàng	Giới tính	Địa chỉ	Số điện thoại
kh001	Khuất Thùy Phương		A21 Nguyễn Oanh quận Gò Vấp	9874125
kh002	Đỗ Lâm Thiên		357 Lê Hồng Phong Q.10	8351056
kh003	Phạm Thị Nhung		56 Đinh Tiên Hoàng quận 1	9745698
kh004	Nguyễn Khắc Thiện		12bis Đường 3-2 quận 10	8769128
kh005	Tô Trần Hồ Giàng		75 Nguyễn Kiệm quận Gò Vấp	5792564
kh006	Nguyễn Kiến Thi		357 Lê Hồng Phong Q.10	9874125
kh008	Nguyễn Anh Tuấn		1/2bis Nơ Trang Long Q.BT TP.HCM	8753159

Hiển thị dữ liệu có yêu cầu

Tạo Phân trang

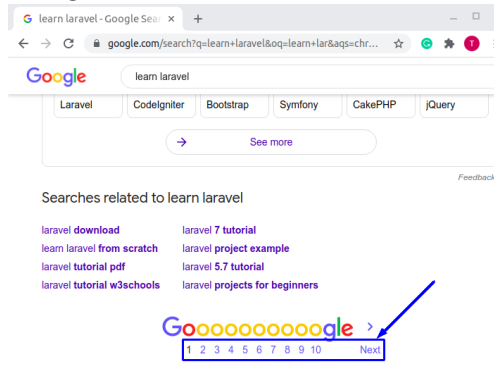
- Đối với những bảng có **số lượng mẫu tin lớn** khi hiển thị dữ liệu sẽ làm cho người dùng cảm thấy khó xem → thực hiện việc phân trang
- Phân trang dữ liệu giúp cho việc **xem và tìm kiếm** thông tin dễ dàng, nhanh chóng và thuận tiện.



Hiển thị dữ liệu có yêu cầu

Tạo Phân trang

- Đối với những bảng có **số lượng mẫu tin lớn** khi hiển thị dữ liệu sẽ làm cho người dùng cảm thấy khó xem → thực hiện việc phân trang
- Phân trang dữ liệu giúp cho việc **xem và tìm kiếm** thông tin dễ dàng, nhanh chóng và thuận tiện.



Phân Trang

- Trường hợp dữ liệu lấy về có quá nhiều mẫu tin (hiển thị bảng quá dài)
- Chia dữ liệu hiển thị trên nhiều trang
 - ▶ Tổng số kết quả trả về: `$numRows`
 - ▶ Số kết quả hiển thị trên 1 trang: `$rowsPerPage`
 - ▶ Tổng số trang hiển thị: `$maxPage`
$$\$maxPage = \text{ceil}(\$numRows / \$rowsPerPage);$$
 - ▶ Số thứ tự trang hiển thị: `$pageNum = 1, 2, \dots`
 - ▶ Chỉ số kết quả bắt đầu trong trang thứ `$pageNum`:
$$\$offset = (\$pageNum - 1) * \$rowsPerPage;$$

Phân Trang

The screenshot shows a Google search results page for the query "Thiết kế web". Several annotations in red starburst shapes highlight key pagination elements:

- offset**: Points to the search bar area.
- rowsPerPage**: Points to the "Kết quả 1 - 10" (Results 1 - 10) text.
- numRows**: Points to the "khoảng 441.000" (about 441,000) text.
- maxPage**: Points to the "Tiếp" (Next) button in the pagination links.
- pageNum**: Points to the current page number "1" in the pagination sequence "1 2 3 4 5 6 7 8 9 10".

The search results include several links related to web design services, such as "Thiết Kế Web Độc Đáo", "Thiết Kế Web Động", and "Thiết Kế Web Tĩnh".

Các Bước Thực Hiện Phân Trang

- Tính các thông số phân trang (`$rowsPerPage`, `$pageNum`, `$offset`)
- Lấy thông tin cần hiển thị (`$numRows`)
- Hiển thị thông tin của trang hiện tại (`$offset`, `$rowsPerPage`)
- Tạo liên kết đến các trang (`$maxPage`, `$pageNum`)

Ví dụ - Hiển thị thông tin sửa

Hiển thị tất cả mẫu tin trên một trang

```
<?php
$conn = mysqli_connect('localhost', 'root', '', 'qlbansua');
mysqli_set_charset($conn, 'UTF8');
$sql=" 'select Ma_sua,ten_sua,Trong_luong,Don_gia from sua";
$result = mysqli_query($conn, $sql);
echo "<p align='center'><font size='5'>THONG TIN SUA</font></P>"
;
echo "<table align='center' width='700' border='1' cellpadding
    ='2'
    style='border-collapse: collapse'>";
echo "<tr><th width='50'>STT</th>
    <th width='50'>Ma_Sua</th>
    <th width='150'>Ten_Sua</th>
    <th width='200'>Trong_Luong</th></tr>";
if(mysqli_num_rows($result) > 0){
    $stt=1; while($rows=mysqli_fetch_row($result)){
        echo "<tr>";
        echo "<td>$stt</td>";
        echo "<td>$rows[0]</td>";
        echo "<td>$rows[1]</td>";
        echo "<td>$rows[2]</td>";
        echo "</tr>";
    }
}
```

Tính các thông số phân trang

```
//phan trang
$rowsPerPage=5; // So mau tin tren moi trang, trong truong hop nay
                 cho bang 5
if (!isset($_GET['page']))
{
    $_GET['page'] = 1;
}
//Vi tri cua mau tin dau tien tren moi trang
$offset =($_GET['page']-1)*$rowsPerPage;
//Lay $rowsPerPage mau tin, bat dau tu vi tri $offset
$query="Select * from sua LIMIT $offset, $rowsPerPage";
$result = mysqli_query($conn,$query);
//Tong so mau tin can hien thi
$numRows= mysqli_num_rows($result);
//Tinh tong so trang
$maxPage = ceil($numRows / $rowsPerPage);
```

Tính các thông số phân trang

Ví dụ khi phân thành 5 hàng/1 trang

THÔNG TIN SỮA

SO TT	Tên Sữa	Hang Sữa	Loai Sữa	Trong Luong	Don gia
AB0001	Gain Advance	AB	SB	400	107000
AB0002	Gain IQ	AB	SB	400	107000
AB0003	Abbott Grow	AB	SB	400	87000
AB0004	Abbott Grow School	AB	SB	400	87000
AB0005	Abbott Pedia Sure	AB	SB	400	146000

Tạo link đến các trang

Tạo link tương ứng tới các trang

```
for ($i=1 ; $i<=$maxPage ; $i++)
{
    if ($i == $_GET['page'])
    {
        echo '<b>Trang'. $i. '</b>_';
    }
    else
    {
        echo "<a href=" . $_SERVER['PHP_SELF'] . "?page="
        . $i. ">Trang". $i. "</a>_";
    }
}
```

Thêm nút Back, Next

Nút Back

```
echo "<a_href=" . $_SERVER['PHP_SELF'] . "?page="
    . ($_GET['page']-1) . ">Back</a>";
for ($i=1 ; $i<=$maxPage ; $i++) //Nut Back
{
    if ($i == $_GET['page'])
        echo '<b>Trang'. $i . '</b>';
    else
        echo "<a_href=" . $_SERVER['PHP_SELF'] . "?page=" . $i .
            ">Trang". $i . "</a>";
}
```

Nút Next

```
echo "<a_href=" . $_SERVER['PHP_SELF'] . "?page="
    . ($_GET['page']+1) . ">Next</a>"; //Next
```

Thêm nút Back, Next

Lưu ý

- Điều kiện để hiển thị nút **Back** là trang hiện tại > 1

```
if ($_GET['page'] > 1)
{
    echo "<a href=" . $_SERVER['PHP_SELF'] . "?page=" . ($_GET['page']-1) . ">Back</a>";
}
```

- Điều kiện để hiển thị nút **next** là trang hiện tại $< \text{tổng số trang}$

```
if ($_GET['page'] < $maxPage)
{
    echo "<a href=" . $_SERVER['PHP_SELF'] . "?page=" . ($_GET['page']+1) . ">Next</a>";
}
```

Bài Tập 1

Thiết kế form nhập dữ liệu như hình dưới

Nhap Thông tin SV

Ten	<input type="text" value="Trieu"/>
Ho	<input type="text" value="Nguyen"/>
Địa chỉ	<input type="text" value="1234 a@"/>
ID của Lớp	<input type="text" value="1"/>

New record created successfully

Bài Tập 1

Mô tả chức năng:

- Khi nhấn nút gửi, dữ liệu sẽ lưu vào database.
- Khi nhấn vào nút xem kết quả, hiển thị dữ liệu những người có ID của Lop bằng 1 trong DB.

THONG TIN SINH VIEN

[Tro ve trang truoc](#)

STT	Ten	Ho	Địa chỉ	Id lop
35	Trieu	Nguyen	1234 a@@	1
40	trieu 2	Nguyen 2	1234 a@@nha trang	1
41	trieu 3	Nguyen 3	1234 a@@nha trang	1

Bài Tập 2

Thiết kế website có chức năng CRUD (có thể sử dụng thủ tục hoặc hướng đối tượng—code tham khảo) như hình dưới



Advanced CRUD App Using PHP & MySQLi Prepared Statement (Object Oriented)

Updated Successfully!

Add Record

Choose File No file chosen

Add Record

Records Present In The Database

Show 10 entries

Search:

#	Image	Name	Email	Phone	Action
15		TRIEU	trieu.science@gmail.com	89625510266	Details Delete Edit

Showing 1 to 1 of 1 entries

Previous 1 Next

Reference

- [1] Lê Thị Bích Hằng, Bài giảng *Phát Triển Phần Mềm Mã Nguồn Mở*.
- [2] <https://www.w3schools.com/php/>