Bài tập thực hành 06

**MSSV:**

**Họ và tên sinh viên:**

Tạo CRUD với Laravel và CakePHP

**Mục tiêu cần đạt**:

Sau buổi này, các bạn cần nắm thao tác tạo các trang cơ bản CRUD (Create-Read-Update-Delete) với Laravel và CakePHP:

* Cài đặt project CakePHP với composer
* Thiết lập các lệnh sinh code tự động với Cakephp, nắm các quy tắc đặt tên
* Cài đặt gói sinh code tự động Laravel, nắm các lệnh và quy trình để sinh code với Laravel

Các bạn thực hiện viết code như hướng dẫn để làm quen, sau đó trả lời các **Yêu cầu** ở dưới bài hướng dẫn.

**Gợi ý Tham khảo:**

* <https://viblo.asia/p/gioi-thieu-package-laravel-generator-Eb85oMRkZ2G>
* [https://github.com/CrestApps/laravel-code-generato](https://github.com/CrestApps/laravel-code-generator)
* <https://book.cakephp.org/3/en/installation.html>: Cài đặt bản v3
* <https://book.cakephp.org/4/en/installation.html>: Cài đặt bản v4
* <https://book.cakephp.org/4/en/console-commands.html>: Các lệnh console cho cakephp
* <https://book.cakephp.org/4/en/orm/retrieving-data-and-resultsets.html#using-finders-to-load-data> và các nguồn khác mà sinh viên tìm được

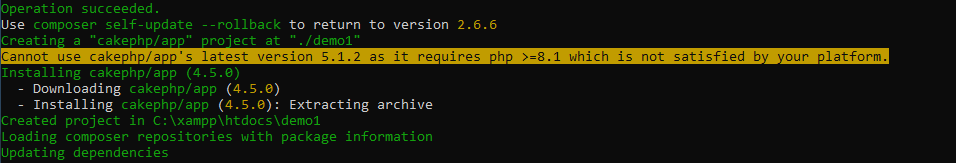
# Yêu cầu bài thực hành:

1. Bạn hãy chạy tất cả các lệnh hướng dẫn ở trên và chụp lại màn hình kết quả.
   1. Tạo project với Cakephp

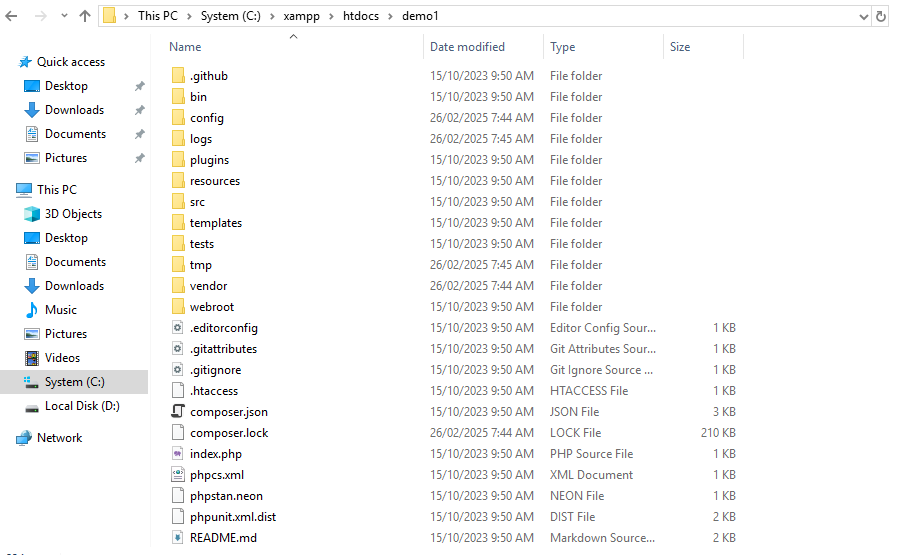
* Các bạn cd vào xampp\htdocs sau đó dùng composer để tạo project Cakephp tên **demo1 bản v3.9**:

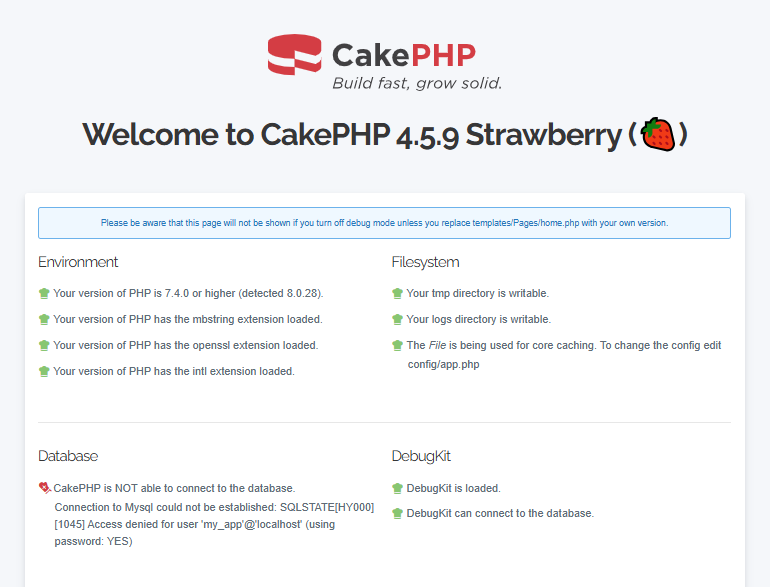
*composer create-project --prefer-dist cakephp/app:^3.9 demo1*

* Quá trình tạo project demo1 sẽ mất vài phút

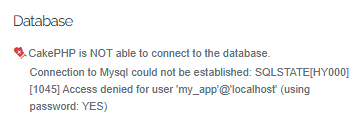


* Cấu trúc các thư mục và tập tin được tạo trong demo1:



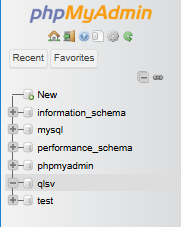


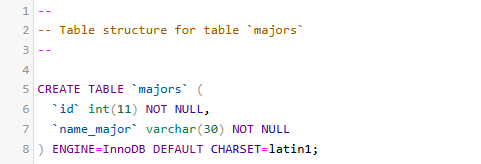
* Chú ý là lúc này chúng ta chưa thiết lập kết nối với csdl, nên khi kéo xuống dưới ta sẽ thấy:

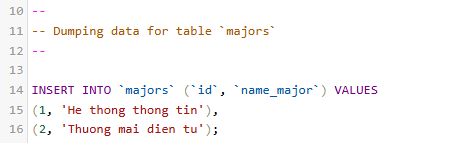


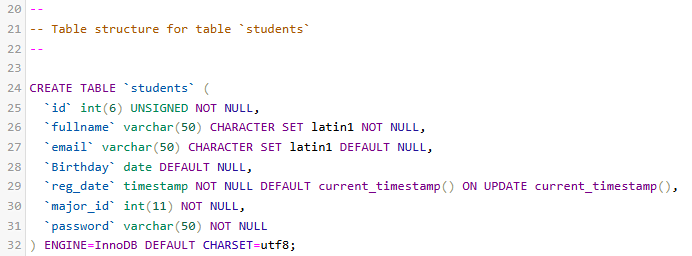
* 1. Tạo CSDL để thực hiện kết nối với project Cakephp vừa tạo:

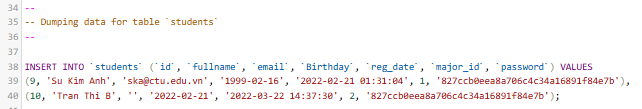
- Để demo kết nối dữ liệu, chúng ta tạo 1 csdl **qlsv** ví dụ như sau (lệnh đế kết nối khóa ngoại các bạn tự thực hiện):

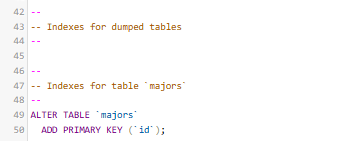


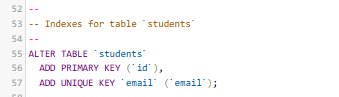


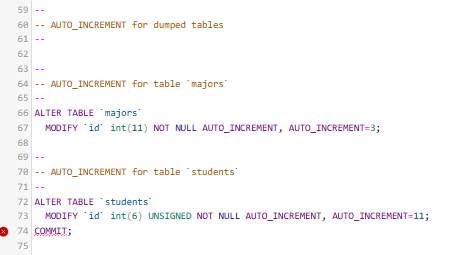




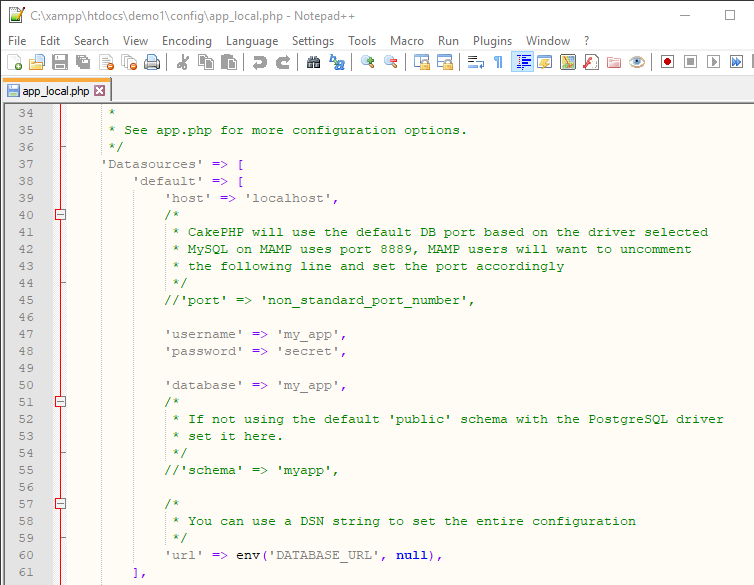




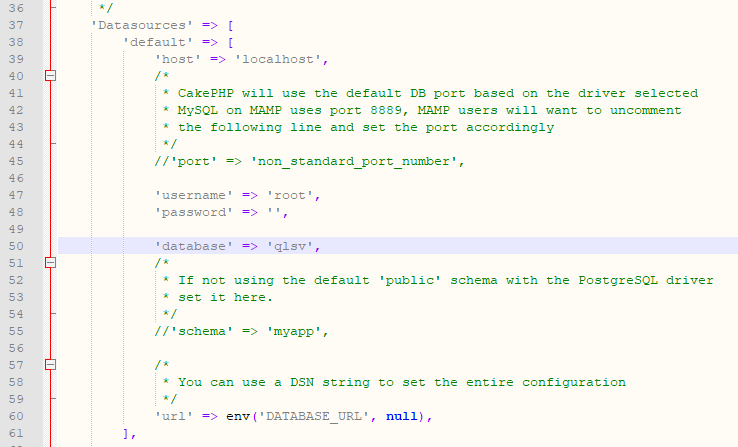




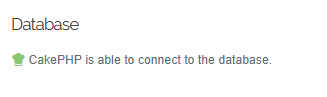
* Truy cập đến C:\xampp\htdocs\demo1\config, mở file cấu hình **app\_local.php**



* Chỉnh sửa lại username và password thành các tài khoản kết nối mysql trên phpmyadmin hợp lý (thường mặc định là root, password rỗng), chúng ta tạo csdl tên là qlsv nên khi chỉnh sửa sẽ là:



* Bấm f5 lại đường link: <http://localhost/demo1/>, ta sẽ thấy:



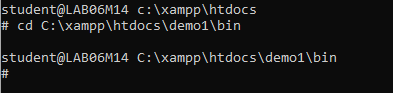
* Lúc này chúng ta đã kết nối đến CSDL thành công sau khi sửa file cấu hình ở app\_local.php.

Sinh code tự động với CakePHP

Sau khi chúng ta tạo CSDL qlsv có 2 bảng students và majors, các bảng này chúng ta đã đặt theo quy cách đặt tên của Cakephp và chúng ta có thể dùng lệnh hỗ trợ trong framework cakephp để sinh code tự động.

* Chúng ta trở lại giao diện dòng lệnh cmd ban nãy và gõ:

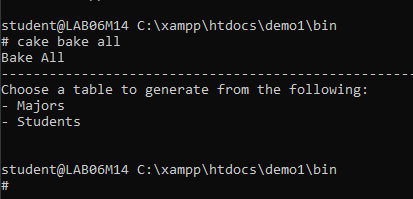
*cd C:\xampp\htdocs\demo1\bin*



* 1. **Gõ lệnh CMD để tạo CRUD từ bảng dữ liệu:**

*cake bake all*

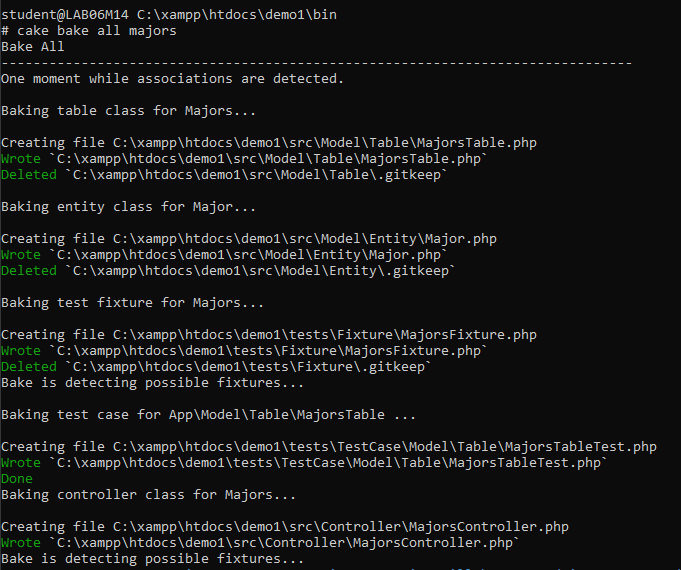
* Chúng ta thấy cakephp nhận dạng được các bảng trong CSDL đã kết nối.

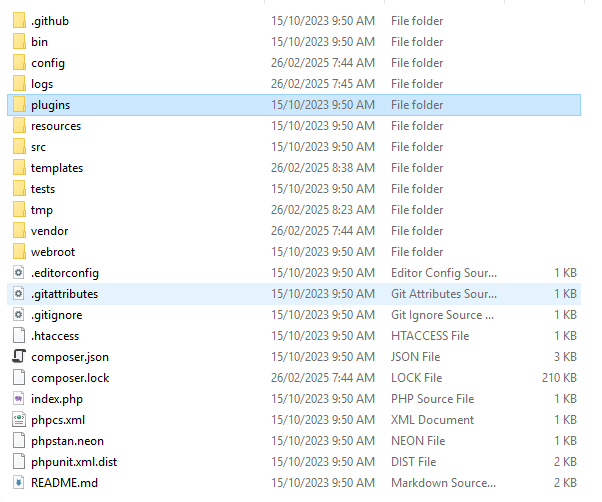


* Để tạo CRUD cho bảng majors (là bảng cha), ta thực hiện lệnh

*cake bake all majors*

Một loạt tập tin sẽ được tạo, các tập tin này hình thành từ kiến trúc M-V-C (Model-View Controller) với các giao diện index, thêm, sửa, hiển thị chi tiết của bảng majors nằm trong các thư mục Model, View (hoặc trong Template), Controller

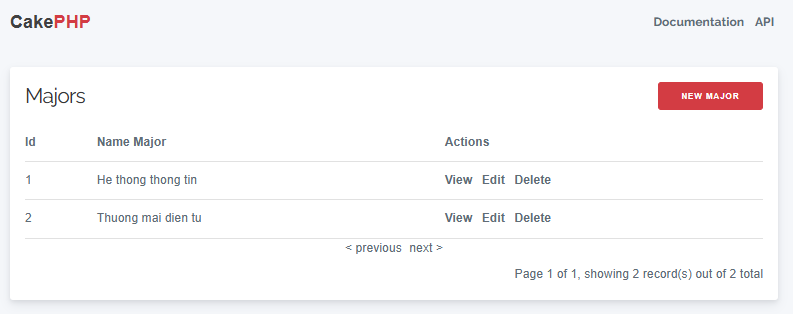




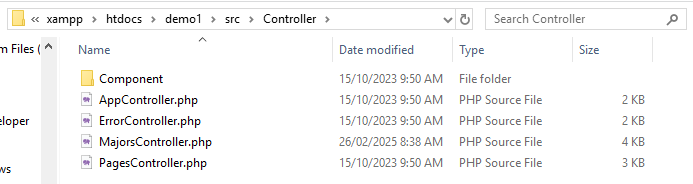
* Giờ lên trình duyệt ta gõ: <http://localhost/demo1/majors>



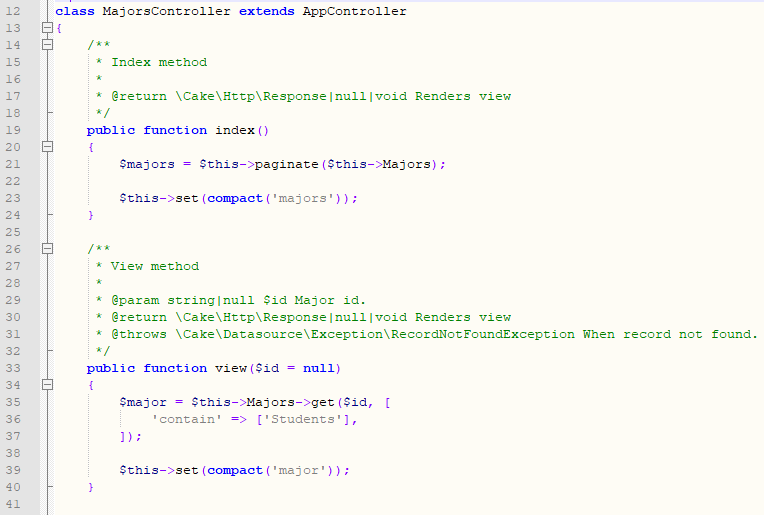
Ta sẽ truy cập đến trang index thể hiện danh sách các majors có trong bảng majors với những phương thức đã được cài đặt sẵn: hiển thị chi tiết (view), sửa (edit), và xóa (delete).



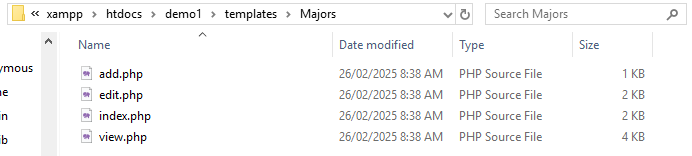
* Các phương thức này đã được cakephp định nghĩa trong Controller: majors:



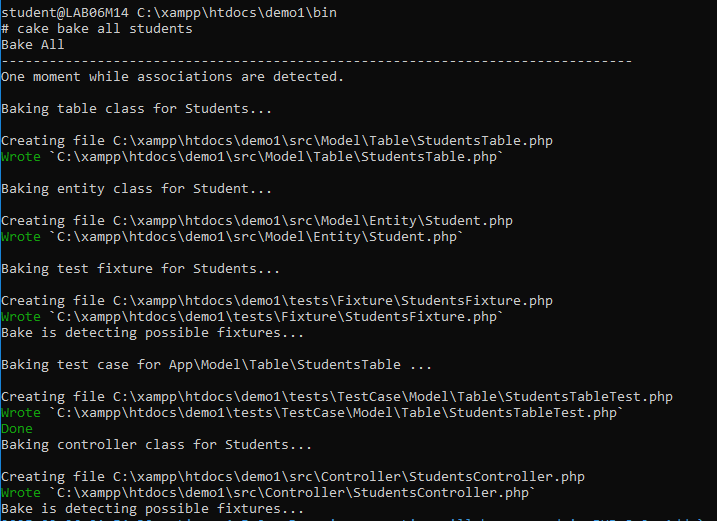
* Mở tập tin MajorsController.php, chúng ta sẽ thấy các phương thức đã được định nghĩa trong tập tin này:



* Các phương thức định nghĩa trong Controller sẽ xử lý những hành động trong các tập tin ở phần hiển thị nằm trong thư mục Template/Majors/



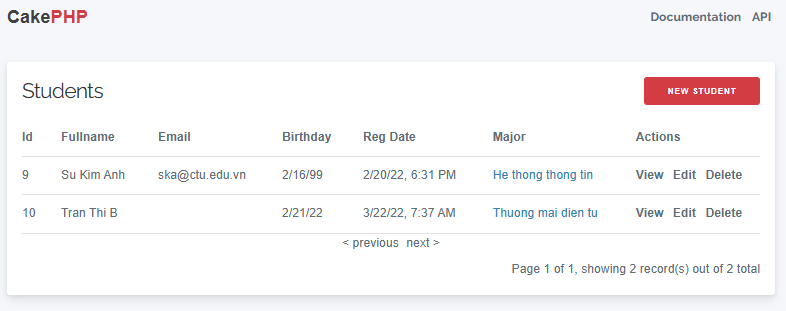
* Giờ lặp lại tương tự đối với bảng students:



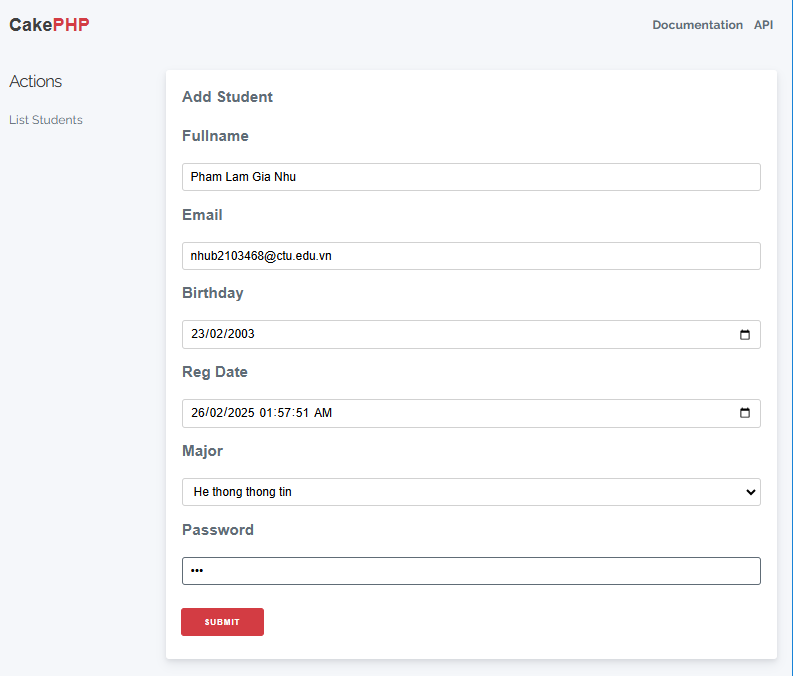
* Truy cập thử: <http://localhost/demo1/students>

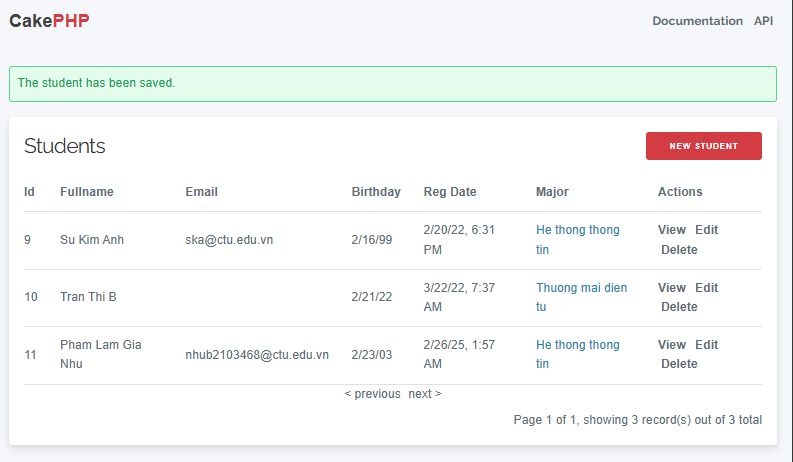


Ta sẽ truy cập đến trang index thể hiện danh sách các majors có trong bảng students với những phương thức đã được cài đặt sẵn: hiển thị chi tiết (view), sửa (edit), và xóa (delete).



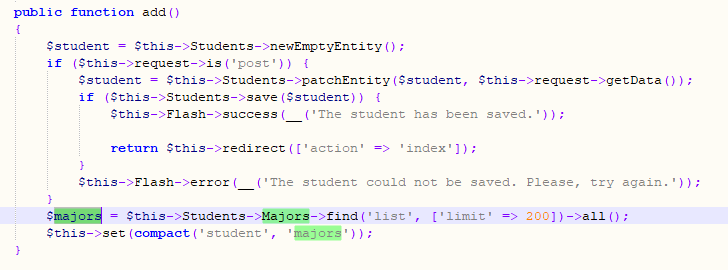
* Click vào “New Student” để đến form tạo sinh viên mới:



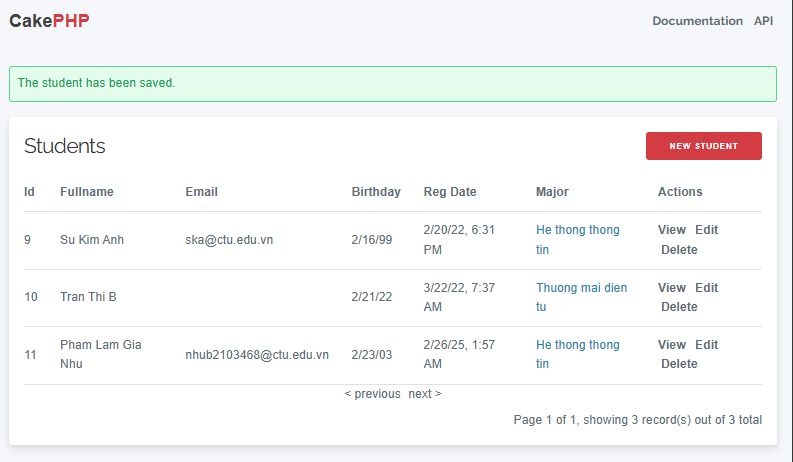


* Ta sẽ thấy được cakephp tự động tạo cho chúng ta các ô nhập tùy theo kiểu dữ liệu mà chúng ta đã tùy chỉnh trong MySQL, ngay cả khóa ngoại cũng sẽ là dạng combobox (Major)

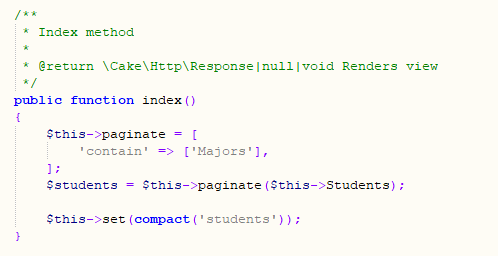
Quan sát trong phương thức add trong Controller của Student ta sẽ thấy biến này được thấy từ bảng Majors đã được kết nối tự động với bảng Student



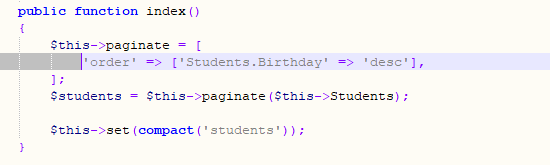
* 1. **Để sắp xêp dữ liệu hiển thị trong trang index, ví dụ ds ban đầu trong students:**

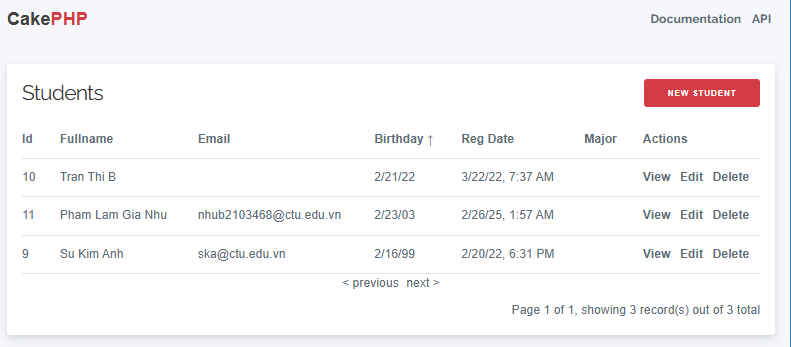


* Ta muốn sắp xếp dữ liệu với ngày sinh lớn hơn sẽ hiển thị trước, chúng ta sẽ vào Controller của Students:



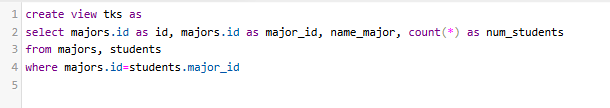
* Chỉnh sửa nội dung phương thức index() tùy chỉnh sắp xếp giảm dần theo ngày sinh:

  
- Quan sát kết quả:

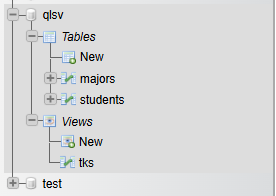


* 1. **Hướng dẫn tạo CRUD từ view CSDL với CakePHP**
* Giả sử chúng ta muốn hiển thị danh sách các chuyên ngành cùng với số thành viên trong chuyên ngành đó, chúng ta có thể dùng view CSDL để tạo query trả về kết quả mong muốn và rồi dùng Cakephp để tạo ra giao diện web hiển thị kết quả từ query này.

Trước tiên chúng ta viết lệnh tạo view (vd tên là tks) để thống kê số lượng sinh viên của mỗi chuyên ngành

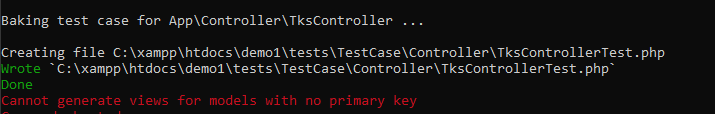


* Sau khi thực hiện xong ta sẽ thấy xuất hiện view này từ PHPmyadmin

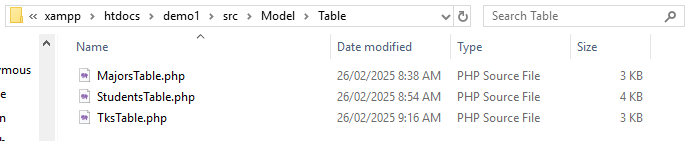


* Chú ý: Khi tạo CRUD trong cakephp có thể cần 1 khóa ID giả. Ở đây, chúng ta sẽ tạo 1 “khóa giả” id lấy dữ liệu từ majors.id và đổi tên lại là id để phù hợp với quy tắc nhận khóa trong cakephp. Trong 1 số trường hợp phức tạp, có thể bạn sẽ cần ghép chuỗi lại các khóa chính có tham gia trong view CSDL để tránh bị trùng dữ liệu khóa. Trường hợp ví dụ này, vì thống kê theo từng chuyên ngành nên mã chuyên ngành có thể là đủ để phân biệt tránh bị trùng dữ liệu.

*cake bake all tks*

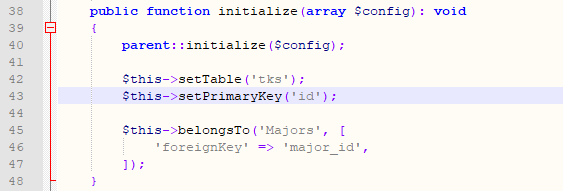


* Quan sát thông báo sau khi thực hiện lệnh, *nếu trong trường hợp cakephp vẫn chưa nhận được khóa* như trên thì ta sẽ tiến hành chỉnh sửa trong tập tin TksTable.php nằm trong thư mục **demo1/src/Model/Table**



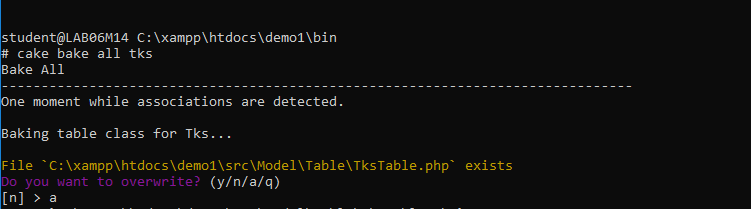
Trong phương thức *initialize*

Sau lệnh $this->setTable(“tks”);

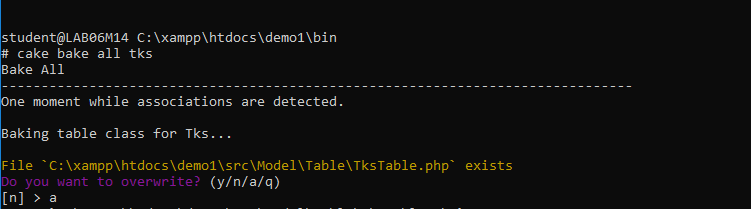


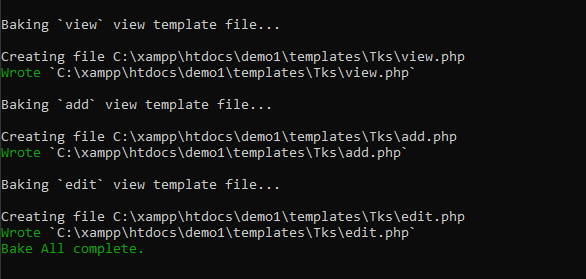
Sau đó mở CMD chúng ta tiến hành chạy lại:

*cake bake all tks*



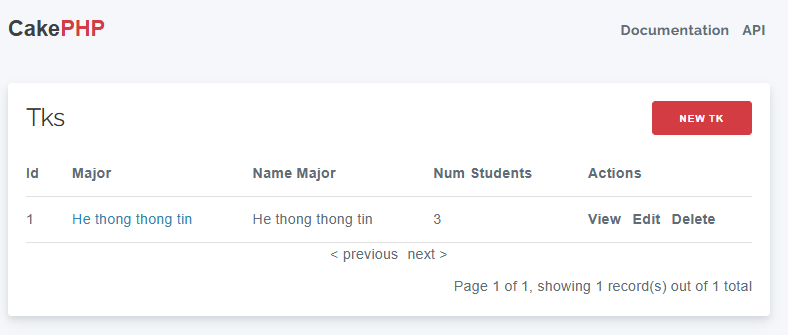
Chọn “a” để thay thế toàn bộ tập tin cũ





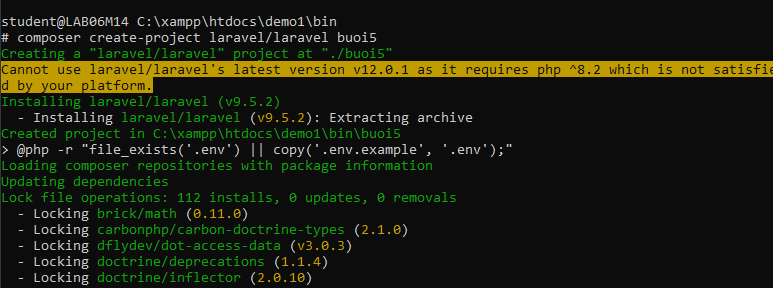
* Bây giờ chạy localhost/demo1/tks, ta sẽ được giao diện lấy dữ liệu từ view CSDL đã tạo:

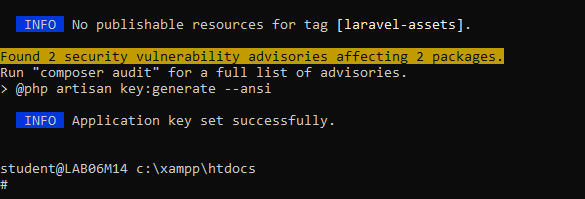




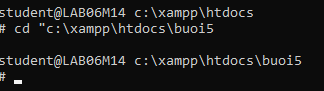
* Chú ý: Thường dữ liệu từ view CSDL sẽ tự động cập nhật khi ta thêm dữ liệu có liên quan, nên khi triển khai thực tế chúng ta nên bỏ những link dẫn đến các hành động Add/Edit/Delete trên “bảng” Tks ở trang index của tks.
  1. **Tạo project Laravel và kết nối CSDL**
* Trước tiên bạn cần tạo một project mới với lệnh composer, vd: chúng ta tạo 1 project buoi5

*composer create-project laravel/laravel buoi5*

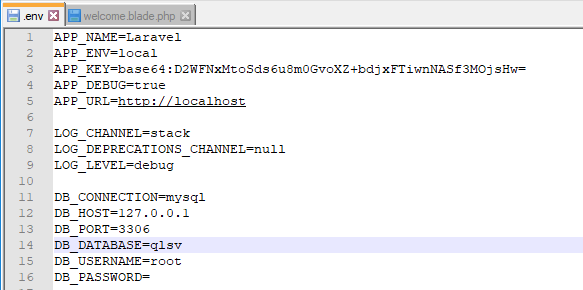




* Sau khi tạo xong project Laravel, các bạn cd vào trong thư mục project đã tạo:



Chú ý là chúng ta sẽ cần đặt tên csdl trong .env



* Chèn đoạn code sau

<?php

$results = DB::select('select \* from students', array(1));

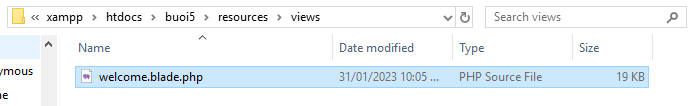
print\_r($results);

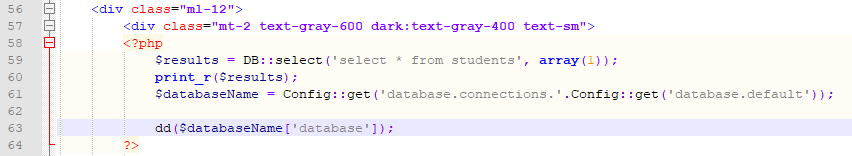
$databaseName = Config::get('database.connections.'.Config::get('database.default'));

dd($databaseName['database']);

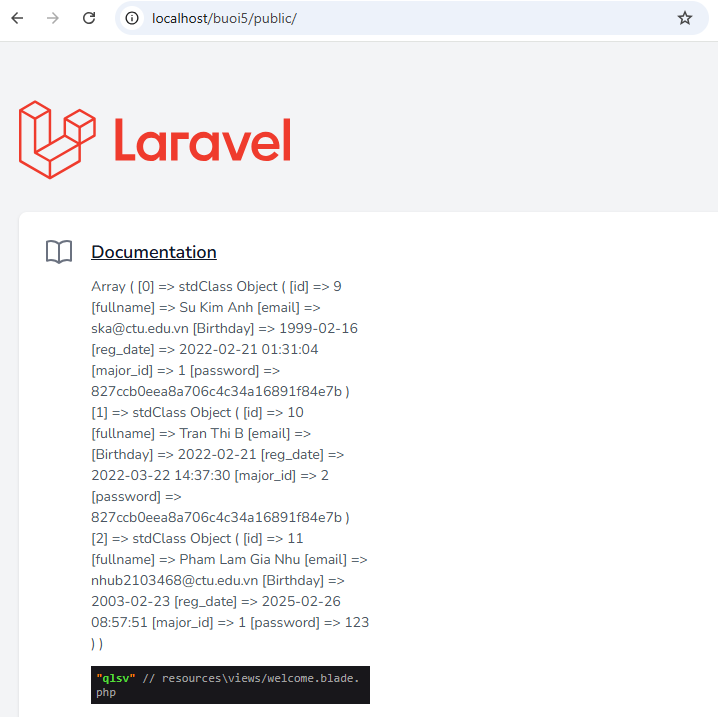
?>

để kiểm tra việc kết nối CSDL trong





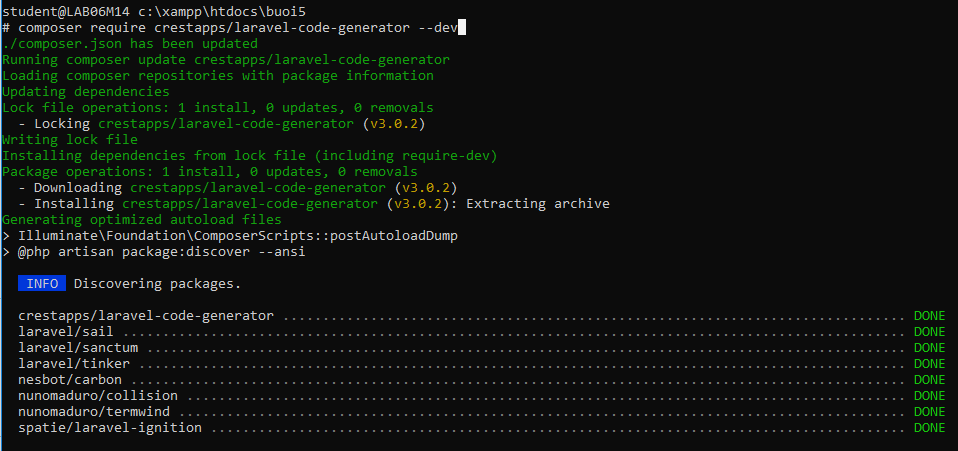
* Kết quả hiển thị:



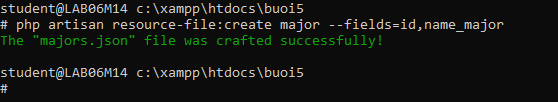
* 1. **Cài đặt CrestApps/laravel-code-generator**
* Chúng ta thực hiện lệnh sau

composer require crestapps/laravel-code-generator –dev

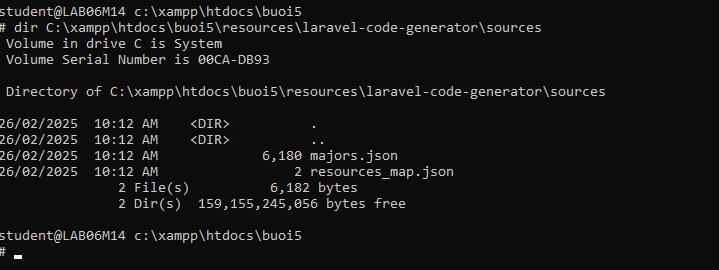
* Để đưa gói này vào project vừa tạo



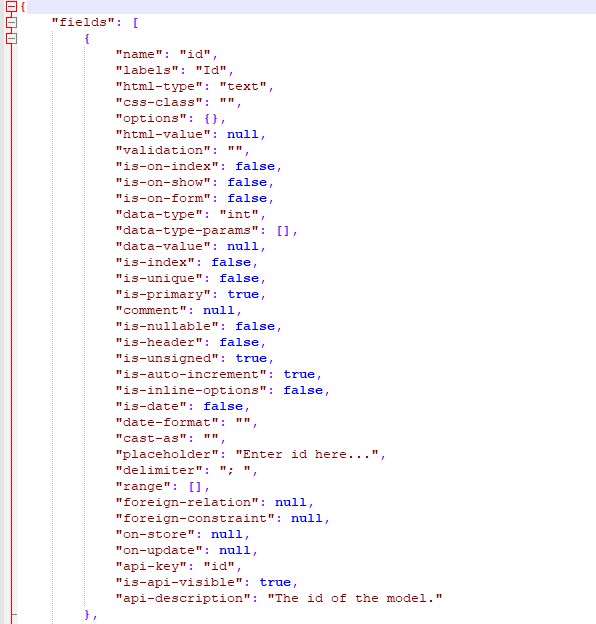
* 1. **Tạo các script mô tả để sinh ra code**



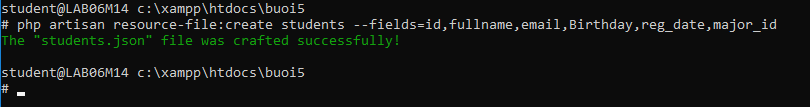
* Ta sẽ thấy tập tin majors.json được tạo ra ở:

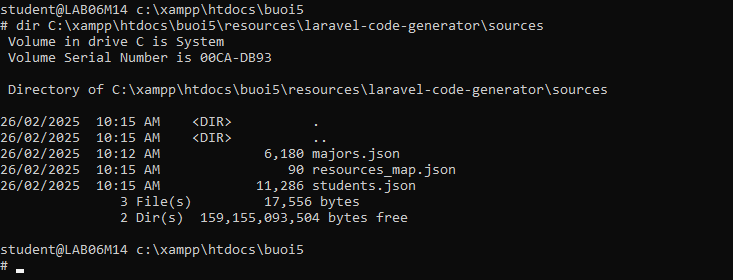


* Nội dung trong tập tin là:



* Tương tự ta tạo tập tin cho student

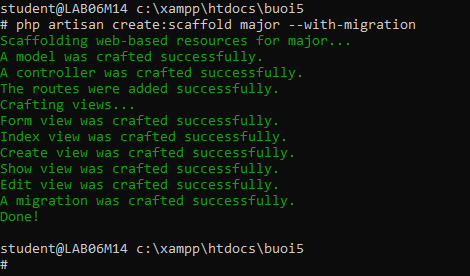






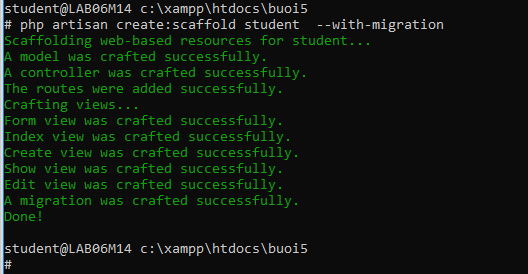
* 1. **Thực hiện tạo CRUD cho bảng Major**

php artisan create:scaffold major --with-migration



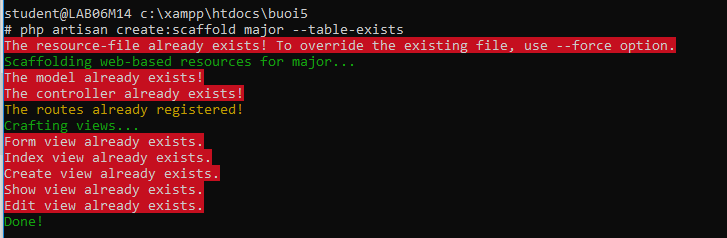
Đối với bảng Student

php artisan create:scaffold student --with-migration

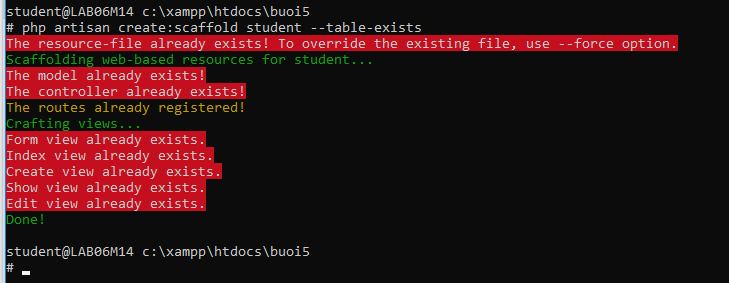


Hoặc tạo từ CSDL

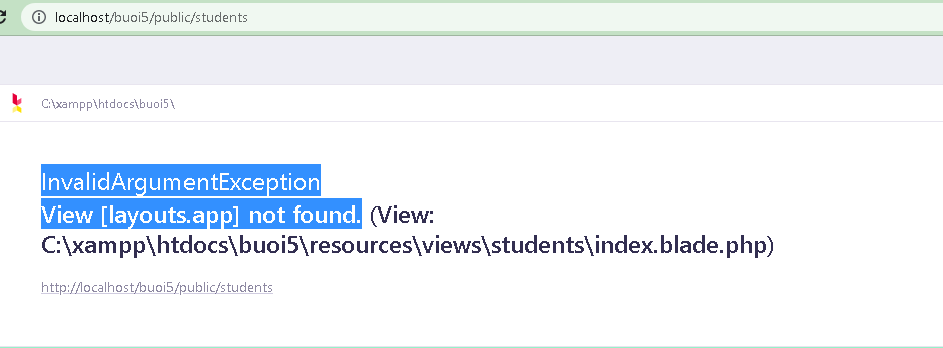
php artisan create:scaffold major --table-exists



php artisan create:scaffold student --table-exists

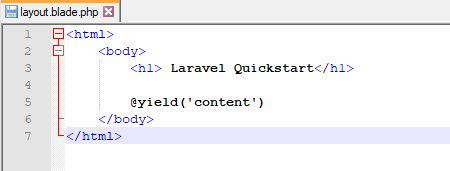


* Một số lỗi cần chú ý:



* Giải pháp:

Code của tập tin layout.blade.php trên sẽ đặt ở vị trí sau **buoi5/resources/views/**



Tham khảo thêm ở:

<https://stackoverflow.com/questions/48836709/view-layouts-app-not-found-laravel-framework-5-4-36>

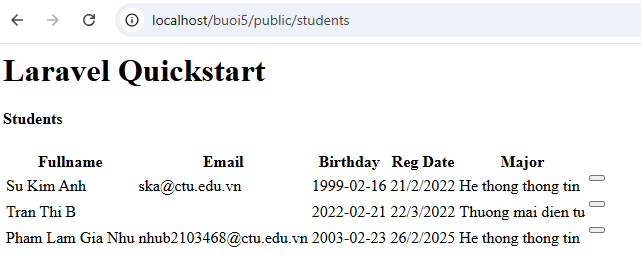
Các bạn vào buoi5/resources/views/students/ mở tập tin index.blade.php

🡪Sửa dòng

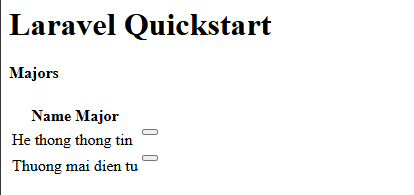
@extends('layouts.app') thành như hình:



Mở lại trang ta sẽ thấy:



Nếu gặp lỗi tương tự cho majors các bạn có thể sửa lại tương tự như students



* 1. **Chú ý quy tắt đặt tên bảng khi dùng laravel**

### Important Naming Convention[[1]](#footnote-0)

Laravel-Code-Generator strive to generate highly readable, and error free code. In order to keep your code readable, it is important to follow a good naming convention when chosing names for your models, fields, tables, relations and so on. Here is a list of recommendation that we belive is important to keep your code clean and highly readable.

1. Since each model represents a single object/row in a list/database, naming the model should be written in singular-form while using [Studly Case](https://laravel.com/docs/5.5/helpers#method-studly-case). For example, Post and PostCategory...
2. Since a database is a collection of model's object, table naming should always be plural and written in lowercase while using [Snake Case](https://en.wikipedia.org/wiki/Snake_case). For example, users, post\_categories...
3. Primary keys should be named id in the table.
4. Since the foreign key represents a foreign/other table, the name should always end with \_id. For example, post\_id, user\_id, post\_category\_id...
5. Field naming should always be in a singular-form and written in lowercase while using [Snake Case](https://en.wikipedia.org/wiki/Snake_case). For example, title, first\_name, description...
6. **Đọc và tìm hiểu về Cakephp tại đây: [[2]](#footnote-1), bạn hãy cho biết Cakephp là gì? Hãy mô tả các quy tắc trong các Layer Model, View, Controller của Cakephp. Tìm hiểu thêm trên Internet, các bạn trình bày Cakephp ra đời năm nào? Tác giả, lịch sử phát triển các phiên bản.**

* **CakePHP** là một framework phát triển web được xây dựng trên ngôn ngữ lập trình PHP. Nó cung cấp một cơ sở hạ tầng phát triển web linh hoạt và hiệu quả, giúp phát triển ứng dụng web nhanh chóng và dễ dàng bảo trì.
* **Mô tả ngắn về quy tắc trong các layer Model, View, Controller của CakePHP:**

1. **Model Layer:**

* **Trách nhiệm:** Đại diện cho dữ liệu và logic xử lý dữ liệu.
* **Quy tắc:** Sử dụng các lớp Model để tương tác với cơ sở dữ liệu, xử lý logic liên quan đến dữ liệu, và thực hiện các thao tác CRUD (Create, Read, Update, Delete).

1. **View Layer:**

* **Trách nhiệm:** Đại diện cho phần giao diện người dùng (UI).
* **Quy tắc**: Sử dụng các file View để hiển thị dữ liệu và tương tác với người dùng. Các view thường được thiết kế để chứa HTML và các thành phần giao diện.

1. **Controller Layer:**

* **Trách nhiệm:** Điều khiển luồng điều hướng của ứng dụng và làm trung gian giữa Model và View.
* **Quy tắc:** Sử dụng các lớp Controller để xử lý các yêu cầu từ người dùng, tương tác với Model để lấy dữ liệu, và chuyển dữ liệu đến các View để hiển thị cho người dùng.
* **Về lịch sử phát triển của CakePHP:**
* **Ngày ra mắt:** CakePHP ra đời chính thức vào tháng 4 năm 2005.
* **Tác giả:** Là Cake Software Foundation.
* **Lịch sử phiên bản:** CakePHP đã trải qua nhiều phiên bản với cải tiến và bảo trì. Các phiên bản quan trọng nhất có thể được theo dõi trên trang chính thức của CakePHP hoặc trong kho lưu trữ GitHub của dự án.

1. **Dựa vào [[3]](#footnote-2) hãy cho biết các quy tắc đặt tên các bảng trong CSDL để Cakephp có thể nhận dạng được hỗ trợ cho việc sinh CRUD tự động.**

- **Bảng chính (Main Table):**

+ Tên bảng chính được đặt theo quy tắc CamelCase và số ít.

+ Ví dụ: nếu model là User, thì bảng chính là users.

* **Khóa chính (Primary Key):**

+ Khóa chính của bảng nên được đặt tên là **id**. CakePHP tự động kết nối model với bảng theo tên khóa chính là id.

* **Bảng liên quan (Related Table):**

+ Bảng liên quan thường sử dụng hợp âm của tên bảng chính, theo quy tắc CamelCase và số ít.

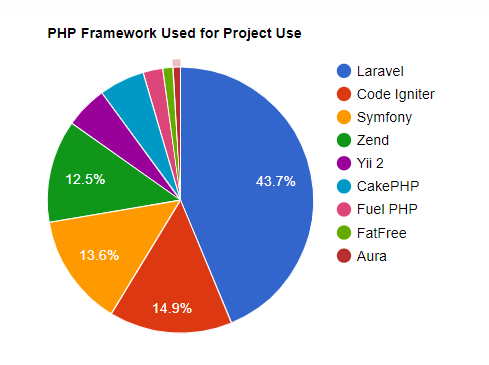
+ Ví dụ: nếu có bảng users, bảng liên quan có thể là user\_profiles hoặc user\_details.

* **Khóa ngoại (Foreign Key):**

+ Khóa ngoại thường được đặt tên theo quy tắc CamelCase và số ít, và có thể kết hợp với tên bảng chính.

+ Ví dụ: nếu có bảng users, thì khóa ngoại có thể là user\_id.

1. **Hãy tìm và trình bày các số liệu mới nhất về người sử dụng Cakephp và Laravel mà bạn tìm thấy được trên Internet.**



* Theo như số liệu trên: Người sử dụng Laravel chiếm **43.7%** và sử dụng CakePHP chiếm số lượng ít hơn **10%.**



* Số lượng sao trên GITHUB.

1. **Hãy phân tích điểm mạnh và điểm yếu của Cakephp và Laravel, tiến hành so sánh 2 framework này.**

**-CakePHP:**

- **Điểm Mạnh:**

**1. Simplicity and Convention Over Configuration (Cấu hình theo quy ước):** CakePHP hướng tới việc giảm độ phức tạp bằng cách sử dụng quy ước mặc định, giúp giảm thiểu cấu hình và làm cho việc phát triển trở nên nhanh chóng.

**2. Thư viện tích hợp (Batteries Included):** CakePHP đi kèm với nhiều thư viện và công cụ sẵn có, giúp giảm thiểu việc phải triển khai nhiều thứ từ bên ngoài.

**3. Sự ổn định:** Framework này có chu kỳ phát triển ổn định, giúp đảm bảo tính ổn định và tương thích ngược với các phiên bản trước đó.

- **Điểm Yếu:**

**1. Cộng đồng nhỏ hơn:** So với Laravel, CakePHP có một cộng đồng nhỏ hơn, điều này có thể làm cho việc tìm kiếm giải pháp và tài liệu hỗ trợ ít thuận tiện hơn.

**2. Nhận thức thấp hơn:** Mặc dù có những cải tiến đáng kể, nhưng một số người lập trình vẫn coi CakePHP là ít phổ biến và ít linh hoạt hơn so với một số framework khác

**- Laravel:**

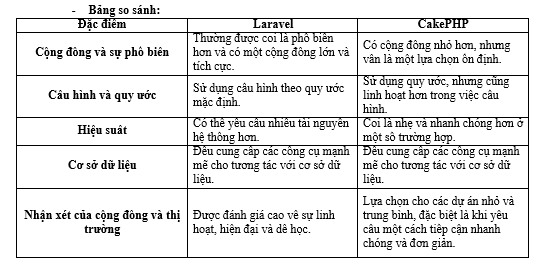
- **Điểm Mạnh:**

1. **Eloquent ORM và Migration:** Laravel cung cấp Eloquent ORM, giúp tương tác với cơ sở dữ liệu trở nên dễ dàng và hiệu quả. Migration giúp quản lý cấu trúc cơ sở dữ liệu.
2. **Lập trình hướng đối tượng mạnh mẽ:** Laravel sử dụng ngôn ngữ PHP mạnh mẽ và sử dụng tính năng lập trình hướng đối tượng một cách hiệu quả.
3. **Blade Template Engine:** Blade cung cấp một cú pháp rõ ràng và mạnh mẽ cho việc quản lý giao diện người dùng.

* **Điểm Yếu:**

1. **Đòi hỏi tài nguyên hệ thống cao:** Laravel có thể đòi hỏi tài nguyên hệ thống cao hơn so với một số framework khác.

2**. Tính linh hoạt và độ phức tạp:** Mặc dù có những tính năng mạnh mẽ, nhưng đôi khi Laravel có thể trở nên quá phức tạp cho các dự án nhỏ và đòi hỏi nhiều cấu hình.



1. **Bạn hãy tạo 1 CSDL đơn giản để lưu trữ dữ liệu với các mô tả bên dưới:**

**Môn học (mã môn, tên môn, số tín chỉ, có tính điểm tích lũy)**

**Sinh viên (mã sinh viên, họ tên sinh viên, ngày sinh, email, giới tính, số điện thoại, mật khẩu)**

**Lớp môn học (Mã lớp môn học, mã môn, học kỳ niên khóa)**

**Điểm (mã lớp môn học, mã sinh viên, điểm)**

**Tên bảng, tên cột, kiểu dữ liệu các bạn tự đặt sao cho thỏa các quy tắc để các framework nhận dạng được, vd dùng quy tắc với Cakephp:[[4]](#footnote-3), có thể thêm các cột khóa giả (ID) nếu thấy thực sự cần thiết cho việc lập trình và tự động sinh mã. Tạo 1 số dữ liệu mẫu với dữ liệu là thông tin điểm số các môn học mà bạn đã học.**

**Đưa các lệnh SQL tạo CSDL, tạo bảng,... vào bài làm**

1. **Bạn lựa chọn hoặc dùng Cakephp hoặc dùng Laravel để tạo project và tự động sinh CRUD cho CSDL đã tạo ở câu 6.**
2. **Từ các giao diện đã tạo bạn sẽ thêm các bạn sinh viên cùng nhóm làm bài tập của bạn, và nhập thêm 1 số dòng dữ liệu nữa để đảm bảo có ít nhất 10 sinh viên có điểm trong hệ thống.**
3. **Tạo 1 khung nhìn (view CSDL) bằng câu lệnh SQL để tính điểm trung bình tích lũy của mỗi sinh viên, thông tin bao gồm: mã số sinh viên, họ tên, điểm tích lũy, số môn đã học, số môn đã tích lũy, tổng số tín chỉ đã tích lũy. Tên view CSDL nên đặt tên sao cho thỏa điều kiện để có thể tạo CRUD tự động.**
4. **Hãy tạo CRUD cho view CSDL đã tạo ở câu 9, quan sát và chụp hình kết quả. Vào controller chỉnh sửa sao cho trang index của CRUD đã được tạo hiển thị điểm tích lũy của các sinh viên sắp xếp giảm dần.**

**Chú ý:**

* Các bạn nộp file word: Quy tắc đặt tên file: **<mssv>-<hoten>-<bai><stt\_bai thực hành>.docx** nộp lên Classroom (VD: **B123456-NguyenVanA-bai1.docx**), kèm với các file khác được yêu cầu như phần câu hỏi đã nêu. **Ngoại trừ file word trả lời câu hỏi, các file còn lại các bạn nén vào 1 file zip**. File zip đặt tên như file word.
* Mỗi câu các bạn trả lời bằng hình hoặc dạng text tùy vào yêu cầu của câu hỏi và **TRẢ LỜI THEO ĐÚNG THỨ TỰ CÂU HỎI**. Nếu câu nào không trả lời được các bạn cứ để số thứ tự câu hỏi và bỏ trống phần trả lời.
* Các câu trả lời có tham khảo trên Internet phải trích dẫn link/nguồn.
* *Vi phạm 1 trong các điều sau đây bài thực hành sẽ bị 0 điểm:*
  + Đặt tên KHÔNG ĐÚNG quy tắc được yêu cầu.
  + Bài không đủ các thành phần (word, code+data (nếu có),...) đã được yêu cầu.
  + Bài không thực hiện đúng yêu cầu “**Ngoại trừ file word trả lời câu hỏi, các file còn lại các bạn nén vào 1 file .zip”**
  + Bị phát hiện copy, sao chép từ các bạn khác
  + Phần trả lời không ghi rõ trả lời cho câu nào
  + Thứ tự câu trả lời không đúng thứ tự câu hỏi

1. https://crestapps.com/laravel-code-generator/docs/2.2 [↑](#footnote-ref-0)
2. https://book.cakephp.org/4/en/intro.html [↑](#footnote-ref-1)
3. https://book.cakephp.org/4/en/intro/conventions.html [↑](#footnote-ref-2)
4. https://book.cakephp.org/4/en/intro/conventions.html [↑](#footnote-ref-3)