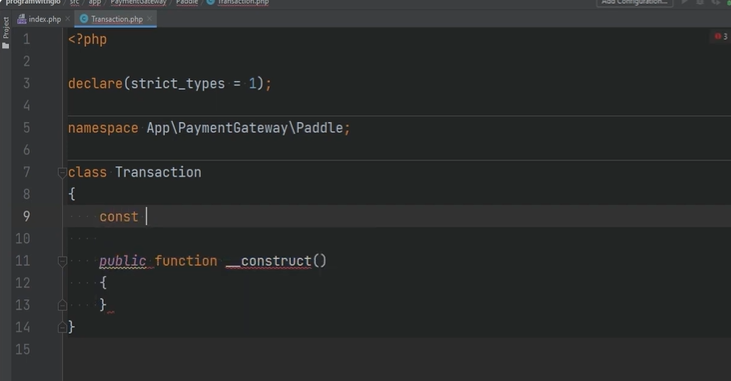
**2.6. PHP HƯỚNG ĐỐI TƯỢNG – HẰNG SỐ LỚP**

Đề cập đến các hằng số trong phần đầu tiên phần của khóa học này: các hằng số được xác định cho mỗi lớp và đó là những gì chúng ta sẽ đề cập đến video này cùng với phạm vi toán tử phân giải.

Hằng số là bất biến có nghĩa là bạn không thể thay đổi giá trị một khi nó được xác định, việc xác định các hằng số tương tự như tài sản nhưng chúng tôi làm theo tiêu chuẩn để giữ tất cả các tên không đổi viết hoa và sử dụng dấu gạch dưới làmdải phân cách vì vậy ví dụ nếu tôi đi đến lớp giao dịch từ trước bài học.

Xác định một số hằng số cho điều này lớp một giao dịch có thể có trạng thái giao dịch đúng có thể ở trạng thái thanh toán hoặc nó có thể ở trạng thái chờ xử lý hoặc nó có thể ở trong trạng thái từ chối, v.v.

Vậy hãy tạo một số trong số đó để tạo ra các hằng số chúng tôi sử dụng từ khóa const theo sau là tên hằng và for tên không đổi chúng tôi theo sau:

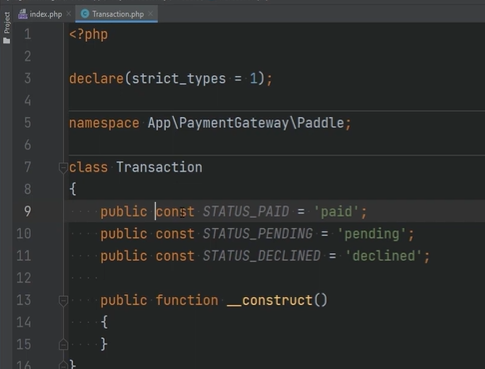


tiêu chuẩn và giữ chúng ở dạng chữ hoa và sử dụng dấu gạch dưới làm dấu phân cách chúng ta sẽ gọi nó là trạng thái đã thanh toán và với giá trị mà chúng tôi có thể đặt nó thành số nguyên hoặc chuỗi tùy thuộc vào cách bạn muốn lưu trữ và giữ những trạng thái này.

Trong trường hợp này tôi sẽ sử dụng tất cả giá trị chữ thường nên đã trả tiền cho việc này và sau đó chúng ta có thể sao chép cái này và nói trạng thái đang chờ xử lý và đặt trạng thái này thành đang chờ xử lý và chúng tôi đặt trạng thái bị từ chối bạn cũng có thể chỉ định công cụ sửa đổi quyền truy cập hoặc khả năng hiển thị của hằng số lớp nếu theo mặc định thì bạn không đặt chúng thành công khai nhưng tôi khuyến khích và giới thiệu để luôn xác định khả năng hiển thị ngay cả khi nó được công khai nên trong trường hợp này tôi sẽđặt chúng ở chế độ công khai.

Vậy bây giờ chúng ta đã xác định được các hằng số làm cách nào chúng tôi thực sự có thể truy cập chúng với tư cách là bạn?

Hãy nhớ khi chúng ta xác định các hằng số không thuộc lớp chúng tôi có thể truy cập chúng bằng cách chỉ cần giới thiệu hoặc sử dụng tên không đổi nhưng để truy cập



hằng số lớp chúng ta cần truy cập chúng sử dụng tên lớp vì vậy nếu bạn đi đến chỉ mục.php và hãy để tôi nhận xét điều này bây giờ chúng tacó thể sử dụng tên lớp đó là giao dịch và sau đó chúng tôi sử dụng gấp đôi các cột đó làđược gọi là toán tử phân giải phạm vi và sau đó là tên không đổi để chúng ta có thể bắt đầu như đã trả tiền và hãy lặp lại điều đó và nếu chúng tôi làm mới trang chúng tôi thấy các hằng số lớp được in trả phí được phân bổ một lần cho mỗi lớp chứ không phải cho mỗi trường hợp.

Điều đó có nghĩa là chúng ta không cần phải có thể hiện của lớp để truy cập hằng số lớp bây giờ ngoài truy cập các hằng số ở cấp độ lớp thực ra bạn có thể truy cập chúng ở cấp độ đối tượng vì vậy nếu tôi mang cái này trở lại đây nơi tôi có phiên bản giao dịch thay vì thực hiện giao dịch tôi có thể thực sự truy cập nó trên biến cách này và cách này vẫn hoạt động nên nếu tôi làm mới nó vẫn sẽ in trả phínếu bạn đặt hằng số lớp thành riêng tư bạn sẽ không thể truy cập được nữa.

Hằng số đó bên ngoài lớp học giống như bạn sẽ không thể truy cập vào lớp họcthuộc tính vì vậy hãy thay đổi điều này thành riêng tư và nếu tôi làm mới trang, chúng tôi sẽ nhận được lớp nguy hiểm tuy nhiên bạn có thể truy cập

A screenshot of a computer

Description automatically generated

họ trong lớp và để truy cập các hằng số trong lớp bạn cần sử dụng tên lớp đó là một cách để làm vậy thì hãy var bỏ đi và bây giờ hãy bình luận điều này hãy làm mới trang chúng ta được trả tiền và một cách khác để truy cập nó là sử dụng

A screenshot of a computer

Description automatically generated

2:57

từ khóa self và self đề cập đến lớp học hiện tại vì vậy nó tương tự như biến này đề cập đến đối tượng gọi nhưng tự đề cập đến lớp hiện tại hoặc lớp nơi nó được gọi là có nhiều hơn sự khác biệt giữa bản than và điều này và chúng ta sẽ nói về điều đó trong video tiếp theo khi chúng ta đến phần số liệu thống kê nhưng nếu tôi làm mới trang nó vẫn in ra một lớp đã trả tiền cũng có một hằng số ma thuật được gọi là lớp giải quyết khi biên dịch time và trả về một lớp đủ điều kiện đặt tên như vậy chẳng hạn.

ở đây chúng ta có thể thực hiện một giao dịch sử dụng toán tử phân giải phạm vi và sau đó sử dụng lớp và nếu bạn làm mới trang đó in tên lớp đủ điều kiện bây giờ điều này có thể hoạt động trên các đối tượng và bạn có thể cũng làm việc trên chính lớp đó vì vậy chúng ta có thể làm theo cách này và nó sẽ vẫn in tên lớp đủ điều kiện

A screenshot of a computer

Description automatically generated

3:42

có nhiều trường hợp sử dụng hằng số nếu bạn có dữ liệu hoặc thông tin không thay đổi và tiếp tục tham khảo toàn bộ mã đó tốt hơn là chuyển nó vào một hằng số lớp vậy nên theo cách đó nếu nó chỉ thay đổi bạn thay đổi nó ở một nơi trường hợp sử dụng khác là xác định hằng số như một kiểu liệt kê hoặc liệt kê giống như cách tôi vừa làm cho giao dịch nó cho phép bạn không cần viết mã cứng những thứ này giá trị nếu bạn nhận thấy rằng bạn khó khan mã hóa một số phần dữ liệu không thay đổi điều đó thường xuyên.

4:09

xem xét thay thế chúng bằng hằng số và tôi sẽ cho bạn xem một ví dụtạo một phương thức ở đây được gọi là trạng thái đã đặt và nó sẽ chấp nhận một sợi dây trạng thái và nó sẽ thiết lập thuộc tính tình trạng và hãy đơn giản trả lại cái này hãy thiết lập kiểu trả về bản thân và như bạn nhớ trước đây khi chúng tôi đặt kiểu trả về hiện tại lớp chúng tôi sử dụng tên lớp như thế này giao dịch nhưng bạn cũng có thể sử dụng từ khóa self trong đó đề cập đến lớp hiện tại

vì vậy nó vẫn hoạt động bây giờ chúng ta cần tạo thuộc tính trạng thái ngay như vậy hãy làm chuỗi riêng tư trạng thái và hãy đặt trạng thái mặc định đang chờ xử lý và để tôi nhận xét những điều này cho bây giờ để tôi có thể cho bạn xem cả hai có và không có hằng số bây giờ bạn có thể đặt trạng thái mặc định ngay tại đây cách này đang chờ xử lý hoặc bạn có thể đặt nó vào một nhà xây dựng, điều đó hoàn toàn phụ thuộc vào bạn

A screenshot of a computer

Description automatically generated

4:57

bạn muốn làm thế nào bây giờ hãy nói trạng thái thiết lập giao dịch và hãy thiết lập nó để trả tiền bây giờ hãy thực hiện giao dịch kết xuất và chúng tathấy trạng thái giao dịch được đặt thành được trả đúng, đó là tài sản riêng và giá trị nó được đặt thành.

bây giờ đã được thanh toán, tất cả đều tốt và nó đang làm việc nhưng hãy tưởng tượng làm việc này khắp nơi mã của bạn về cơ bản là mã hóa cứng những thứ này và bạn có thể mắc lỗi đánh máy và vượt qua một cái gì đó như thế này và nó vẫn sẽ hoạt động tốt và mặc dù điều này không gây ra bất kỳ lỗi nào.

5:24

bạn có thể giới thiệu một số lỗi ở đâu đó khác trong mã của bạn, bạn cũng có thể vượt qua bất kỳ trạng thái không hợp lệ nào và nó sẽ vẫn hoạt động bình thường và điều này có thể giới thiệu lỗi ở phần khác của bạn mã số đây là lúc các hằng số có thể giúp bạn nếu chúng ta quay lại đây chúng ta có những hằng số nà vì vậy thay vì mã hóa cứng dữ liệu này, chúng tôi có thể đơn giản là đặt trạng thái tự chờ xử lý và chỉ mục chúng tôi có thể chuyển giao dịch đã thanh toán ngay bây giờ nếu chúng tôi làm cho khỏe lại mọi thứ vẫn hoạt động và chúng tôi có đã giải quyết được một vấn đề.

vấn đề đã giải quyết là lỗi chính tảbây giờ chúng ta đang truyền hằng số nên chúng ta biết rằng chúng tôi sẽ không mắc lỗi chính tả tuy nhiên chúng tôi vẫn có thể chuyển bất kỳ thông tin không hợp lệ nào trạng thái ở đây và điều này vẫn hoạt động giải quyết trường hợp thứ hai này chúng ta có thể đơn giản

giới thiệu một số loại xác nhận trên đặt phương thức trạng thái để ở đây chúng ta có thể nếu trạng thái được thanh toán hoặc trạng thái đang chờ xử l hoặc trạng thái bị từ chối thì mọi thứ đều ổn

nếu không thì có thể ném một ngoại lệ hoặc đưa ra một số loại lỗi nhưng điều này sẽ có rất nhiều điều kiện ở đây đúng và như bạn sẽ bổ sung them hằng số và nhiều trạng thái hơn bạn sẽ cần sửa đổi phương pháp này thay vì thế này chúng ta có thể đơn giản giới thiệu một hằng số khác ở đây đã gọi tất cả các trạng thái và đặt trạng thái này thành mảng các phím có thể là trạng thái thực tế để chúng ta có thể tự trạng

và bạn thực sự có thể giới thiệu một số loại tên thân thiện với trạng thái này bởi vì mảng này bạn có thể thân thiện với nó tên mà sau này bạn có thể sử dụng để hiển thị

cho người dùng nên ở đây chúng ta có thể nóiđược trả bằng chữ cái đầu tiên viết hoa

chúng tôi có thể chờ xử lý và sau đó chúng tôi có thể là suy giảmtheo cách này, đây cũng là một cái gì đó được gọi làbảng tra cứu và chúng ta sẽ nói về việc tra cứu

các bảng sau trong khóa học và bây giờ trongphương pháp thiết lập trạng thái chúng ta có thể đơn giản kiểm tra xem trạng thái đã cho có tồn tại trong tất cả các mảng trạng thái và nếu không tồn tại thì chúng ta có thể ném một số loại ngoại lệ hoặc lỗi và chúng tôi sẽ đề cập đến các trường hợp ngoại lệ sau này trong khóa học vì vậ đừng lo lắng về điều đó bây giờ hãy chịu đựn tôi ở đây bây giờ nếu chúng ta làm mới trang chúng ta sẽ gặp lỗi này giao dịch không tồn tại bây giờ chúng ta có thể đi quay lại đây và sửa cái này và chuyển cái thích hợp

giao dịch ổn nên đã bán cả haivấn đề nhưng có một điều làm phiền tôi là chúng ta đang ghép nối những thứ này các hằng số với lớp giao dịch lớp giao dịch phải chịu trách nhiệm để xử lý và làm việc với giao dịch tôi không nghĩ nó nên như vậy chịu trách nhiệm lưu trữ các hằng số trạng thái thật vô nghĩa khi đề cập đến lớp giao dịch bên ngoài đối tượng để truy cập các hằng số hãy tưởng tượng chúng ta phải truy cập vào hằng số này một nơi khác nơi giao dịch\đối tượng không cần thiết có thể để hiển thị tham chiếu một lớp có logic kinh doanh khác trong đó chỉ để có đượccác giá trị không đổi sẽ trông không đẹp với tôi vậy chúng ta có thể làm gì chú ý cách chúng tôi đang chuẩn bị các hằng số với từ trạng thái đó là một dấu hiệu khá tốt cho thấy chúng ta có thể trích xuất các hằng số này thành của riêng nó lớp vì hiện tại chúng tôi đang thêm tiền tố tất cả các trạng thái có trạng thái

nhưng tôi nghĩ điều gì đó giống như trạng thái trả phí sẽ trông sạch sẽ và hơn thế nữa biểu cảm vì vậy hãy tạo lớp này trong thư mục ứng dụng và chúng tôi có thể lưu trữ cái này trong không gian tên enums ứng dụng và chúng ta có thể thực sự tạo thư mục và nó sẽ nằm trong ứng dụng enums danh mục.

vì vậy hãy thêm nó và bây giờ chúng ta có thể di chuyển bốn người này đến đây và chúng ta có thể thoát khỏi tiền tố trạng thái bây giờ chúng ta có thể tiến hành giao dịch và thay vào đó chúng ta có thể tham chiếu hằng số bằng cách sử dụng trạng thái để chúng tôi có thể chờ xử lý và thay đổi nó thành trạng thái và bây giờ nếu tôi làm mới trang mọi thứ vẫn hoạt động nhưng điều này có vẻ như dễ đọc hơn nhiều trông biểu cảm hơn và chúng tôi không kết đôi dữ liệu trạng thái của chúng tôi với giao

cả lớp cũng có điều tôi muốn cho bạn biết biết là php sẽ được thêm vào hỗ trợ riêng cho loại enum để bạn có th tạo các lớp enum để bạn không cần để tạo ra một lớp học như thế này nơi bạn đặt.

các hằng số bạn có thể tạo mộtlớp enum tôi sẽ để lại liên kết đến rfc trong phần mô tả và bạn có thể hãy kiểm tra nếu bạn quan tâm nhưng nó sẽ là một tính năng khá thú vị nó sẽ được thêm vào trong php 8.1

**2.7. THUỘC TÍNH VÀ PHƯƠNG THỨC TĨNH TRONG PHP HƯỚNG ĐỐI TƯỢNG**

Tâm trí về các biến tĩnh trong bài học này chúng ta sẽ nói chuyện về các thuộc tính và phương thức tĩnh của lớp vì vậy hãy thêm một thuộc tính tĩnh và một phương thức cho lớp giao dịch này bạn có thể xác định thuộc tính của mình và phương pháp tĩnh sử dụng từ khóa tĩnh vì vậy hãy thêm một đếm thuộc tính cho lớp giao dịch này và hãy làm cho nó tĩnh để chúng ta có thể làm công cộng số lượng tĩnh và chúng ta có thể đặt mặc định giá trị về 0 và hãy biến điều này thành một.

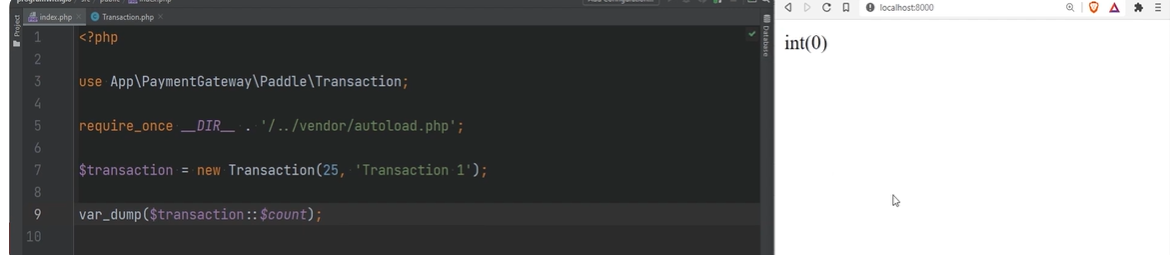
số nguyên bây giờ bạn có thể đặt tĩnh từ khóa trước công cụ sửa đổi truy cập hoặc công khai trong trường hợp này và nó vẫn hoạt động nhưng nếu bạn theo tiêu chuẩn bạn nên đặt nó

A screenshot of a computer program

Description automatically generated

sau công cụ sửa đổi truy cập nhưng nó đã hoạt động cho bạn miễn là bạn luôn kiên định bạn có thể truy cập các thuộc tính tĩnh và phương pháp giống như cách bạn có thể truy cập hằng số sử dụng toán tử phân giải phạm vi để

nếu chúng ta truy cập index.php chúng ta có thể thực hiện đối tượng giao dịch var dump toán tử phân giải phạm vi và số đếm nếu chúng tôi làm mới trang, chúng tôi sẽ nhận được số 0



vấn đề là bạn không thực sự cần đối tượng để truy cập các phương thức tĩnh và thuộc tính của lớp bởi vì chúng không được tạo ra và chúng không liên quan đến mỗi đối tượng chúng được tạo ra và là liên kết trên cơ sở lớp học để họ thuộc cho chính lớp học đó chứ không phải cho bất kỳ ai.

đối tượng cụ thể hoặc một thể hiện và ở đây như bạn có thể thấy điều này là hoàn hảo hợp lệ và nó sẽ hoạt động bạn có thể truy cập các thuộc tính và phương thức tĩnh trên đồ vật nhưng bạn không cần bạn có thể truy cập chúng trên lớp giống như cách bạn đang truy cập hằng số lớp

vì vậy nếu chúng ta làm mộtsố lượng giao dịch và chúng tôi làm mới trang chúng tôi nhận được số 0 bây giờ tôi đã đề cập rằng chúng không được liên kết với nhau phản đối điều đó có nghĩa là các thuộc tính của lớp thông thường như số lượng và mô tả trong trường hợp này mỗi khi bạn tạo một phiên bản này thuộc tính được tạo và nhận liên quan đến cụ thể đó đối tượng để bạn có thể có nhiều giao dịch khác nhau.

đối tượng và mỗi người có thể có khác nhau giá trị cho các thuộc tính đó bạn có thể nghĩ về các thuộc tính tĩnh như các biến toàn cục theo cách mà giá trị của chúng được chia sẻ trên các đối tượng bởi vì chúng không gắn liền với một điều cụ thể đối

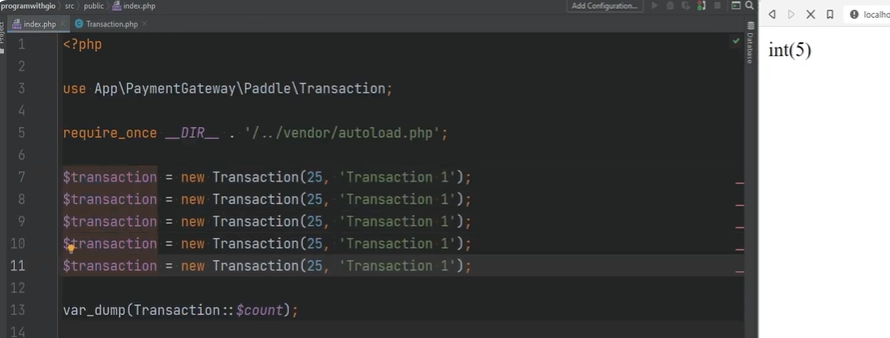
chính lớp học đó vậy nên hãy để tôi chỉ cho bạn một ví dụ về điều đó hãy tăng lên

đếm mỗi lần có phiên bản mới được tạo khi tham khảo thuộc tính hoặc phương thức tĩnh từ bên trong lớp bạn có thể sử dụng cho chính mình từ khóa hoặc bạn có thể sử dụng tên lớp cũng như vậy chúng ta có thể tự đếm và chúng ta có thể đơn giản làm cộng cộng bây giờ nếu chúng talàm mới trang như bạn nhận thấy nó đã thay đổi từ 0 thành 1

A screen shot of a computer

Description automatically generated

bây giờ một cái nếu chúng ta nhân đôi cái này thì nói là 5



lần và chúng tôi làm mới trang khi bạn có thể xem bây giờ nó đang in năm nếu chúng tôi nhận xét tất cả chúng và chúng tôi không có đối tượng nào bây giờ nó sẽ bằng 0, đó là giá trị mặc định để bạn có thể thấy giá trị được chia sẻ và đó là loại có thể truy cập toàn cầu mà bạn không cần một thể hiện của lớp bạn có thể chỉ cần truy cập chúng bằng cách sử dụng lớp chính nó bây giờ hãy thay đổi công cụ sửa đổi truy cập thành riêng tư ở đây và bây giờ thuộc tính này chỉ có thể truy cập được trong lớp và nó không còn có thể truy cập được bên ngoài lớp đó vì vậy nếu bạn làm mới trang

chúng ta sắp gặp phải một lỗi nghiêm trọng cần khắc phục.

A screenshot of a computer

Description automatically generated

điều này chúng ta cần phải thêm một phương thức tĩnh để chúng tôi có thể làm một cái gì đó như được đếm và chúng ta có thểtạo phương thức này với tĩnh từ khóa và chỉ cần trả lại tự đếm nếu chúng tôi làm mới trang mọi thứ vẫn hoạt động nếu tôi đi phía trước và bỏ ghi chú những điều này nó vẫn sẽ in năm vì truy cập các phương pháp tĩnh không yêu cầu một thể hiện của lớp và bởi vì họ thuộc về lớp chính nó chứ không phải một đối tượng cụ thể biến.

cái này không có sẵn trong tĩnh phương pháp để bạn không thể làm điều gì đó như lặp lại số tiền này bạn sẽ nhận được lỗi nghiêm trọng và có ý nghĩa đúng không bởi vì biến này đề cập đến gọi đối tượng nhưng trong trường hợp số liệu thống kê chúng tôi không thực sự có phản đối vì vậy thay vì sử dụng cái này biến bạn sẽ phải sử dụng bản than từ khóa và có hai từ khóa khác bạn có thể sử dụng như cha mẹ và tĩnh và chúng ta sẽ nói về những điều đó sau tất nhiên một khi chúng ta có được quyền thừa kế và liên kết tĩnh muộn nên đừng lo lắng về hai từ khóa đó ngay bây giờ tôi cũng không muốn làm bạn choáng ngợp nhiều thông tin khác nữa bạn

nên nhận thức.

nếu bạn cố gắng sử dụng phạm vitoán tử phân giải trên một không tĩnh tài sản bạn cũng sẽ nhận được một lỗi nên ví dụ chúng tôi có số tiề và các thuộc tính mô tả ở đây là công cộng nếu chúng ta truy cập index.php và chúng ta đã làm như vậy cái gì đó như toán tử phân giải phạm vi giao dịch

số tiền này sẽ không hiệu quả, hãy để tôi xóa bốn giao dịch khác và nếu tôi làm mới trang, chúng tôi sẽ nhận được lỗi nghiêm trọng vì thuộc tính tĩn số lượng không tồn tại chúng tôi chỉ có số lượng tài sản không tĩnh và bạnkhông thể truy cập các thuộc tính không tĩnh tĩnh và áp dụng tương tự cho các phương thứcnếu bạn cố gắng truy cập một phương thức không tĩnh tĩnh bạn cũng sẽ nhận được mộtlỗi kể từ php8 và trước phva bạn sẽ nhận được một tin nhắn không được dùng nữa vì vậy nếu tôi thay đổi điều này để xử lý đây là một phương pháp không tĩnh và tôi làm mới trang chúng tôi nhận được cái chết lỗi vì tôi đang dùng php8 nhưng trước đó

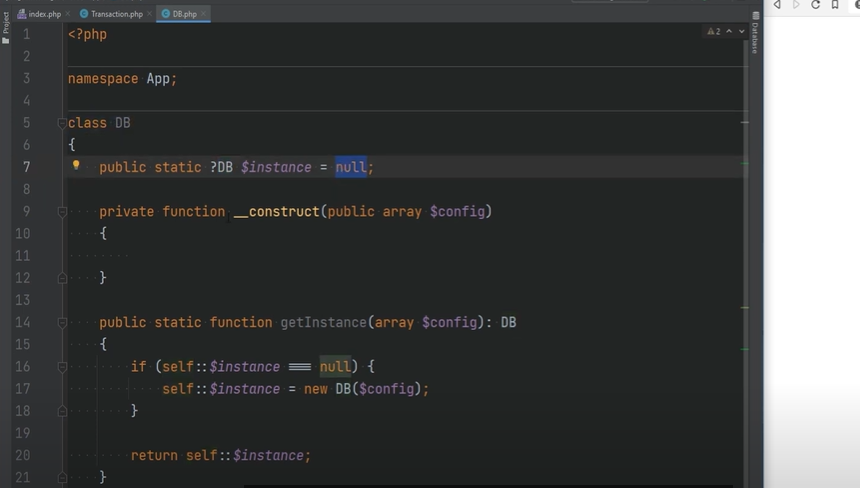
php8 bạn sẽ nhận được tin nhắn không còn được dùng nữa rằng chúng tôi đã đề cập đến những điều cơ bản của số liệu thống kê chúng ta hãy thực sự nói về trường hợp sử dụng cho các thuộc tính tĩnh và phương pháp sử dụng các phương pháp tĩnh và tài sản thường được coi là xấuthực hành nhưng có một số trường hợp sử dụng nơi việc sử dụng số liệu thống kê có thể phát huy tác dụng một trường hợp sử dụng tiện dụng cho các thuộc tính tĩnh là có một loại máy đếm như tôi đã làm

trong ví dụ này hoặc để lưu trữ các giá trị để nó thường có thể được được sử dụng để thực hiện một kỹ thuật được gọi là ghi nhớ có thể tăng tốc hoạt động tốn kém bởi lưu trữ kết quả để truy cập sau này và tôi đã trình bày một ví dụ về điều đó trong biến tĩnh bài học có liên kết trong mô tả và bạn có thể kiểm tra nó nếu bạn muốn bạn cũng có thể đến trên một mã sử dụng tĩnh

các thuộc tính để thực hiện một giai điệu duy nhất mẫu chúng ta sẽ nói về các mẫu thiết kế sau trong khóa học nhưng mẫu đơn duy trì một thể hiện duy nhất của một lớp nhất định trong suốt quá trình triển khai ứng dụng của bạn, tôi sẽ trình bày bạn là người đơn giản nhất ví dụ về mẫu đơn thực hiện sử dụng các thuộc tính và phương thức tĩnh vì vậy tôi có một lớp db được tạo ở đây và đây là cơ bản nhất

\*ví dụ về mẫu singleton bây giờ điều này không hẳn là hoàn toàn đơn lẻ bởi vì nó không chứa những thứ như nhân bản và các phương pháp ma thuật khác vàlý dochính cho điều đó là bởi vì chúng tôi chưa đề cập đến những phương pháp kỳ diệu đó và

tôi quyết định không đưa chúng vào đây nhưng đây là cái cơ bản nhất và tôi sẽ

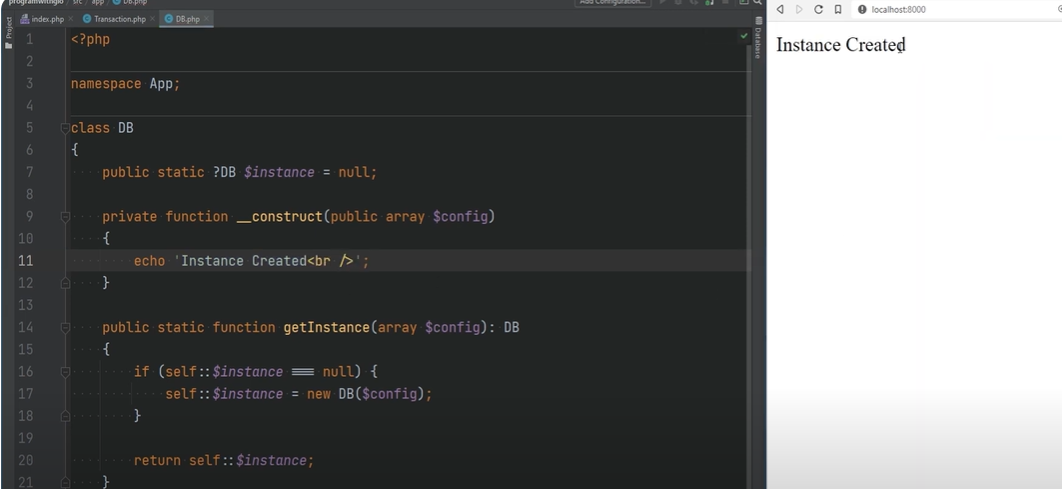


giải thích đây là gì để chúng ta có một tĩnh trường hợp thuộc tính đó là nullable nhưng loại của nó là hiện hành lớp db và nó được đặt thành null theo mặc định sau đó trong hàm tạo, chúng tôi chấp nhận cấu hình hoặc có thể thông tin xác thực db hoặc một cái gì đó như thế và đó là quyền tài sản tài sản không tĩnh và lưu ý rằng quyền truy cập của hàm tạo công cụ sửa đổi được đặt ở chế độ riêng tư, điều này có nghĩa là rằng chúng ta không thể trực tiếp khởi tạo điều này

lớp học nếu tôi đi vào index.php và tôi đã cố gắng lấy ví dụ của db bằng cách đơn giản khởi tạo lớp db điều này sẽ không hoạt động nếu tôi làm mới trang chúng ta sẽ nhận được một lỗi nghiêm trọng và do đó chúng tôi có ví dụ này lấy phương thức đó là một phương thức tĩnh và chúng tôi có thể gọi điều này để lấy ví dụ của db lớp này một lần nữa chấp nhận tương tự đối số làm hàm tạo và kiểm tra

nếu phiên bản được đặt thành null nếu nó được đặt thành null nó sẽ đặt thuộc tính đó thành phiên bản mới của db và sau đó chỉ cần trả về lớp đó ví dụ như ở đây chẳng hạn nếu chúng ta đã làm db lấy ví dụ và chúng tôi đã vượt qua một số config ở đây cho dù chúng tôi có bao nhiêu lần gọi cái đó get instance nó vẫn sẽ nhận được giống nhau

ví dụ và tôi có thể chỉ cho bạn điều đódễ dàng nếu tôi truy cập db.php tôi có thể đơn giản tiếng vọng ví dụ được tạo bằng một dòng ngắt và nếu tôi làm mới trang chúng tôi thấy rằng nó chỉ được in một lần

tuy nhiên nếu chúng ta thay đổi điều này thành công khai và thay vì phiên bản db chúng tôi đã làm cơ sở dữ liệu mới để tạo phiên bản mới của db lớp học theo cách này nếu tôi làm mới trang chúng ta thấy rằng nó được in năm lần nhưng trong trường hợp khác khi chúng ta có được dụ và khi chúng tôi đặt cái này thành riêng tư chúng tôi chỉ nhận được một Ví dụ tôi chỉ cho bạn xem cái này bởi vì bạn sẽ gặp một số mã điều đó có thể thực hiện mẫu đơn cách này.

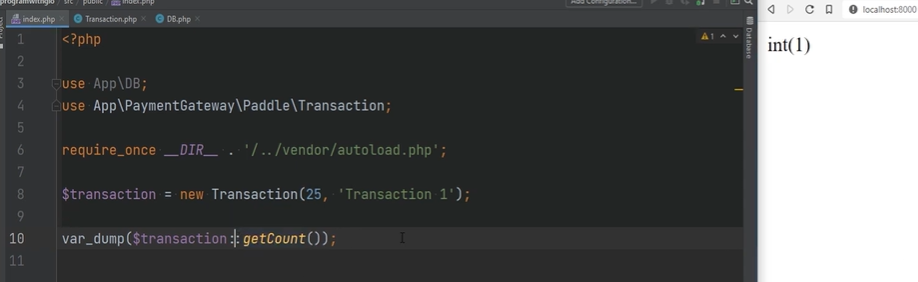
và điều đó không có nghĩa đó là cách đúng đắn tiêm phụ thuộc là cách đúng đắn và chúng ta sẽ nói về nó chi tiết hơn và về singleton mẫu ở phần sau của khóa học này vì vậy đừng lo lắng về nó ngay bây giờ trường hợp sử dụng khác cho các phương thức tĩnh sẽ là tạo ra một loại tiện ích nào đó những phương pháp không thực sự cần thiết

ví dụ của lớp nên có thể bạn đã c một số loại lớp định dạng có một loạt các phương pháp để định dạng tiền tệ của bạnhoặc định dạng số tiền hoặc định dạng giá trị phần trăm, v.v.

nếu chúng ta muốn nhà máy xử lý khởi tạo và tạo ra cái này đối tượng chúng ta có thể làm một cái gì đó như giao dịch tương đương với nhà máy giao dịch thực hiện và chuyển các đối số cần thiết và phương thức tạo tĩnh này sẽ là chịu trách nhiệm tạo giao dịch đối tượng đừng lo lắng về điều này bây giờ

chúng ta sẽ đề cập đến mô hình nhà máytrong khóa học này và điều này không đúng hoặc là cách tốt hơn là sử dụng lại tính năng tiêm phụ thuộc nhưng tôi sẽ chỉ cho bạn cách thích hợp để thực hiện mô hình nhà máy không sử dụng các phương pháp tĩnh

đề cập và chỉ cho bạn một vài điều một trong số chúng là bạn có thể truy cập các phương thức tĩnh không tĩnh, ví dụ như vậy nếu chúng tôi thực hiện var dump phạm vi giao dịch toán tử phân giải đếm cái này sẽ hoạt động đúng cái này in một vì getcount là tĩnh phương pháp bạn cũng có thể sử dụng toán tử đối tượng

 để truy cập phương thức tĩnh đó và nó vẫn sẽ hoạt động mặc dù tôi không khuyên bạn nên làm điều này bởi vì không rõ ràng rằng phương pháp này là tĩnh và hơn nữa bạn không thể truy cập biến này trong phương thức này nên nó có thể gây ra sự nhầm lẫn và vấn đề con một điều khác tôi muốn đề cập là rằng bạn có thể có các cuộc gọi lại tĩnh hoặc đóng cửa tĩnh chẳng hạn

nếu chúng ta có thứ gì đó như bản đồ mảng v chúng tôi đã chuyển lệnh gọi lại ở đây rồi chức năng và chúng tôi đã thông qua một số loại mảng và bạn làm bất cứ điều gì bạn cần làm trong chức năng bản đồ mảng này bạn có thể đặt chức năng này ở chế độ tĩnh và lý do chính tại sao bạn sẽ thấy ai đó làm điều này là bởi vì khi bạn thiết lập chức năng tĩnh bạn không còn có thể

truy cập vào biến này vì vậy nếu chúng ta di chuyển cái này bên trong lớp học như lớp giao dịch chẳng hạn nếu chúng ta có thể đã làm điều này ở đâu đó trong phương pháp xử lý khi bạn xác định gọi lại hoặc đóng ở dạng tĩnh bạn không thể truy cập cái này được nữacó thể thay đổi nên số tiền này sẽ không hoạt động nên nếu chúng ta làm điều gì đó như thanh bãi rác

tối ưu hóa tôi sử dụng các bao đóng tĩnh khi tôi muốn chắc chắn rằng biến này không có sẵn và có thể truy cập được trong chức năng đó, đây là nó dành cho việc này

