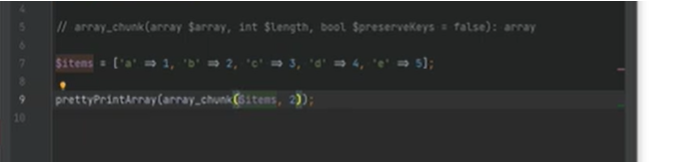
**Video 1.26: How to work with Array in PHP**



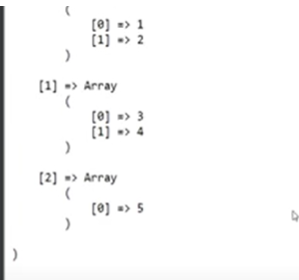
**1. PHP has a lot of built in array functions to work with arrays. And you've already seen me use some of these array functions in previous lessons. But in this lesson, let's go over some of the other important functions that I think will come in handy later and you will see me use from time to time. I will also leave the link to this page in the description where you can check the rest of the built in array functions.As you can see, there are a lot of them.**

(PHP có rất nhiều hàm mảng dựng sẵn để làm việc với mảng và bạn đã thấy tôi sử dụng một số hàm mảng này trong các bài học trước nhưng trong bài học này chúng ta hãy điểm qua một số hàm quan trọng khác mà tôi nghĩ sẽ có ích sau này và thỉnh thoảng bạn sẽ thấy tôi sử dụng tôi cũng sẽ để lại liên kết đến trang này trong phần mô tả, nơi bạn có thể kiểm tra phần còn lại của các hàm mảng tích hợp sẵn vì bạn có thể thấy có rất nhiều hàm trong số đó.)



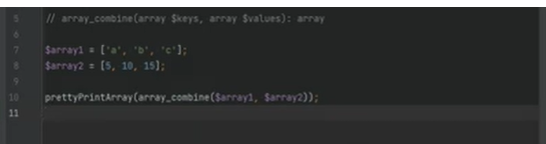
**- First we have array chunk. Sometimes you might want to split an array into chunks of arrays with specified length. The first argument that array chunk accepts is the array. The second argument is length. And then the third argument is optional, which is whether you want to preserve keys or not. So let's see what happens. As you notice, I'm using here a pre print array function, and this is a custom function that I built and created for this demonstration. And I put that under the helpers. php.**

(Đầu tiên, đôi khi chúng tôi có đoạn mảng, bạn có thể muốn chia một mảng thành các phần của mảng có độ dài xác định. Đối số đầu tiên mà mảng chấp nhận là mảng đối số. Thứ hai là độ dài và sau đó đối số thứ ba là tùy chọn, đó là liệu bạn có muốn giữ lại khóa hay không. Vì vậy hãy xem điều gì xảy ra khi bạn nhận thấy ở đây tôi đang sử dụng một hàm mảng in khá đẹp và đây là một hàm tùy chỉnh mà tôi đã xây dựng và tạo cho phần demo này. Và tôi đặt nó dưới helpers.php )



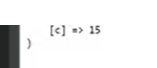
**- So if I go into it, it simply just echoes out the pre tags and prints the array. So if we refresh the page, we see the array of size.three and each of that array contains two elements from the original array. Now, as you notice, the keys were not preserved, right? Because this is zero one and we have A and B in here. If you want to preserve the keys, you need to pass in the third argument as true, and now if we refresh the page, we get A, B.C, D, and E.**

(Nên nếu tôi đi sâu vào nó, nó chỉ đơn giản là lặp lại phần trước -tags và in mảng. Vì vậy nếu chúng ta làm mới trang ,chúng ta sẽ thấy mảng có kích thước ba và mỗi mảng đó chứa hai phần tử từ mảng ban đầu khi bạn nhận thấy các khóa không được giữ nguyên . Bởi vì, đây là số 0 và chúng ta có A và B ở đây. Nếu bạn muốn giữ lại các khóa bạn cần truyền trong đối số thứ ba là đúng ,và bây giờ nếu bạn làm mới trang, chúng ta sẽ nhận được A, B, C, D và E )



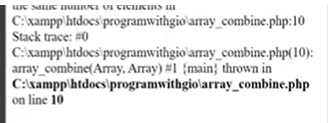
**- Next, we have array combine. This function creates an array from the given keys and values. It will throw an array if the number of elements in the first array don't match the number of elements in the second array. So basically, it will use the values from the array one as the keys, and it will use the values from the array two as the values, and it will create the new array.If we refresh the page here.**

(Tiếp theo chúng ta có mảng kết hợp, hàm này sẽ tạo một mảng từ các khóa đã cho và đánh giá.Nó sẽ báo lỗi nếu số phần tử trong mảng đầu tiên không khớp với số phần tử trong mảng thứ hai. Nên về cơ bản nó sẽ sử dụng các giá trị từ mảng 1 làm khóa và nó sẽ sử dụng các giá trị từ mảng 2 làm khóa các giá trị và nó sẽ tạo mảng mới nếu chúng ta làm mới trang ở đây.)



**- We are getting an array where the keys are a, b, and c, and values are 5, 10, and 15. If size did not match, you would throw an error.**

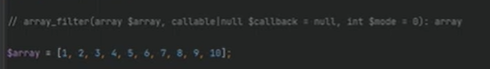
(Chúng ta sẽ nhận được một mảng trong đó các khóa là b và c và các giá trị là 5 10 và 15. Nếu kích thước không khớp, bạn sẽ gặp lỗi.)



**- So for example, if I add the value d. Here and I refresh, we're going to get an error. Note that prior to PHP 8, this would throw just a warning and return false instead. But in PHP 8, most internal functions now throw an error instead of the warning when parameter validation fails.**

(Ví dụ như nếu tôi thêm giá trị d vào đây và tôi làm mới, chúng ta sẽ nhận được một lưu ý lỗi rằng trước php 8, điều này sẽ chỉ đưa ra một cảnh báo và thay vào đó trả về false

nhưng trong php 8, hầu hết các hàm nội bộ giờ đây đều đưa ra lỗi thay vì cảnh báo khi xác thực tham số thất bại.)



- **Next, we have array filter. Array filter iterates over each array value and it passes the value to the given callback. If the return of that callback is true, then the value is returned into the resulting array.Otherwise, the element will be discarded. For example, let's say that we wanted to filter out the odd numbers from this array and only keep the even numbers**

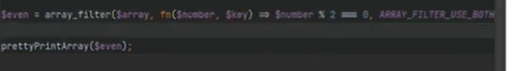
(Tiếp theo, chúng ta có bộ lọc mảng. Bộ lọc mảng lặp lại từng giá trị mảng và nó chuyển giá trị cho lệnh gọi lại đã cho. Nếu kết quả trả về của lệnh gọi lại đó là đúng thì giá trị được trả về mảng kết quả.Nếu không phần tử sẽ bị loại bỏ, ví dụ như giả sử rằng chúng tôi muốn lọc ra các số lẻ từ mảng này và chỉ giữ lại các số chẵn)

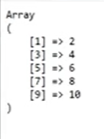


**- We could create a variable called even and call array filter. Pass the array as the first argument, and then the second argument is a callable or a callback function, and we could just inline that using the short arrow function syntax. And by default the array value is passed to the callback as the argument, so we could accept the number here, and then here we can check if number is divisible by two, then keep that in the array, otherwise filter it out.**

(Chúng tôi có thể tạo một biến gọi là số chẵn và gọi bộ lọc mảng. Truyền mảng làm đối số đầu tiên và sau đó đối số thứ hai là hàm có thể gọi hoặc hàm gọi lại và chúng tôi có thể chỉ nội tuyến bằng cách sử dụng cú pháp hàm hàng ngắn hơn

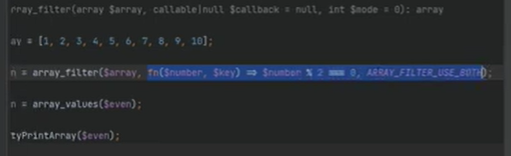
Và theo mặc định, giá trị mảng được chuyển đến hàm gọi lại làm đối số để chúng ta thể chấp nhận số ở đây và sau đó ở đây chúng ta có thể kiểm tra xem số có chia hết cho hai hay không, sau đó giữ số đó trong mảng của chúng, nếu không thì lọc )





**- So if we create a print that array now. We refresh the page. We only see the even numbers in the list. As mentioned before, by default, the value is passed to the callback as the argument, but it could change that by specifying the mode flag as the third parameter, so you could specify array filter use key, and now instead of number.Being passed as the value, you will be passed as the key, or you could use the array filter use both. And now the first argument is the value and the second argument is the key.**

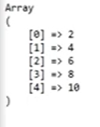
(Vì vậy, nếu chúng ta tạo một bản in mảng đó bây giờ, chúng ta làm mới trang, chúng ta chỉ thấy các số chẵn trong danh sách như đã đề cập trước đó, theo mặc định, giá trị được chuyển đến hàm gọi lại làm đối số nhưng nó có thể thay đổi điều đó bằng cách chỉ định cờ chế độ là tham số thứ ba để bạn có thể chỉ định khóa sử dụng bộ lọc mảng và bây giờ thay vì số được truyền làm giá trị, bạn sẽ được chuyển làm khóa hoặc bạn có thể sử dụng bộ lọc mảng sử dụng cả hai và bây giờ đối số đầu tiên là giá trị và đối số thứ hai là Chìa khóa)



**- Notice how the keys of these elements have gaps in between them. That's because array keys are not preserved. So it might result in gaps like this if the array is numerically indexed.**

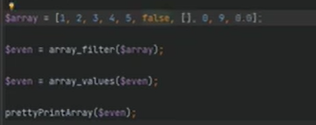
**+) If you want to re-index an array after filtering it, you need to use a function called array underscore values, which will simply return all values of array indexed numerically.**

(Lưu ý rằng các khóa của các phần tử này có các khoảng trống ở giữa chúng, đó là do các khóa của mảng không được giữ nguyên nên có thể dẫn đến các khoảng trống như thế này nếu mảng được lập chỉ mục bằng số nếu bạn muốn lập chỉ mục lại mảng sau khi lọc, bạn cần sử dụng một hàm được gọi là các giá trị gạch dưới mảng sẽ đơn giản trả về tất cả các giá trị của mảng được lập chỉ mục bằng số.)



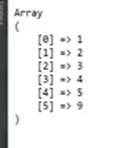
**- And now if we print, it's indexed correctly. Note that the callback is optional, and if you don't pass a callback, then it will just filter out all falsy values from the array.**

(Và bây giờ nếu chúng ta in nó được lập chỉ mục chính xác, hãy lưu ý rằng lệnh gọi lại là tùy chọn và nếu bạn không chuyển lệnh gọi lại thì nó sẽ chỉ lọc ra tất cả các giá trị sai từ mảng.)



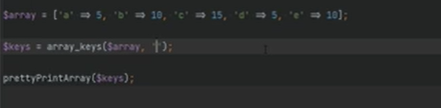
**- So for example, if we remove this, and we change some of the values to falsy values, like 0, false, empty array, 0. 0, and so on.**

(Ví dụ: nếu chúng tôi loại bỏ phần này và chúng tôi thay đổi một số giá trị thành giá trị falsie như zero false mảng trống 0,0, v.v)



**- If we refresh the page, Those falsy values will be filtered out from the array.**

(Nếu chúng tôi loại bỏ phần này và chúng tôi thay đổi một số giá trị thành giá trị falsie như zero false mảng trống 0,0, v.v)



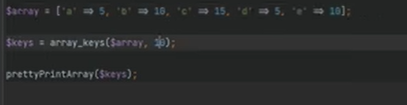
**- Similar to array values, which returns the values of the arrays, we could use array keys to get the keys of the array. You could also search for specific values and only return the keys for the values that match. For example, we could create a variable called keys and set that to array keys, array, and create**

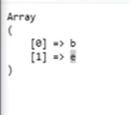
(Nếu chúng tôi làm mới trang thì các giá trị falsie đó sẽ được lọc ra khỏi mảng các khóa mảng để lấy các khóa của mảng, bạn cũng có thể tìm kiếm các giá trị cụ thể và chỉ trả về các khóa cho các giá trị khớp, ví dụ: chúng ta có thể tạo một biến có tên là khóa và đặt biến đó thành mảng khóa và đẹp các phím in)



**- And if we refresh, we get A, B, C, D, E**

(Và nếu làm mới, chúng ta sẽ nhận được a b c d)

.



- **If we specify the search value here and say that we want to search for 10, it's going to return B and E because that's where the value 10 is found.**

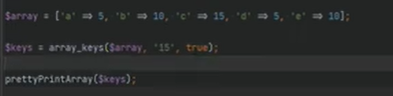
(Nếu chúng ta chỉ định giá trị tìm kiếm ở đây và nói rằng chúng ta muốn tìm kiếm 10 sẽ trả về b và e vì đó là nơi tìm thấy giá trị 10)





**- If we search for 15, it will return C.**

(Nếu chúng ta tìm kiếm 15 thì nó sẽ trả về c)





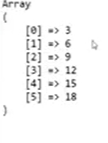
**- Note that it does the loose comparison by default, but it could do the strict comparison bypassing the true as the third argument. So we could do true. This will still return C, but if we set this to quotes, it will no longer. finder value**

(Lưu ý rằng nó thực hiện so sánh lỏng lẻo theo mặc định nhưng nó có thể thực hiện so sánh chặt chẽ bằng cách chuyển giá trị đúng làm đối số thứ ba để chúng ta có thể làm đúng, điều này vẫn sẽ trả về c nhưng nếu chúng ta đặt giá trị này thành dấu ngoặc kép thì nó sẽ không còn tìm thấy)



- **ArrayMap is one of the other functions that you will probably use more often, like ArrayFilter. It simply applies the callback to each element of the given arrays. For example, let's say we had an array of numbers, and we want to multiply each of these values by 3. We could do that by using ArrayMap and pass the callback here, which takes the argument of value, and then we could say number times 3, and the second argument is array. And then we can pretty print array.**

(Bản đồ mảng giá trị của chúng ta nữa một trong những hàm khác mà bạn có thể sẽ sử dụng thường xuyên hơn như bộ lọc mảng, nó chỉ đơn giản áp dụng lệnh gọi lại cho từng phần tử của mảng đã cho, ví dụ: giả sử chúng ta có một mảng số và chúng ta muốn nhân từng giá trị này với ba thì chúng ta có thể làm điều đó bằng cách sử dụng sơ đồ mảng và chuyển lệnh gọi lại ở đây lấy đối số là giá trị và sau đó chúng ta có thể nói số nhân ba và đối số thứ hai là mảng, sau đó chúng ta có thể tạo một mảng in)



**- And if we refresh the page, we get 3, 6,12 15 and 18.**

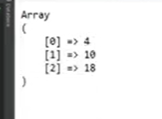
(Và nếu làm mới trang, chúng ta sẽ nhận được 3 6 9 12 15 và 18)



- **Now let me show you an example with multiple arrays, where we also have custom keys. So let's say we set the keys here, like that. Let's remove the rest, let's put a second array. And let's pass array one and array two. We're getting the number one and number two.**

**+) And now instead of multiplying by three, we can simply multiply number one by number two. So this should do one times four, two times five, and three times six.**

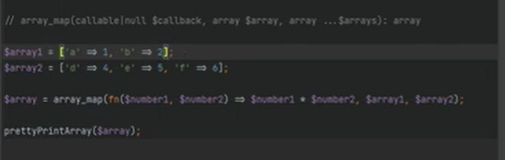
(Bây giờ hãy để tôi chỉ cho bạn một ví dụ với nhiều mảng trong đó chúng ta cũng có các khóa tùy chỉnh vì vậy giả sử chúng ta đặt các khóa ở đây như thế hãy xóa phần còn lại hãy đặt một mảng thứ hai và chuyển mảng một và mảng hai chúng ta có lấy số một và số hai và bây giờ thay vì nhân với ba chúng ta có thể chỉ cần nhân số một với số hai vì vậy điều này sẽ thực hiện một nhân bốn hai nhân năm và ba nhân sáu)

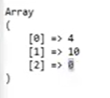


- **If we refresh the page, we get the expected result. But note that now array has been re indexed numerically and it no longer has the string keys.**

**+) That's because we're passing more than one array. Note that if you're passing only a single array to the ArrayMap function, it will preserve the keys. However, if you're passing more than one arrays, then it will re index the array numerically. Also note that when doing ArrayMap on multiple arrays, it is best to have arrays of the same length because the callback will be applied in parallel.**

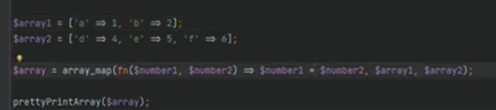
(Nếu chúng ta làm mới trang, chúng ta sẽ nhận được kết quả mong đợi nhưng lưu ý rằng bây giờ mảng đã được lập chỉ mục lại bằng số và nó không còn có các khóa chuỗi nữa vì chúng tôi đang chuyển nhiều hơn một mảng. Lưu ý rằng nếu bạn chỉ chuyển một mảng duy nhất cho hàm bản đồ mảng thì nó sẽ giữ nguyên các khóa, tuy nhiên nếu bạn đang truyền nhiều hơn một mảng thì nó sẽ lập chỉ mục lại mảng đó bằng số. Ngoài ra, hãy lưu ý rằng khi thực hiện ánh xạ mảng trên nhiều mảng, tốt nhất nên có một cuộc đua có cùng độ dài vì lệnh gọi lại sẽ được áp dụng song song )

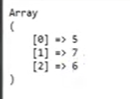




**- If you had uneven length, then the shorter array will be extended with the empty elements. So in this case, zero will be multiplied by six for the third element, and we're going to get zero**

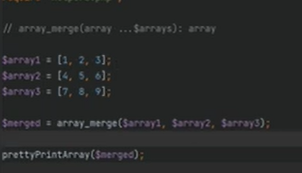
(Nếu khi đó bạn có độ dài không đồng đều mảng ngắn hơn sẽ được mở rộng với các phần tử trống nên trong trường hợp này 0 sẽ được nhân với 6 cho phần tử thứ ba và chúng ta sẽ nhận được 0)





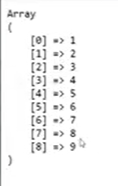
- **If we did addition instead, we would get six. You could also pass null as the callback, and what that will do is that it will simply build up an array from the given arrays.**

(Nếu chúng ta thực hiện phép cộng thay vào đó chúng ta sẽ nhận được 6. bạn cũng có thể chuyển null thành gọi lại và những gì họ sẽ làm là nó sẽ chỉ đơn giản xây dựng một mảng từ các mảng đã cho.)



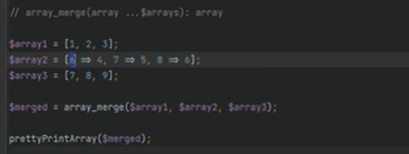
**- Next we have array merge that allows you to merge multiple arrays. If arrays have the same numeric keys then it will not overwrite the values but instead they will be appended. Also numeric keys will be re indexed starting from zero. So for example in this case I have three arrays where I don't specify the keys it's just the values and when I merge**

(Tiếp theo, chúng tôi có tính năng hợp nhất mảng cho phép bạn hợp nhất nhiều mảng nếu các mảng có cùng khóa số thì nó sẽ không ghi đè các giá trị mà thay vào đó chúng sẽ như vậy các khóa số cũng được nối thêm sẽ được lập chỉ mục lại bắt đầu từ 0, vì vậy, ví dụ trong trường hợp này, tôi có ba mảng trong đó tôi không chỉ định các khóa, đó chỉ là các giá trị và khi tôi hợp nhất)



**- And I refresh it will simply append at the end and we're going to get all the items from all the arrays. Even though this key is zero and this is also zero, it is not going to be overwritten.**

(Và làm mới, nó sẽ chỉ thêm vào cuối và chúng tôi 'sẽ lấy các mục cũ từ tất cả các mảng mặc dù khóa này là 0 và đây cũng là 0, nó sẽ không bị ghi đè )



**- Even if I, for example, had this as the key six or index six and had this index seven and had this index eight, when we refresh the page, they will be reordered. If however, you have the same string keys, then the later value will overwrite the previous one.**

(Ngay cả khi ví dụ như tôi có khóa này là khóa 6 hoặc chỉ mục 6 và có chỉ số 7 này và có chỉ số này chỉ mục 8 khi chúng tôi làm mới trang, chúng sẽ được sắp xếp lại nếu tuy nhiên bạn có cùng khóa chuỗi thì giá trị sau sẽ ghi đè lên giá trị trước đó)



**- So for example, if instead of six here, we had. A, and here we had B, and here we had C. Now when we refresh, we see A, B, C here, but if we added another one here, like B equals to 10, now this B is going to override this B, and instead of five, it will equal to 10. So if we refresh, now B is equal to 10**

(Ví dụ: nếu thay vì 6 ở đây chúng tôi có a và ở đây chúng tôi có b và ở đây chúng tôi có c bây giờ Khi làm mới, chúng ta thấy abc ở đây nhưng nếu chúng ta thêm một số khác ở đây như b bằng 10 thì b này sẽ ghi đè b này và thay vì 5 thì nó sẽ bằng 10 vì vậy nếu chúng ta làm mới bây giờ thì b bằng 10)

.

- **Another really useful function is array reduce, which will reduce the array to a single value using the callback function that you passed. So for example, let's say we had this invoice items array and we wanted to run some calculations where we multiply the price by quantity and sum it all up to get the total invoice price. Now of course you could do that many other ways, but one way is to use array reduce.**

(Thực sự là một cái khác Hàm hữu ích là giảm mảng sẽ giảm mảng xuống một giá trị duy nhất bằng cách sử dụng hàm gọi lại mà bạn đã chuyển, ví dụ: giả sử chúng tôi có mảng các mục hóa đơn này và chúng tôi muốn chạy một số phép tính trong đó chúng tôi nhân giá với số lượng và tính tổng tất cả Tất nhiên bây giờ để có được tổng giá hóa đơn, bạn có thể làm điều đó theo nhiều cách khác nhưng có một cách là sử dụng mảng giảm)



- **So we can call total equals array reduce. And the first argument is the array, the second argument is the callback, and the first argument that's passed into the callback is the return value of the previous iteration. So we can call that sum, and the second argument that's passed is the current iteration value, so we can call this item. And then within here we can do item quantity times item price, and we need to add all this to the sum for the previous iteration.**

(Để chúng ta có thể gọi tổng mảng giảm bằng và đối số đầu tiên là mảng đối số thứ hai là hàm gọi lại và đối số đầu tiên được chuyển vào lệnh gọi lại là giá trị trả về của lần lặp trước đó nên chúng ta có thể gọi tổng đó và đối số thứ hai được chuyển là giá trị lặp lại hiện tại để chúng ta có thể gọi mục này và sau đó ở đây chúng ta có thể tính số lượng mặt hàng nhân với giá mặt hàng và chúng ta cần thêm tất cả những thứ này vào tổng hoặc lần lặp trước đó)

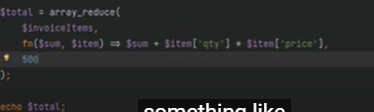
**- And on each iteration whatever the value we return from here gets passed as the first argument onto the next. iteration. Initially on the first iteration, so when we're at this iteration, right, the sum will equal to nothing or null, but you could customize that by passing the third argument, which is the initial value, and you could start off initial value with some other value instead of the null.**

(Và trên mỗi lần lặp, bất kỳ giá trị nào chúng ta trả về từ đây sẽ được chuyển làm đối số đầu tiên cho lần lặp tiếp theo ban đầu ở lần lặp đầu tiên, vì vậy khi chúng ta ở lần lặp này thì tổng sẽ bằng không hoặc null nhưng bạn có thể tùy chỉnh điều đó bằng cách chuyển đối số thứ ba là giá trị ban đầu và bạn có thể bắt đầu giá trị ban đầu bằng một số giá trị khác thay vì null)



**- So let's do it without the initial value first. Let's echo out total and we refresh the page. We get 258 Point nine, and that seems accurate.**

(Vì vậy hãy thực hiện mà không có giá trị ban đầu trước tiên, hãy tính tổng và chúng tôi làm mới trang mà chúng tôi nhận được 258 và điều đó có vẻ chính xác)



- **Now, you could also start off the initial value with something like 500. So if we refresh, it's going to be 758.9.**

(Vì vậy hãy thực hiện mà không có giá trị ban đầu trước tiên, hãy tính tổng và chúng tôi làm mới trang mà chúng tôi nhận được 258 và điều đó có vẻ chính xác)



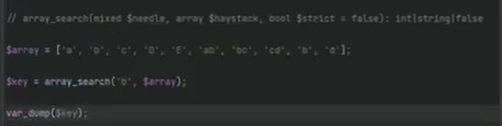
- **You could search array by using array search, which will return the key of the first matching value. So let's take this array as an example, and let's say we want to find the key of the letter B.**

(Bạn có thể tìm kiếm mảng bằng cách sử dụng tìm kiếm gạch dưới mảng sẽ trả về khóa đầu tiên giá trị khớp vì vậy hãy lấy mảng này làm ví dụ và giả sử chúng ta muốn tìm khóa của chữ b)



- **We can do key equals array search, and the first argument is the needle. And in our case, that's B. And second argument is the haystack. And in our case, that's array. And the third argument is whether you want to do the strict comparison or lose.**

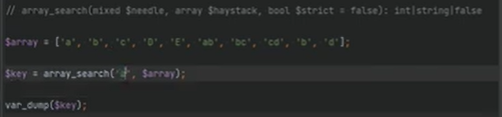
(Chúng ta có thể thực hiện tìm kiếm bằng khóa và đối số đầu tiên là cái kim và trong trường hợp của chúng ta đó là b và đối số thứ hai là đống cỏ khô và trong trường hợp của chúng tôi đó là mảng và đối số thứ ba là liệu bạn muốn thực hiện so sánh nghiêm ngặt hay thua)





**- And let's var dump key. We refresh and we get 1. Note the letter B also appears right here towards the end of the array, but it only returns the key of the first matching value. And in this case, that's 1. The search is also case sensitive. So searching for uppercase D will print 3 and not 9.**

(Và hãy làm mới khóa var\_dump và chúng tôi nhận được một lưu ý rằng chữ cái b cũng xuất hiện ngay tại đây ở cuối mảng nhưng nó chỉ trả về khóa của giá trị khớp đầu tiên và trong trường hợp này, tìm kiếm cũng phân biệt chữ hoa chữ thường nên việc tìm kiếm chữ hoa d sẽ in ra 3 chứ không phải 9)





**- For example. So if we search for uppercase D, we're going to get. three, and not the key for this lowercase d here. Array search will return false if it cannot find the value, but it can return zero if it finds the matching value at the zeroth position. So you should be careful when you're doing comparisons. You should always use the strict comparison.**

(Chẳng hạn, vì vậy nếu chúng ta tìm kiếm chữ hoa d, chúng ta sẽ nhận được ba chứ không phải khóa cho chữ thường d ở đây tìm kiếm mảng sẽ trả về sai nếu bạn không thể tìm thấy giá trị nhưng nó có thể trả về 0 nếu tìm thấy giá trị khớp ở vị trí thứ 0, vì vậy bạn nên cẩn thận khi thực hiện so sánh)





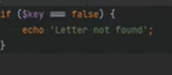
**- For example, if we were searching for lowercase a, this returns zero, right?**

(Ví dụ như bạn phải luôn sử dụng so sánh nghiêm ngặt nếu chúng ta đang tìm kiếm chữ thường a thì cái này trả về 0 đúng)



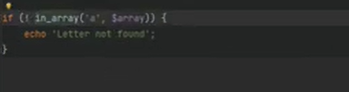
- **But if we did something like if key equals false with double equal and we printed letter not found, we refresh the page, we get letter not found, but the letter a is actually found. So this is what I mean by always using strict comparison.**

(Nhưng nếu chúng ta làm điều gì đó như nếu khóa bằng sai với gấp đôi bằng nhau và chúng ta in ra chữ cái không tìm thấy, chúng ta làm mới trang, chúng ta không tìm thấy chữ cái nhưng chữ cái a thực sự được tìm thấy nên đây là những gì ý tôi là bằng cách luôn sử dụng phép so sánh nghiêm ngặt)



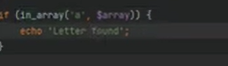
**- You need to do triple equals here, and if we refresh now, we get nothing printed because letter a is found. Alternatively, you could also use function called in\_array to check if the value exists in an array, and instead of returning the key, it will just return boolean true or false.**

(Bạn cần thực hiện gấp ba lần bằng ở đây và nếu làm mới bây giờ, chúng tôi sẽ không in được gì vì chữ cái a được tìm thấy, bạn cũng có thể sử dụng hàm được gọi trong mảng để kiểm tra xem giá trị có tồn tại trong một mảng hay không và thay vì trả về khóa nó sẽ trả về boolean true hoặc false)



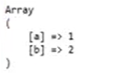
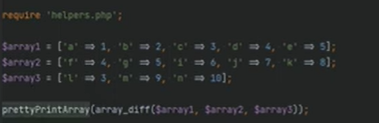
- **So you could do something like in array and first argument is the middle. So in our case, that's letter A and the haystack array. And in this case we will do if not in array. So if we refresh the page, nothing is printed.**

(Nên bạn có thể làm gì đó như trong mảng và đối số đầu tiên là cái kim nên trong trường hợp của chúng ta đó là chữ cái a và mảng cỏ khô và trong trường hợp này chúng ta sẽ làm nếu không có trong mảng nên nếu chúng ta làm mới trang này không được in ra )



**- But if we do in array and we change this to letter found, we refresh the page, we get letter found.**

(Nếu chúng ta in theo mảng và thay đổi thành chữ cái đã tìm thấy, chúng ta làm mới trang mà chúng ta tìm thấy chữ cái.)



**- Sometimes you might want to find the difference between the arrays. There are quite a few functions that can help you do that. For example, one of them is array underscore diff. And what it will do is that it will compare the first array against the rest of the given arrays, and will return the values from the first array that are not present in any of the other arrays.**

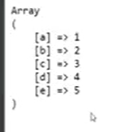
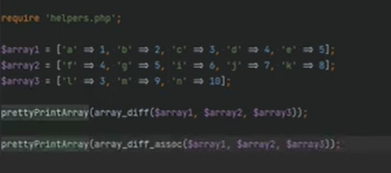
(Đôi khi bạn có thể muốn tìm sự khác biệt giữa các mảng, có khá nhiều hàm có thể giúp bạn làm điều đó ví dụ một trong số đó là mảng gạch dưới khác biệt và điều nó sẽ làm là nó sẽ so sánh mảng đầu tiên với phần còn lại của mảng đã cho và sẽ trả về các giá trị từ mảng đầu tiên không có trong bất kỳ mảng nào khác)

**- For example, if we uncomment the first one here, and we refresh the page, we get 1 and 2. So 4 and 5 from the first array also appear in the second array. Right here. And the number three appears in the third array. So therefore three, four, and five are discarded because they're found in the other arrays and only one and two are kept.**

(Chẳng hạn nếu chúng tôi bỏ ghi chú đầu tiên ở đây và chúng tôi làm mới trang chúng tôi nhận được 1 và hai nên bốn và năm từ mảng đầu tiên cũng xuất hiện trong mảng thứ hai ngay tại đây và số ba xuất hiện trong mảng thứ ba nên do đó ba bốn và năm bị loại bỏ vì chúng được tìm thấy trong các mảng khác và chỉ một và hai được giữ lại)

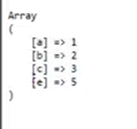
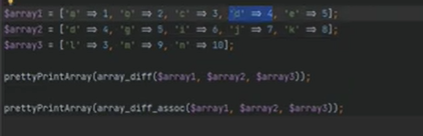
**- And that's what we see on the screen. Note that array diff only checks the values and not the keys. If you want to also check for the keys, you will need to use another function called array diff associated.**

- Và đó là những gì chúng ta thấy trên màn hình, lưu ý rằng mảng khác chỉ kiểm tra các giá trị chứ không phải các khóa nếu bạn cũng muốn kiểm tra các khóa mà bạn sẽ cần sử dụng một khóa khác hàm được gọi là kết hợp khác biệt mảng



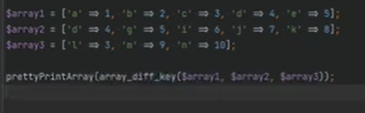
**- And if we uncomment that and refresh the page, now we're getting A, B, C, D, E. So right now we're getting all of the elements from the array because none of these appear in the other arrays. as the key value pair. Meaning we do have the number four in here but the key is f and therefore it doesn't match.**

(Và nếu chúng ta bỏ ghi chú đó và làm mới trang thì bây giờ chúng ta sẽ nhận được a b c d e nên ngay bây giờ chúng ta sẽ nhận được tất cả các phần tử từ mảng vì không có phần tử nào trong số này xuất hiện trong các mảng khác dưới dạng cặp giá trị q nghĩa là chúng ta có số 4 ở đây nhưng khóa là f và do đó nó không khớp )



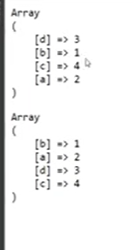
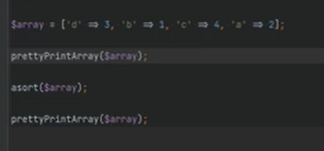
**- If we change this key to d, now this is matching with this and therefore this line should no longer be there and if we refresh it is gone. You could also use array diff key to compute the difference only on keys instead of the values.**

(Nếu chúng ta thay đổi khóa này thành d bây giờ cái này khớp với cái này và do đó dòng này sẽ không còn ở đó nữa và nếu chúng ta làm mới thì nó sẽ biến mất cũng sử dụng khóa khác biệt mảng để chỉ tính toán sự khác biệt trên)



**- So you could do array diff key and we refresh. We get a, b, c, and e because d again appears here.**

(Để bạn có thể thực hiện khóa khác biệt mảng và chúng tôi làm mới, chúng tôi nhận được a b c và e vì d lại xuất hiện ở đây)

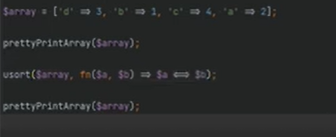


**- There are also plenty of functions available to help you sort arrays. Here are some of them. We have a function called a sort, which will sort array by values and you can pass a different sorting flag if needed. So we can pass their array here and let's keep the sorting flag as default. And let's pre to print array. Let's also print it. before sorting and we refresh. Before it was 3, 1, 4, 2. Now it's 1, 2,3, 4.**

(Ngoài ra còn có rất nhiều hàm có sẵn để giúp bạn sắp xếp mảng ở đây là một số trong số chúng, chúng ta có một hàm gọi là asort sẽ sắp xếp mảng theo giá trị và bạn có thể chuyển một cờ sắp xếp khác nếu cần để chúng ta có thể chuyển mảng vào đây và hãy giữ cờ sắp xếp làm mặc định và hãy in mảng đẹp trước đó. sắp xếp và chúng tôi làm mới trước khi nó là 3,1,4,2 bây giờ là 1,2,3,4)



**- If you wanted to sort by keys, we have a function called ksort and this will now sort by keys instead of the values. And if you refresh, now it's sorted by keys. Note that some of these array functions don't return a new array. Instead, they return boolean. And they take the array argument as reference, so it modifies the original variable. Another useful sorting function is usort, where you can pass custom callback and sort it whichever way you want.**

(Nếu bạn muốn sắp xếp theo khóa, chúng tôi có một hàm gọi là k sắp xếp và bây giờ hàm này sẽ sắp xếp theo khóa thay vì giá trị và nếu bây giờ chúng tôi làm mới thì nó được sắp xếp theo các phím lưu ý rằng một số hàm mảng này không trả về một mảng mới thay vào đó chúng trả về boolean và chúng lấy đối số mảng làm tham chiếu để sửa đổi biến ban đầu, một hàm sắp xếp hữu ích khác là bạn sắp xếp nơi bạn có thể chuyển lệnh gọi lại tùy chỉnh và sắp xếp nó tùy theo điều kiện nào theo cách bạn muốn)

**- So for example, we could do usort here, and as the second argument, it accepts callback, so we can do fn, and it gets two values as the arguments, and we'll call them a and b. We need to return 0 if a and b are equal. We need to return a number less than 0 or minus 1 when a is less than b. And we need to return number greater than 0 or 1 when a is greater than b.**

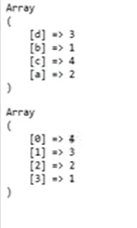
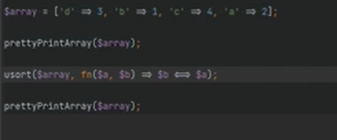
**Now, you can do this manually, of course, but there is a spaceship operator, which we covered before, and it's a perfect use case for this. So we can do a, spaceship, b.**

(Chẳng hạn như chúng ta có thể sắp xếp ở đây và vì đối số thứ hai, nó chấp nhận gọi lại nên chúng ta có thể thực hiện fn và anh ta lấy hai giá trị làm đối số và chúng ta sẽ gọi chúng là a và b, chúng ta cần trả về 0 nếu a và b bằng nhau, chúng ta cần trả về một số nhỏ hơn 0 hoặc trừ một khi a nhỏ hơn b và chúng ta cần trả về số lớn hơn 0 hoặc 1 khi a lớn hơn b. Bây giờ, tất nhiên bạn có thể thực hiện việc này một cách thủ công nhưng có một Toán tử tàu vũ trụ mà chúng ta đã đề cập trước đây và đây là trường hợp sử dụng hoàn hảo cho việc này nên chúng ta có thể tạo tàu vũ trụ b )



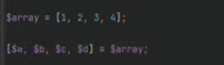
**- And now if we refresh the page, it's going to sort it numerically.**

(Và bây giờ nếu chúng ta làm mới trang, nó sẽ sắp xếp nó theo số)



**- We could also replace this with B spaceship A, and it's going to sort in reverse order, so it should be 4 in the beginning, and 3, 2, 1.Notice that the keys were removed, and it is numeric now. That's because you sort will remove custom keys, and it will use the numeric keys instead**

(Chúng ta cũng có thể thay thế nó bằng b tàu vũ trụ a và nó sẽ sắp xếp ngược lại theo thứ tự nên ban đầu phải là bốn và thông báo từ ba đến một rằng các phím đã bị xóa và bây giờ nó là số vì bạn sắp xếp sẽ loại bỏ các phím tùy chỉnh và thay vào đó nó sẽ sử dụng các phím số )



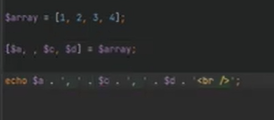
- **And finally, let's move on to array destructuring. You can pull variables out of array or destructure the array into separate variables. For example, let's say that we have this array of numbers 1, 2, 3, 4. We could destructure this and assign them to variables in one line. And we could use a language construct called list or a shorthand version of it, which is just a square brackets like that. and remove the list. Now, even though list is not a built in function and instead it's a language construct, I still wanted to cover this in this video.**

(Và cuối cùng hãy chuyển sang cấu trúc mảng mà bạn có thể kéo các biến ra khỏi mảng hoặc hủy cấu trúc mảng thành các biến riêng biệt, ví dụ: giả sử chúng ta có mảng số 1 2 3 4, chúng ta có thể cấu trúc mảng này và gán cho chúng hai biến trong một dòng và chúng ta có thể sử dụng cấu trúc ngôn ngữ được gọi là danh sách hoặc a phiên bản viết tắt của nó chỉ là dấu ngoặc vuông như thế và xóa danh sách ngay bây giờ mặc dù danh sách không phải là một hàm tích hợp và thay vào đó nó là một cấu trúc ngôn ngữ. Tôi vẫn muốn đề cập đến vấn đề này trong video này)



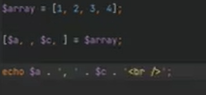
**- So now we can do echo a, b, c, and d. If we refresh the page, we get 1,2,3,4**

(Nên bây giờ chúng ta có thể làm echo a b c và d nếu chúng tôi làm mới trang, chúng tôi nhận được 1,2,3,4)



- **So as you notice, it destructured the array into separate variables. You could also skip elements by simply removing them and keeping the comma. So right now b is skipped, so we can remove the b. And if we refresh, we get 1,3,4**

(Nên bây giờ chúng ta có thể làm echo a b c và d nếu chúng tôi làm mới trang, chúng tôi nhận được 1,2,3,4)



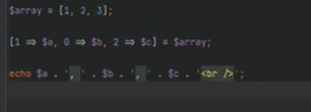
- **You could also skip d. If we refresh, we get 1 and 3.**

(Bạn cũng có thể bỏ qua d chúng tôi làm mới chúng tôi nhận được một và ba)

****

**- You could also destructure nest. arrays so for example if instead three and four were within an array, here we can bring the b and d back and c and d we could put within the bracket. If we refresh we get one two three four.**

(Bạn cũng có thể hủy cấu trúc các mảng lồng nhau, vì vậy, ví dụ: nếu thay vào đó ba và bốn nằm trong một mảng ở đây chúng tôi có thể đưa b và d trở lại và c và d chúng ta có thể đặt trong ngoặc nếu chúng ta làm mới chúng ta nhận được 1,2,3,4)



- **You could also specify keys so for example if I get rid of this and I just put one two and three and we have a, b and c, we can say that a is the value for For the key one, B is the value for the key zero and C is the value for key two. Let's remove this from here and we refresh the page. We get two, one, and three.**

(Bạn cũng có thể chỉ định các khóa vì vậy, ví dụ: nếu tôi loại bỏ cái này và tôi chỉ đặt một hai và ba và chúng ta có a b và c chúng ta có thể nói rằng a là giá trị cho khóa một b là giá trị cho khóa 0 và c là giá trị cho khóa 2. Hãy xóa giá trị này khỏi đây và làm mới trang mà chúng tôi nhận được 2 1 và 3)

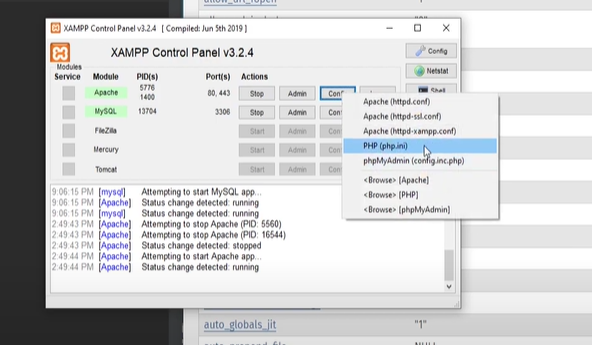
**Video 1.27: How To Work With PHPs Configuration File - PHP**

**2. When working with PHP, you might want to configure and customize certain things within the PHP. That's why PHP comes with the PHP configuration file, which is called PHP.ini. The location of this file will depend on a few factors like server operating system and so on. For example, if you're using xampp, like Im in this case, It makes it simple to check and update the PHP ini file.**

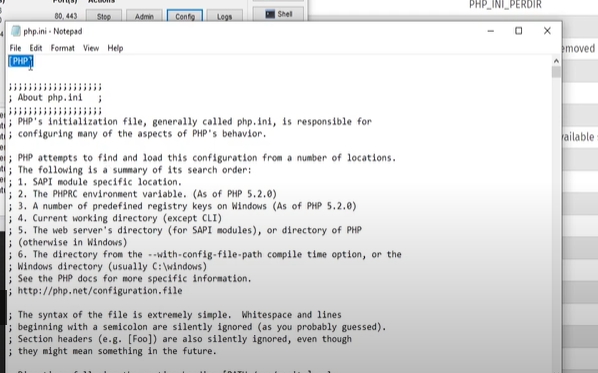
(Khi làm việc với PHP, bạn có thể muốn định cấu hình và tùy chỉnh một số thứ nhất định trong PHP. Đó là lý do tại sao PHP đi kèm với tệp cấu hình PHP, được gọi là PHP.ini. Vị trí của tệp này sẽ phụ thuộc vào một số yếu tố như hệ điều hành máy chủ, v.v. Ví dụ: nếu bạn đang sử dụng xampp, giống như tôi trong trường hợp này, việc kiểm tra và cập nhật tệp PHP ini sẽ trở nên đơn giản)

- **You can just click on config next to Apache and you can find the P H P ini and here. Open it and update whatever you need.**

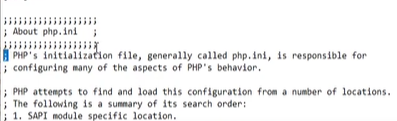
( Bạn chỉ cần nhấp vào cấu hình bên cạnh Apache và bạn có thể tìm thấy P H P ini và tại đây. Mở nó và cập nhật bất cứ điều gì bạn cần.)



**- Just a few things to note here. The text enclosed in the square brackets here is ignored.**

( Văn bản nằm trong dấu ngoặc vuông ở đây sẽ bị bỏ qua.)  


**- Semicolon indicates that it's a comment, so all this will be ignored.** (Dấu chấm phẩy cho biết đó là một nhận xét nên tất cả điều này sẽ bị bỏ qua.)

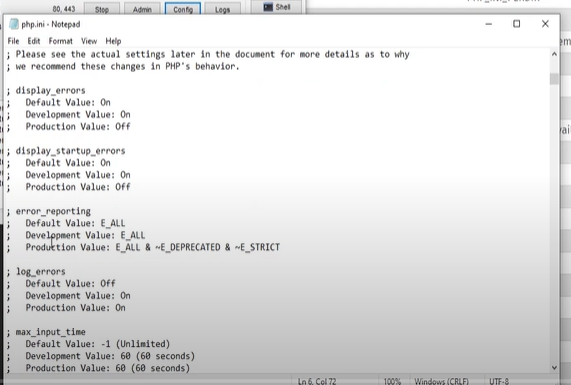


**- It's just comments or documentation**, ( Đó chỉ là nhận xét hoặc tài liệu)

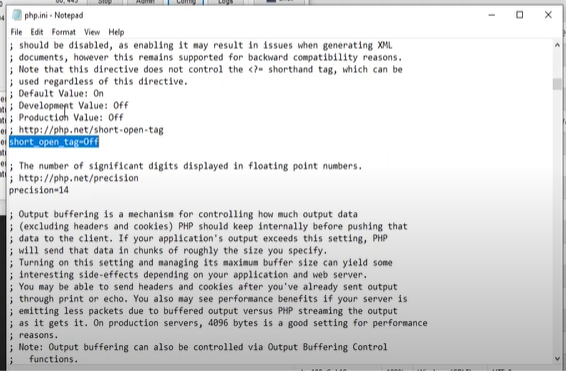


**- And then you can find the individual, settings are also called directives in here**

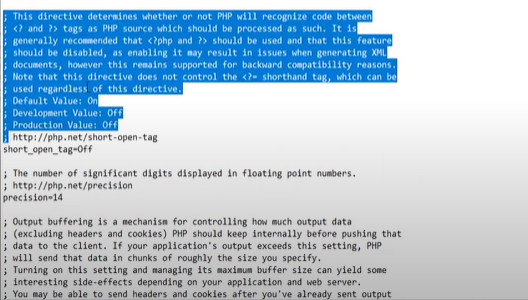
( Và sau đó bạn có thể tìm thấy từng cài đặt riêng lẻ, còn được gọi là chỉ thị ở đây.)



**- So if we scroll down, we can see the short open tag is set to off,**( Vì vậy, nếu cuộn xuống, chúng ta có thể thấy thẻ mở ngắn được đặt thành tắt)



**- And everything above it is just comments explaining what this directive does.** (và mọi thứ phía trên chỉ là những nhận xét giải thích lệnh này làm gì)



**\* There are a lot of options that can be configured and we won't go through all of them because we don't have the time, but we can go through some of the important ones.**

( Có rất nhiều tùy chọn có thể được cấu hình và chúng tôi sẽ không xem qua tất cả chúng vì chúng tôi không có thời gian, nhưng chúng tôi có thể xem qua một số tùy chọn quan trọng.)

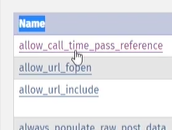
**- I have the list of available, uh, directives open here in the PhP docs, and I will leave the link to this, uh, page in the description.**

(Tôi có danh sách các chỉ thị có sẵn được mở ở đây trong tài liệu P h P và tôi sẽ để lại liên kết tới trang này trong phần mô tả. )

<https://www.php.net/manual/en/ini.list.php>

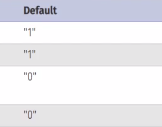
- **The first column here is just the name of the directive.**

(Cột đầu tiên ở đây chỉ là tên của lệnh. )

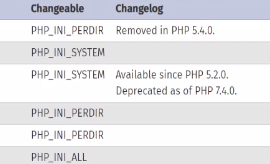


**- The second one is the default value of that directive**

(Giá trị thứ hai là giá trị mặc định của lệnh đó. )

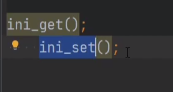


**- The third one is changeable, and then the fourth one is change log.**



**- Now, I would not worry about the changeable column too much. This basically just indicates where this directive can be changed from. You could use functions like InI get and InI set. To get and update the directives**

( Bây giờ, tôi sẽ không lo lắng quá nhiều về cột có thể thay đổi. Về cơ bản, điều này chỉ cho biết lệnh này có thể được thay đổi từ đâu. Bạn có thể sử dụng các hàm như InI get và InI set. Để nhận và cập nhật các lệnh, )



**but some of these directives cannot be updated through P H P during runtime. It can only be updated through the P H P InI file**

(Nhưng một số lệnh này không thể được cập nhật thông qua P H P trong thời gian chạy. Nó chỉ có thể được cập nhật thông qua tệp P H P InI.)

**- If it's says P H P ini\_pedir then you cannot use ini set to update the directive**

( Nếu nó ghi P H P ini\_pedir thì bạn không thể sử dụng ini set để cập nhật lệnh. )

**- Now as mentioned, the InI set updates the value of the directive during the runtime and what that means that it will only be effective during the script execution. I will be using ini set throughout this lesson to set the directive values at runtime for the sake of demonstration**

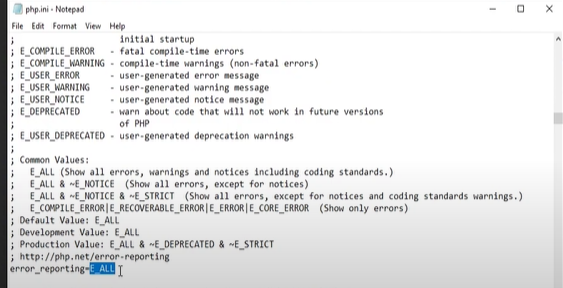
( Như đã đề cập, bộ InI cập nhật giá trị của lệnh trong thời gian chạy và điều đó có nghĩa là nó sẽ chỉ có hiệu lực trong quá trình thực thi tập lệnh. Tôi sẽ sử dụng bộ ini trong suốt bài học này để đặt các giá trị lệnh trong thời gian chạy nhằm mục đích minh họa.)

- **Even though it's fine to use ini\_set for certain use cases in a real application. You would want to set those at. PHP InI configuration file. Note that once you update your P H P InI, you might need to restart your Apache server for those changes to take effect. So let's go over some of the important directives**

( Mặc dù việc sử dụng ini\_set cho một số trường hợp sử dụng nhất định trong ứng dụng thực tế là điều bình thường. Bạn sẽ muốn đặt những cái đó tại. Tệp cấu hình PHP InI. Lưu ý rằng sau khi cập nhật P H P InI, bạn có thể cần phải khởi động lại máy chủ Apache để những thay đổi đó có hiệu lực. Vì vậy, chúng ta hãy điểm qua một số chỉ thị quan trọng.)

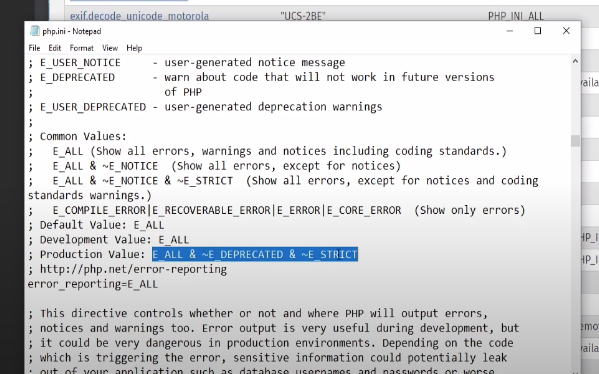
**- The first one we're going to go over is the error reporting, and let me take notes here. Error reporting, error log and display errors.So error reporting allows you to set different levels of uh, P H P error reporting. So for example, if we open the P H P InI and we search for error reporting, we see that it's set to E-ALL**

( Phần đầu tiên chúng ta sẽ đề cập đến là báo cáo lỗi và hãy để tôi ghi chú ở đây. Báo cáo lỗi, nhật ký lỗi và lỗi hiển thị. Vì vậy, việc báo cáo lỗi cho phép bạn đặt các mức báo cáo lỗi P H P khác nhau. Vì vậy, ví dụ: nếu chúng ta mở P H P InI và tìm kiếm báo cáo lỗi, chúng ta sẽ thấy rằng nó được đặt thành E-ALL.)



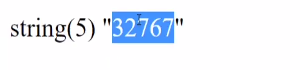
**- That means that all errors, including warnings, notices, and everything will get reported. And as you can see, you could change that and you could use beat wise operator to decide the level of your error reporting. For example, this right here hides the deprecated and strict notices from the page while during the development, you all just reports everything**

( Điều đó có nghĩa là tất cả các lỗi, bao gồm cảnh báo, thông báo và mọi thứ sẽ được báo cáo. Và như bạn có thể thấy, bạn có thể thay đổi điều đó và bạn có thể sử dụng toán tử thông minh để quyết định mức độ báo cáo lỗi của mình. Ví dụ: điều này ở đây ẩn các thông báo nghiêm ngặt và không được dùng nữa khỏi trang trong khi trong quá trình phát triển, tất cả các bạn chỉ báo cáo mọi thứ.)



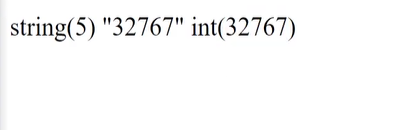
**- As mentioned, you could change this during the runtime, right? So we could do here VAR dump I nine get, and we could do error reporting and let's refresh. And this is just the representation of e All value. This is a constant.**

( Như đã đề cập, bạn có thể thay đổi điều này trong thời gian chạy, phải không? Vì vậy, chúng ta có thể thực hiện ở đây kết xuất VAR mà tôi nhận được và chúng ta có thể báo cáo lỗi và hãy làm mới. Và đây chỉ là sự thể hiện của giá trị e All. Đây là một hằng số)



**- So if we did VAR dump E all, we see that it's the same value.**

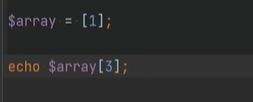
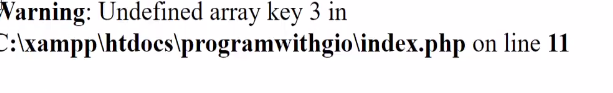
( Vì vậy, nếu chúng tôi thực hiện VAR loại bỏ tất cả E, chúng tôi sẽ thấy rằng nó có cùng giá trị.)



**- For example, if we have an array that has only one value, that is at the Zero Index, if we try to access an array element that doesn't exist, something like this, we'll get a warning**.

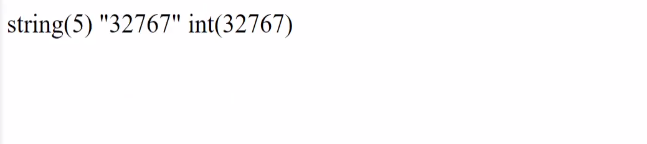
( Ví dụ: nếu chúng ta có một mảng chỉ có một giá trị, đó là tại Chỉ mục 0, nếu chúng ta cố gắng truy cập một phần tử mảng không tồn tại, đại loại như thế này, chúng ta sẽ nhận được cảnh báo.

)

**- We can set it in our error reporting to ignore warnings and not display it on the string. Though this is not recommended at all, you should always use E-ALL for development and even sometimes in production because it allows you to fix your box ahead of time. But for the sake of demonstration, we can do I n I set and simply do not e warning, and we need to set error reporting here and now if we refresh, the warning is gone.**



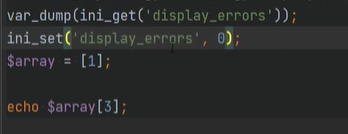
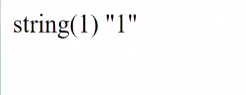


**- Note that it could also set error reporting using the error underscore reporting function. But we'll talk about that in the next lesson. And if you don't understand how this works, it is just the beat twice operators, and you could check my lesson on beat twice operators in this course to review it. So let's remove this.**

( Lưu ý rằng nó cũng có thể đặt báo cáo lỗi bằng chức năng báo cáo gạch dưới lỗi. Nhưng chúng ta sẽ nói về điều đó trong bài học tiếp theo. Và nếu bạn không hiểu cách thức hoạt động của nó thì đó chỉ là toán tử nhịp hai lần và bạn có thể xem bài học của tôi về toán tử nhịp hai lần trong khóa học này để ôn lại. Vì vậy, hãy loại bỏ điều này.)

- **We get the warning displayed again. And the next item is display errors. And display errors simply just indicates whether you want to display errors on the screen or not. So we can say display errors here and we get the value one and we can set to zero. And if we're refresh now, we no longer see the errors.**

(Chúng tôi nhận được cảnh báo hiển thị một lần nữa. Và mục tiếp theo là lỗi hiển thị. Và lỗi hiển thị đơn giản chỉ là cho biết bạn có muốn hiển thị lỗi trên màn hình hay không. Vì vậy, chúng ta có thể nói lỗi hiển thị ở đây và chúng ta nhận được giá trị 1 và chúng ta có thể đặt thành 0. Và nếu chúng tôi làm mới bây giờ, chúng tôi sẽ không còn thấy lỗi nữa.)

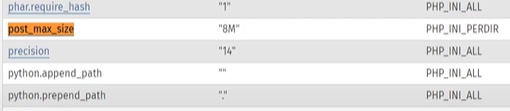
 

**- Now in production, it would probably set display errors to zero, so you wouldn't show any of the sensitive information within your errors to the user, but instead it would log your errors. And that's where the error log comes in. So error log basically indicates where the error should be logged. We'll talk about the error handling in the next lesson more.**

( Hiện đang trong quá trình sản xuất, nó có thể sẽ đặt lỗi hiển thị thành 0, do đó, bạn sẽ không hiển thị bất kỳ thông tin nhạy cảm nào trong các lỗi của mình cho người dùng mà thay vào đó, nó sẽ ghi lại lỗi của bạn. Và đó là nơi xuất hiện nhật ký lỗi. Vì vậy, nhật ký lỗi về cơ bản chỉ ra lỗi sẽ được ghi vào đâu. Chúng ta sẽ nói nhiều hơn về cách xử lý lỗi trong bài học tiếp theo.)

**- Next, we have post max size, and we haven't covered forms or posting data, but this setting basically determines how much data can be posted in a request.**

(Tiếp theo, chúng tôi có kích thước tối đa của bài đăng và chúng tôi chưa đề cập đến biểu mẫu hoặc dữ liệu đăng, nhưng về cơ bản, cài đặt này xác định lượng dữ liệu có thể được đăng trong một yêu cầu.)

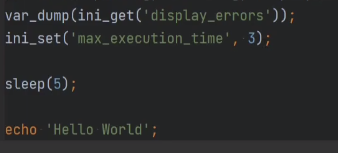


**- Next we have max execution time, and this just says the maximum number of seconds the script can run before it is timed out and terminated. By default, it's set to 30 seconds**

.( Tiếp theo, chúng ta có thời gian thực thi tối đa và điều này chỉ cho biết số giây tối đa mà tập lệnh có thể chạy trước khi hết thời gian chờ và chấm dứt. Theo mặc định, nó được đặt thành 30 giây.)



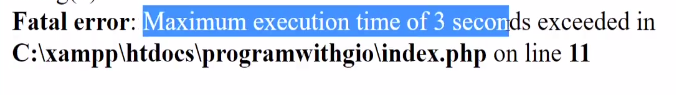
**- And for example, if we set max execution time to say three seconds, and here we sleep for five seconds and then we echo out. Hello world. Let's see what happens.**



**So it's going to sleep for five seconds, but our max execution time is set to three. We're going to get the fair alert that the maximum execution time has exceeded**

( Và ví dụ: nếu chúng ta đặt thời gian thực hiện tối đa là ba giây, và ở đây chúng ta ngủ trong năm giây rồi lặp lại. Chào thế giới. Hãy xem điều gì sẽ xảy ra. Vì vậy, nó sẽ ngủ trong năm giây nhưng thời gian thực hiện tối đa của chúng tôi được đặt thành ba.

Chúng tôi sẽ nhận được cảnh báo công bằng rằng thời gian thực hiện tối đa đã vượt quá.)



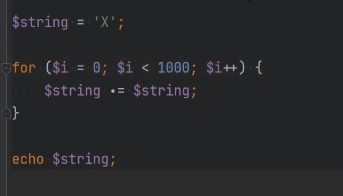
**- Next we have memory limit, and this just says the maximum amount of memory a script can consume. So we can check what is currently set in our case and we're going to refresh the page and we get 512 megabytes.**

(Tiếp theo, chúng ta có giới hạn bộ nhớ và điều này chỉ cho biết dung lượng bộ nhớ tối đa mà tập lệnh có thể tiêu thụ. Vì vậy, chúng tôi có thể kiểm tra những gì hiện được đặt trong trường hợp của chúng tôi và chúng tôi sẽ làm mới trang và chúng tôi nhận được 512 megabyte. )

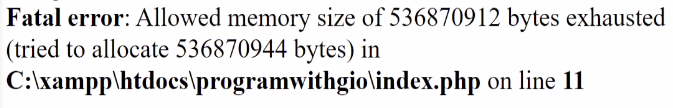
**- Now to trigger the out of memory error, we can set some kind of string here equals to x, and then we could do some kind of loop that runs about a thousand times and just pen string to itself**

( Bây giờ, để kích hoạt lỗi hết bộ nhớ, chúng ta có thể đặt một số loại chuỗi ở đây bằng x, sau đó chúng ta có thể thực hiện một loại vòng lặp nào đó chạy khoảng một nghìn lần và chỉ ghi chuỗi đó vào chính nó.)



**- And then we could echo out string. And if for refresh, we're going to get the fatal error**

( Và sau đó chúng ta có thể lặp lại chuỗi. Và nếu để làm mới, chúng ta sẽ gặp phải lỗi nghiêm trọng.)



**- Now, we could also set the memory limit to minus one, which removes the memory limit entirely, but I would advise against it because it's better to always find a way to optimize your code to see where the memory is being consumed. That way you're not running into memory issues.**

( Bây giờ, chúng ta cũng có thể đặt giới hạn bộ nhớ thành trừ một, điều này loại bỏ hoàn toàn giới hạn bộ nhớ, nhưng tôi khuyên bạn không nên làm như vậy vì tốt hơn hết là bạn nên luôn tìm cách tối ưu hóa mã của mình để xem bộ nhớ đang được sử dụng ở đâu. Bằng cách đó bạn sẽ không gặp phải vấn đề về bộ nhớ.)

**- Let's move on to the next directive, and that is file uploads. This just enables or disables the file uploads on P H P. We haven't talked about the file uploads yet, and we'll talk about that later, so don't worry about it right now**

( Hãy chuyển sang chỉ thị tiếp theo và đó là tải tệp lên. Điều này chỉ kích hoạt hoặc vô hiệu hóa việc tải tệp lên trên P H P. Chúng ta chưa nói về việc tải tệp lên và chúng ta sẽ nói về vấn đề đó sau, vì vậy đừng lo lắng về điều đó ngay bây giờ. )



- **Related to file uploads, we also have upload TMP directory, and this just indicates the temp folder where the temp files will be stored when doing file uploads.**

(Liên quan đến việc tải tệp lên, chúng tôi cũng có thư mục TMP tải lên và thư mục này chỉ cho biết thư mục tạm thời nơi tệp tạm thời sẽ được lưu trữ khi thực hiện tải tệp lên.)



**We also have the upload max file size here.**

(Chúng tôi cũng có kích thước tệp tải lên tối đa ở đây.)



- **Another one we have is the date, time zone, that is the default time zone that you set.**

**(** Một cái khác mà chúng tôi có là ngày, múi giờ và đó là múi giờ mặc định mà bạn đặt.)



**- Next we have include path, and we covered this briefly when we talked about including P h B files, and this just specifies the list of directors where the require include, require ones, and include ones and other functions that deal with opening files would look for files by default**

( Tiếp theo, chúng tôi có đường dẫn bao gồm và chúng tôi đã đề cập ngắn gọn về vấn đề này khi nói về việc bao gồm các tệp P h B và điều này chỉ xác định danh sách các giám đốc trong đó các yêu cầu bao gồm, yêu cầu và bao gồm các tệp cũng như các chức năng khác liên quan đến việc mở tệp sẽ xem xét cho các tập tin theo mặc định.)



**- We'll talk more about the P H P InI and more advanced configuration directives once we get to the security part of this course, because there are some directives that you should keep disabled and some directives like session directives that you should know and be aware of how they work when it comes to security**

( Chúng ta sẽ nói nhiều hơn về P H P InI và các chỉ thị cấu hình nâng cao hơn khi chúng ta đi đến phần bảo mật của khóa học này, vì có một số lệnh mà bạn nên tắt và một số lệnh như chỉ thị phiên mà bạn nên biết và biết cách thực hiện. họ làm việc khi nói đến an ninh.)