

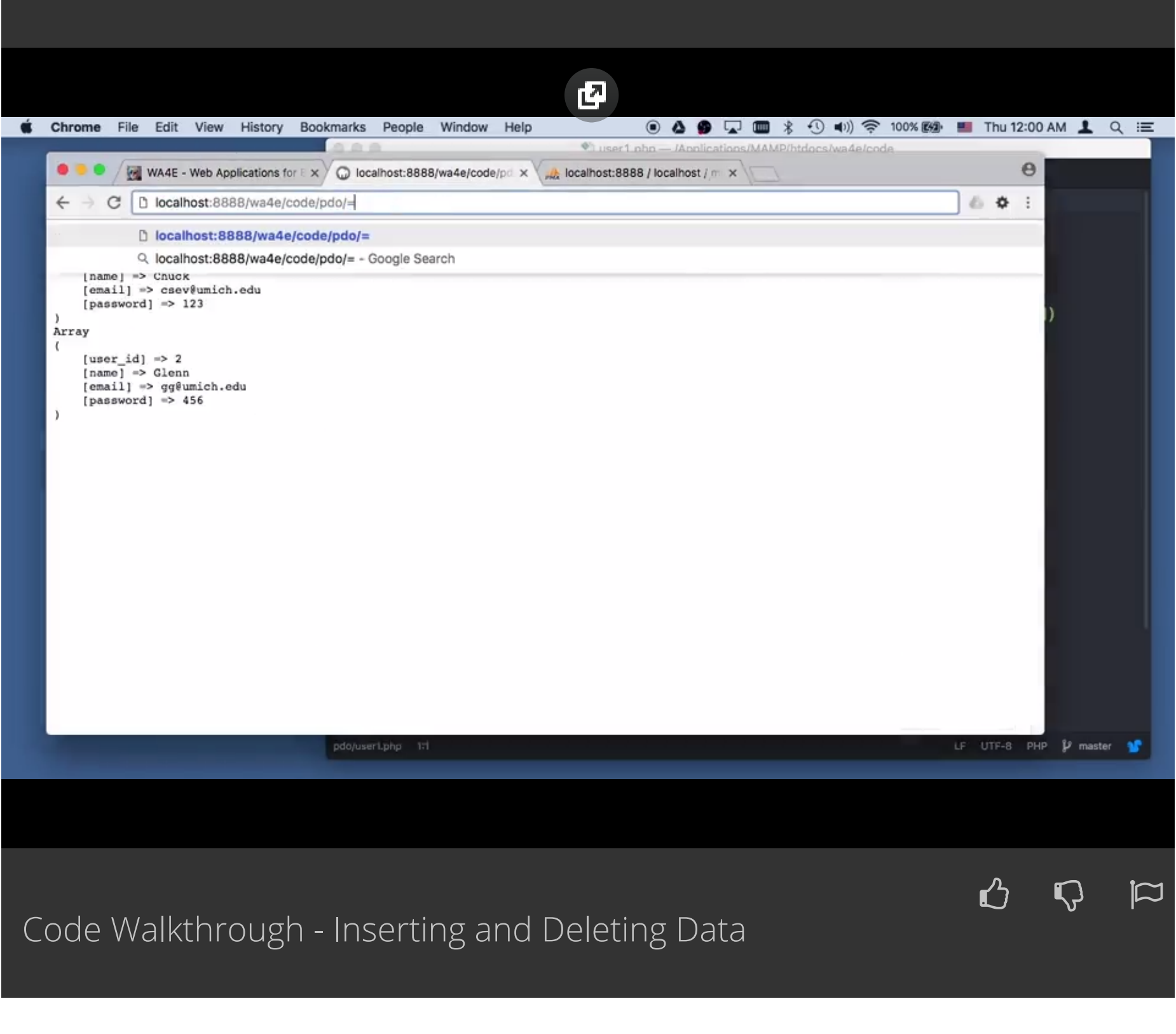
Lecture Content

PHP Database Libraries	7 min
Running SQL Queries in PHP	11 min
Accessing MySQL Using PDO: Inserting Data	10 min
Security Issue: Avoiding SQL Injection	9 min
Error Handling with PDO	8 min
Code Walkthrough - PHP, MySQL, and PDO	8 min
Code Walkthrough - Inserting and Deleting Data	9 min
Code Walkthrough - Security and SQL Injection	8 min

Practice Quiz: PDO 5 questions

Assignments

Bonus Materials



Code Walkthrough - Inserting and Deleting Data

Have a question? Discuss this lecture in the week forums.

Interactive Transcript

English

0:00

Hello and welcome to Web Applications for Everybody. Right now, we're working through some of the code for [the PDO, portable data objects](#). Now you have to run this on your own localhost, and you have to follow the notes to get things setup, we did that in a previous video. So what I'm going to play with now is a little more useful code, user1.php, okay? And so, if I take a look at user1.php, [COUGH] It's followed the pattern of requiring pdo.php and of course, pdo.php has my database connection information with ID and password. And I'm going to use sort of my model-view-controller pattern, where I'm processing the data at the top, and then I have my template at the bottom. So this is all the silent processing code, and then here is the logic. And so, on a get request, it's just going to fall straight through here because there's no post data, and it paints this post form which is exactly this. And it gives me a name, a password, and email. So I can say, Sally, sally@uiuc.edu, and Sally's password is 999. And then, I hit post, and then that's going to come back in the top of this script again. And this time, these three variables are going to be set. Okay, so when I hit Add New here, it's going to add another value. And so it runs through here, and here's the SQL. And this is PDO. PDO basically has place holders for :name, :name, and :password, and we'll talk about why this is later. I just print out the SQL, and that shows this insert. And then it runs that statement, taking the name, email and password from the post data. Now, I didn't have to name name, email, and password the same. But sometimes you do that just to keep yourself sane. And so that did the insert, I did an insert as the result of a post. So if I go take a look at my new rows, you see that now I have a Sally row, pretty cool? Okay, so that's user one.

2:10

Let's take a look at user two. So user two basically is going to in addition, this code here is the same, it's the code to do the insert, and this html is the same. But the only thing that I've done different is I'm printing out a table and then printing out all the data. So if I go to user two instead of user one it looks exactly the same. This form is going to be at the bottom.

2:34

And so, but we see the old ones and so now I can say something like, Fred, fred@uumish.edu

2:50

And a password of some random set of numbers, and this form here is the same. The difference is that now when I do the post, it's going to come in here, it's going to do another insert, and it's going to fall through. And then it's going to show me all the new ones. And so now we have this thing where there you go, and Caitlyn.

3:16

So we'll put Caitlyn in. So we now have this thing that's both showing us, each time. I kind of messed that up, I'll have to fix that later. Her email's wrong, I didn't do any data validation. But you can still see this one here where we're sort of also we're querying. We're reading and constructing the tables. So we kind of communicated a few of these things. And so let's take a look at our database, browse and there's Kate, okay. So now what I'm going to do is I'm going to show you how we might delete.

3:47

Now, the key thing to deleting is you're not supposed to delete on a get request. And so I'm going to write this thing called userdel and pass in the user_id that I want to delete as a post parameter. So let's go into user2del.

4:09

And so now I just painted a form right here. User ID is a field, and I'm going to delete the Kate one that I just made a mistake typing in. So I'll put 5 in and send a post in. So now I'm going to hit this Delete button. It's going to run here again. And this time post will be set, because it'll be set from that 5. And I'll delete from users where user_id = :zip. That is a placeholder, I'll print out the SQL in a pre, I'll prepare it, and then I'll run it mapping zip to the data that came in from this field over here. So this code's going to run when I click this button. All right, so we'll go hide that, so it ran and the user's gone. That didn't look very green. So when I go over here and I browse it, Kate's gone. And I can get rid of 4. I can say 4, Delete. And I take a look. And it's gone. Now here's the thing, just to remind you about SQL, if I delete 42,

5:13

There's no 42 here, that's not an error. Because I said, DELETE FROM users WHERE user_id = 42. Well there was no user ID 42, so it did exactly what I asked, that's not a syntax error. It's a syntax error if don't spell the word delete right or the word from right. But having no records that get deleted is not itself an error. So that's how we do a delete.

5:39

So the next thing I want to talk about is user3.php. And that is really just combining all these things together. So I have the add code here, right? And then I have the delete handling.

5:57

And then I have the HTML, so let's just bring user3 up.

6:04

It's like doing all of it together.

6:07

So it shows everything but the interesting thing is this little, tiny Delete button here. So most of it we've seen before, we read and we put all this stuff out, and then in this last field, right here in this last field right there, probably easier if I just view source and show you what it looks like.

6:30

In this last field, I put a form.

6:37

Let's make this a little bigger, make it bigger. The last, so td is the last column and then /td ends that and in that table is a little miniature form. It's a method = post, input type = hidden, hidden input fields don't show, user_id= 3 and then I have a submit button that has a string del with a name = "delete". [COUGH] So this form has actually four forms on it. I mean this page has four forms on it. There's a form, there's a form, there's a form and this is a form as well. And I can tell which of these buttons was pushed. And that's why I named, This one delete so I know that's a delete. And then I have add new. [COUGH] And I can also tell up in this code, whether or not it was a delete that I post or it was an add post. so that ad post code is right here and if it is a delete none of these things will be set. Email, password won't be set. If it's a delete then innate password, email won't be set, but user_id will be set, and remember, I'm putting user_id in here as a hidden value which is the primary key of the user_id. So how do I construct that text? Well, it's pretty simple. I have the table, td and then I print out the form this is just text the hidden user_id, name = "user_id" value =, and then I concatenate the actual row to user_id. I don't need to use HTML entities here because I know it's a number because it's an internal thing that I invented. And then input type and new line and form. I'm just kind of constructing a form from echo statements, that's exactly what I do. Okay, so if I click on Del here, Glenn is gone and it came back up to the top. So it ran here, it came down, it ran this bit to delete because that's the button I hit. And then it fell through and produced the table, and so for us it looked like this just vanished. So we'll make a new one Roheta,

8:56

UMISS.edu Password is some stuff, and we'll Add, and poof, it shows up. So that time I pushed this and this post code triggered instead. And so I can have different forms doing every different thing here and it all just kind of works. And so that is user one, user two, and user three walking through. Hope this helps. Cheers. [MUSIC]

Downloads

Lecture Video	mp4
Subtitles (English)	WebVTT
Transcript (English)	txt

Would you like to [help us translate](#) the transcript and subtitles into additional languages?