

Lecture Content

- Redirect, Routing, and Authentication

9 min
- Code Walkthrough - Routing and Redirect

5 min
- POST / Refresh / Redirect

10 min
- Code Walkthrough - POST / Redirect

6 min
- Implementing Login and Logout

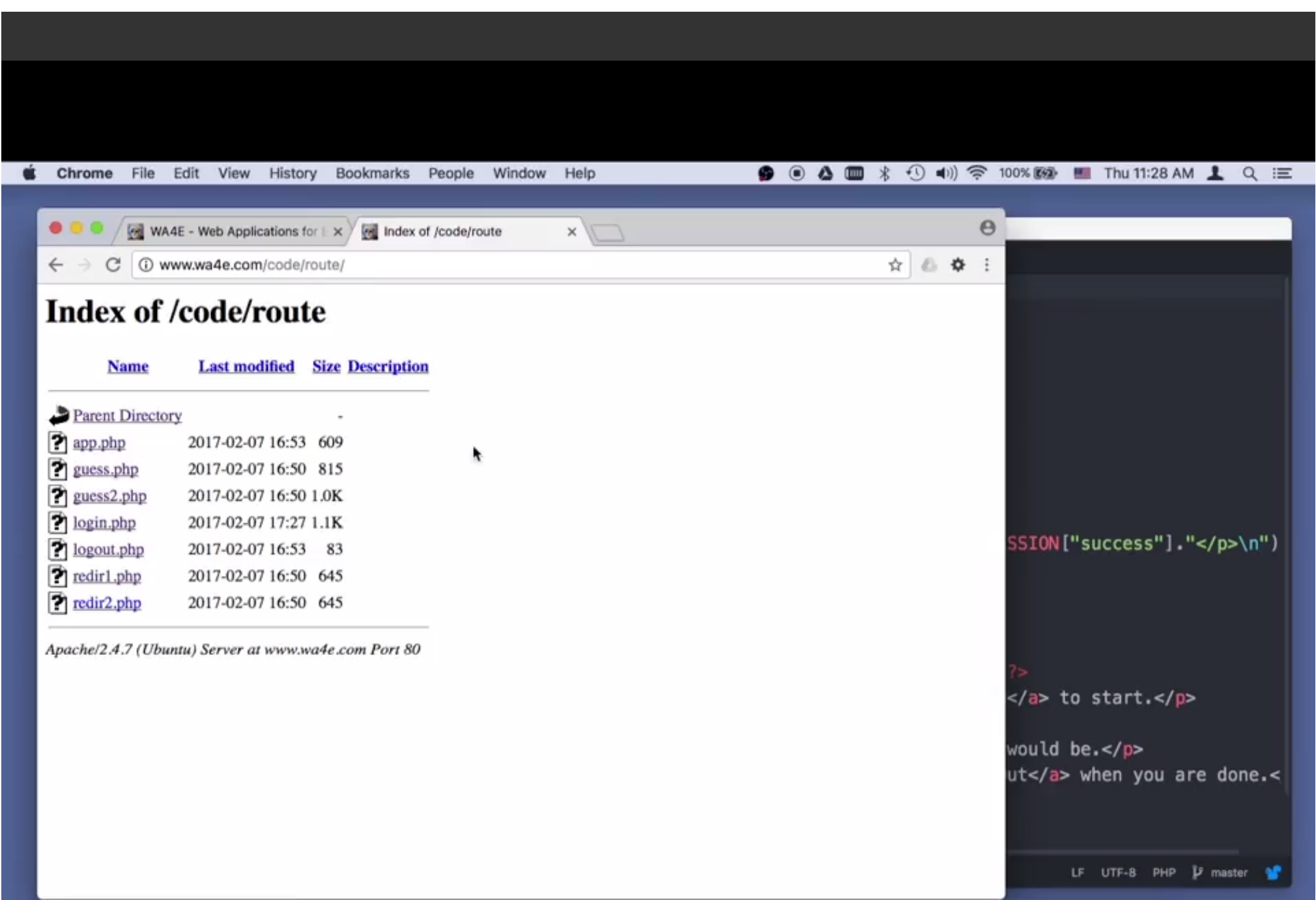
10 min
- Code Walkthrough - Login and Logout (3)

6 min
- Practice Quiz: Routing

11 questions

Assignments

Bonus Materials



talk about the login function where we're talking about all these things together.

Code Walkthrough - Login and Logout (3)

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

Interactive Transcript

English

0:00

Hello and welcome to Web Applications for Everybody. In this little bit we're talking about some of the routing, the redirecting, the POST redirecting, the no refresh on the posting, we did all that before. Now, we're going to actually, sort of, [talk about the login function where we're talking about all these things together](#). And so here we have this application and this application is pretty smart. It detects that you're not logged in and then it sends you to a login screen and so you can login on any account as long as the password is umsi. And, now we're logged in and there's a little success message here and then you can log out and now you're logged out. Okay, so there's several files. There's app.php, this is made up of app.php, login.php, and logout.php. So login, at least, so far this application doesn't do much. It actually doesn't have any POST handling so the model part is just get the session started and then we drop into HTML. Blah, blah, blah, blah, blah. We'll come back to this success bit in a second. This is what's called a flash message. But what we're doing is we're basically using as the indicator, whether you're logged in or not, the keys account being present in the session. So if the account key is not in the session, which it is not there when you first start out, it says "Please Log In". But if the account, if you have logged in, then the account is in the session and that's where we say, hey, here's "a cool application" and go ahead and Log Out. Okay? So it comes in, starts a session. There is no account key in it so it says please log in and that's just a standard href so we go over there. While we're doing that let's look at developer console, get that started up, get that ready for us, because that will be so much fun. Now login is where the fun begins. So here we have login, we have model code up here at the top, session start, of course and here is our POST code, and let th- we'll come back to that really quick. And so now here is our body. And the first thing we're going to do is these things we call flash messages. And so I'm going to put errors in the session under the key error and I'm going to put good things that happen to us under the key success. Okay? And so at the beginning of this "Please Log In", right here between those two things, I'm going to check to see if there is an error in the session and if it is I'm going to print that out, red, and I'm going to unset it and that's what we call a flash message. And so in this case, and the same would be true if there's a success except I print it out green and then I unset it. So right here between "Log In" and the start of the form there might be a message or may not. And so if in the case now where I have a bad login, right? Right, and I press "Log In" it's going to come up here. It's going to see the POST and it's going to unset the session["account"]. That logs out the current user. Remember I said that when the account key is in the session array then we're logged in so that's kind of a logout. If the password is equal to umsi then you're good. And we're going to log in. And if it's not we're going to not log in so we're going into this code first. We're going to set the error to "Incorrect password" and then we're going to redirect back to ourself. We're in login.php and we're going to redirect back to ourself. So we did a POST, that's a POST and then a redirect back to ourselves but we also sneaked into session this little thing. So after the redirect the GET comes back in and this code triggers and runs and out comes "Incorrect password". Now the interesting thing is because we have removed it if I refreshed this page and it is legit to refresh it because I'm sitting on a GET request, if I reset it that password, that error is not there. It only showed up once. It's a flash message. That's why we call it flash. It flashes on this but then if I do another request response cycle it's not there. And we achieved the flash by getting it out of the session once we've displayed it. Okay? So now let's talk about what happens when you have a successful password. So we'll go you know Sarah and then umsi as the password it's going to come up here and this is now really starting to get very controller-y feeling because the controller routes. In this case it decided to route back to the existing script that we were coming from. But now once the password matches we're going to put up a little cute little success message in the session and we're going to set the account. And this account signals to app.php that log in worked. So we're sending a message in the session from one script to the other script. So this redirect is going to happen. So I'm going to "Log In" correctly. So "Log In" is a POST and then it redirects but in this case you see that redirects to app.php and then it does a GET and so this is the response that comes from that GET. But then if we look at app.php we see it checks to see if there's a success message in the session and it prints it out in green and then it unsets it. So it's a flash, so if I refresh it, that's no longer in the session. Okay? And this is where we would have some cool application stuff, playing a game or something and then we're going to go to logout.php. So logout.php is always my favorite script to write because it all pretty much has three lines. We start the session, we wipe out the session removing the account, and then we redirect back to app.php. So let's clear this bit out and I'll press "Log Out" and we're going to do a GET request to log out. There's not a POST request to log out and it clears the session. So it was a GET request and we sent a 302 as a result of a GET request after having wiped out the session and then said go back to app.php so then it goes back to app.php and it runs through. There's no success message and there is no key and their account key has been gone. It's been taken out and so we just logged out. Okay? So I hope that's useful to you. It gets, kind of brings all this stuff a model view controller and POST redirect and all that stuff is now working quite a bit in this particular small application that just has three files.

Downloads

Lecture Video mp4

Subtitles (English) WebVTT

Transcript (English) txt

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