Getting Started:

Hey there, how's it going everybody? In this series of videos we're going to be learning how to build a full-featured web application using the Django framework in Python. So Django is a very popular framework that gives us a lot of functionality right out of the box and makes it really enjoyable to work with these web applications. So first let me show you what we'll be building in this series of videos and then we'll get started learning how to actually put all of this together. So this is the application that we'll be building here and you can see that it's a blog style of an application where different users can write different posts. Now, this can be like blog posts or they can be Twitter updates or whatever it is that you want to do with this.

Now, we have an authentication system, so I'm logged in right now. So we can log out. And if we go to register, then new users can create a new account. If you already have an account, then you can log in. And we can see at the login page, we have this forgot password link. Now, that will allow users to reset their password by getting an email. Now, if we have an account, then we can log in. the username I created and I'll log in with my password. And now that we're logged in, we have a few different options up here at the top. So we can view our profile and update our profile information. Now we can also update the profile picture that we have here.

So if I go to choose file and choose a different file and update that, then we can see that profile picture was updated. Now that's also resized in the background to save room on our web server if that picture is too large. So if we go back to our home page then we can view other people's posts. So we can view an individual post here by someone else. Now if we view an individual post that is something that we've written, then we also have the ability to update or delete that post. So if I go to update and say my latest updated post and post that, then we can see that now that post was updated with that new content. So we also have the ability to delete posts.

So if I click on delete, then we'll ask if we want to confirm that we want to delete the post. So I'll say yes. And then back on the homepage, we can see that that post was deleted. So that's a quick tour of the application that we're going to be building. And building something like this is a great way to learn the ins and outs of a framework because you're going to be exposed to so many different things. So for example, we'll learn how to work with databases and also how to create an authentication system and accept user input from forms and send emails to reset passwords and all kinds of different things like that. Now, since this is a Django application, We also have the ability to access an admin page if we have the correct permissions.

And within here, you get a nice GUI to be able to view all of this backend information and update it on the fly if you'd like. So let me go back to the main site here. So we're going to be learning how to add a lot of functionality to an application within this series. Now, I'm going to mention this several times throughout the series, but if you're following along and would like to download the source code of each step in the process, then I will have links to the source code of each video in the description section below so that you can download those if you'd like. And also, if you'd like to know how to build the same application in another framework, then I also have a Flask series where I've created the same application using the Flask framework.

So if you're interested in that, then I'll put a link to that in the description section below as well. Also, I'll be doing a video in the near future where I compare the two frameworks and when it might be best to use one over the other. Okay, so let's get started with learning how to build this application using Django. So I'm going to close down my browser here and pull up my terminal. And if you're on Windows, you can pull up your command line. So first off, let's start off by installing the packages that we need to get started out. So you can do this in a virtual environment or in your default Python environment. But it's always a good idea to separate different projects into their own virtual environments.

Now, I'm mainly going to focus on Django in this series. So if you need to install Python or want to learn how to work with virtual environments or are wondering how I set up my text editor or anything like that, then I'm going to put links to those videos in the description section below. But I'm not going to go into those in detail in this series. I'm just going to assume that you're specifically ready to start learning Django.









