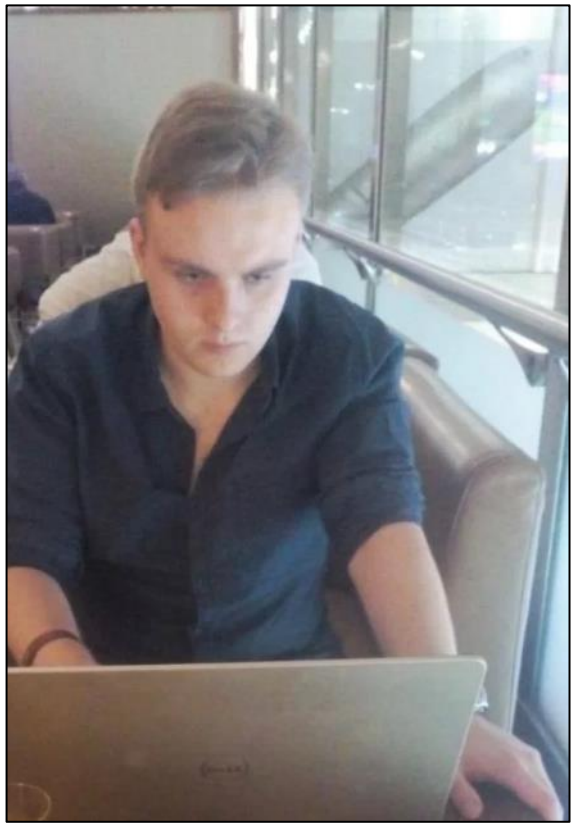
SECTION:

COURSE INTRODUCTION



Arno Pretorius

My name is Arno!

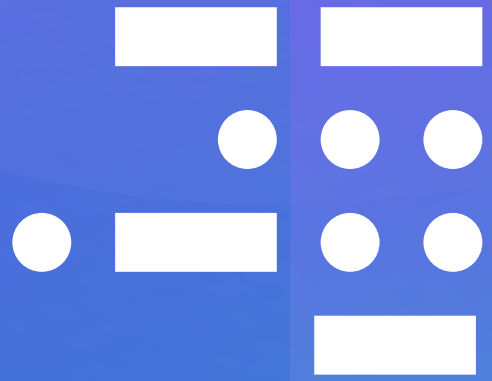
I’m a qualified IT teacher with teaching experience both in-person and online.

Over the years, I have created and deployed many real-world Django applications to the AWS cloud.

AWS Certified

PYTHON DJANGO: ULTIMATE SECURITY CHECKLIST - 2022

By Arno Pretorius



COURSE OVERVIEW

Are there any pre-requisites?

- A basic knowledge of HTML, CSS and JavaScript is required
- You also need to have a good knowledge of Django

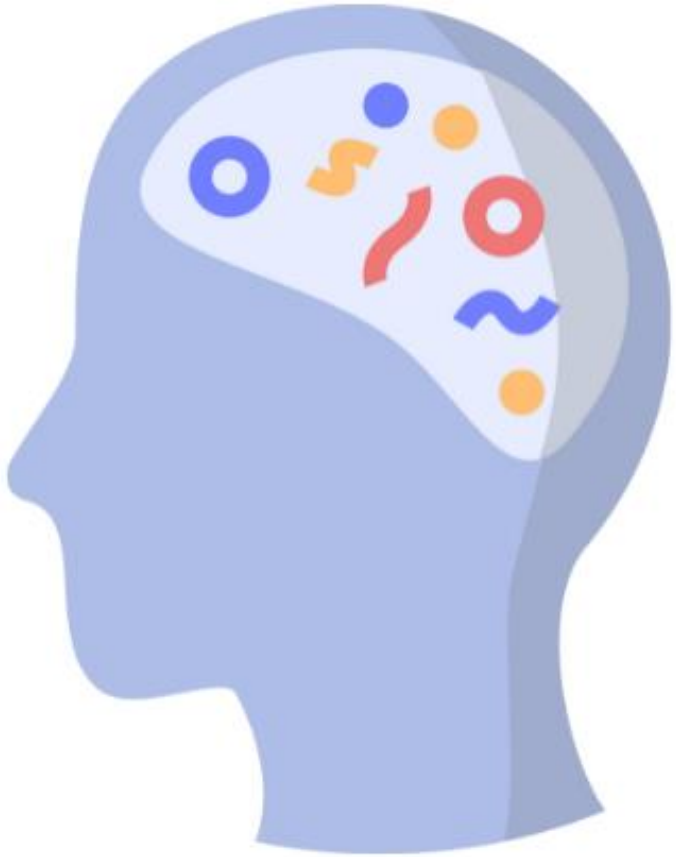
What will we learn?

- We will learn how to secure our Django web application in the following ways:

- Adding a reCAPTCHA
- Implementing Two-Factor Authentication (2FA)
- Adding a session timeout
- Protection against brute force attacks
- Creating environment variables
- Configure and set various built-in security settings
- Extra tips and tricks to know…

THOUGHT PROCESS

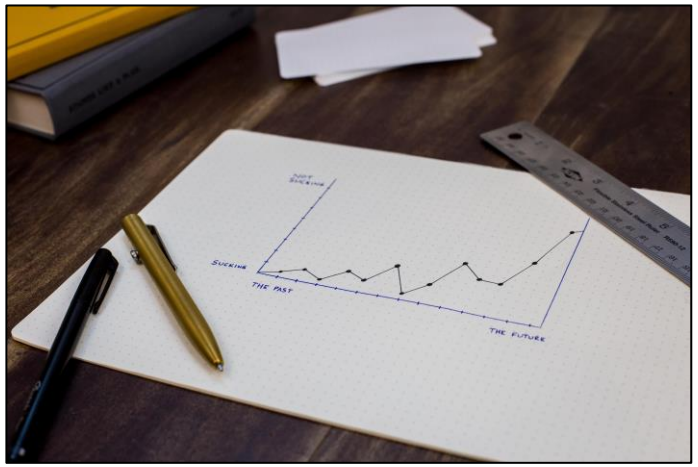
Before we begin...



Structure our thought process

One step at a time...

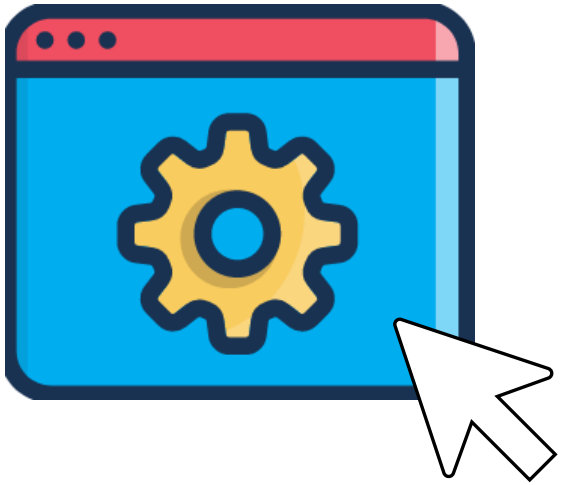
Focus on progress and not on time



JUST A QUICK NOTE...

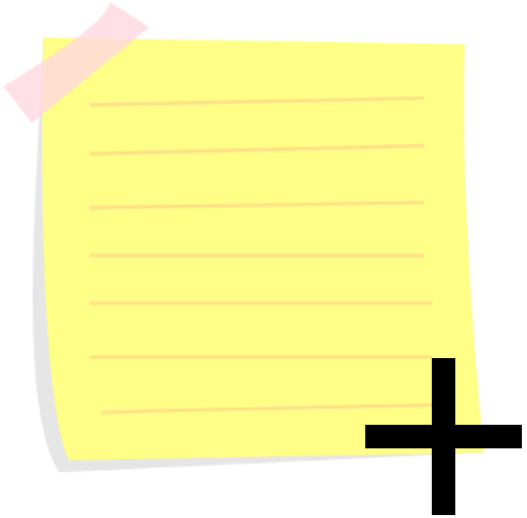
A QUICK NOTE.

Practical / hands-on activity



ANOTHER QUICK NOTE.

Includes a downloadable PDF guide.



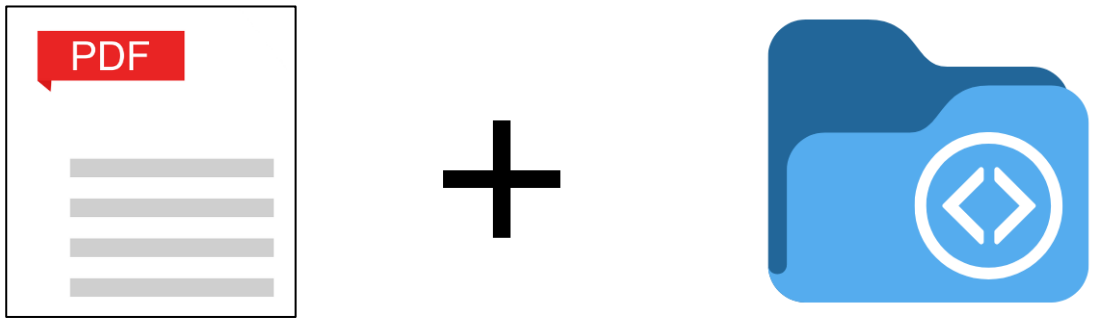
SECTION:

RESOURCES

Resources

Resources - download

- The project code will be attached as a zip file
- The lecture slides will be attached as a downloadable PDF



SECTION:

BASELINE INTRODUCTION AND SETUP

Python:

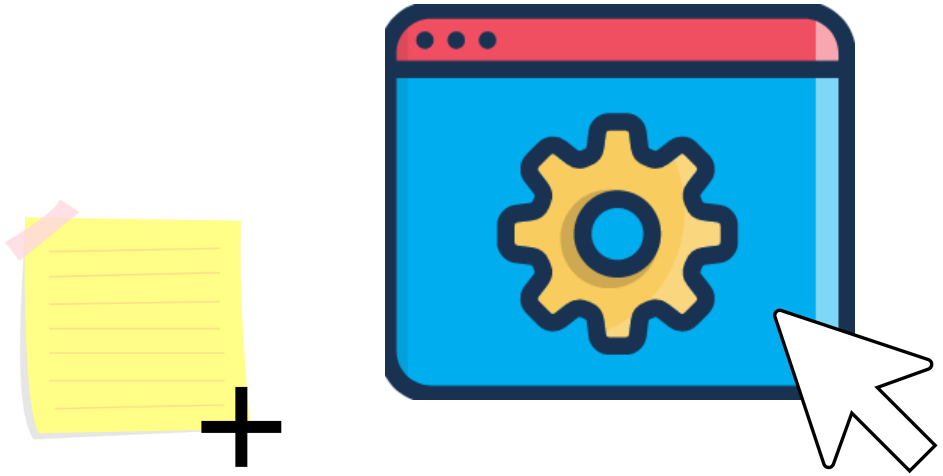
Installation and setup

Python installation and setup

- Python needs to be installed on our computer before we can use Django
- We will install and setup Python version 3.9.13
- It is **highly recommended** that you use the exact same version for the purpose of this course
- During the process, be sure to add Python to your path

Practical

- Install and setup Python



Visual Studio Code:

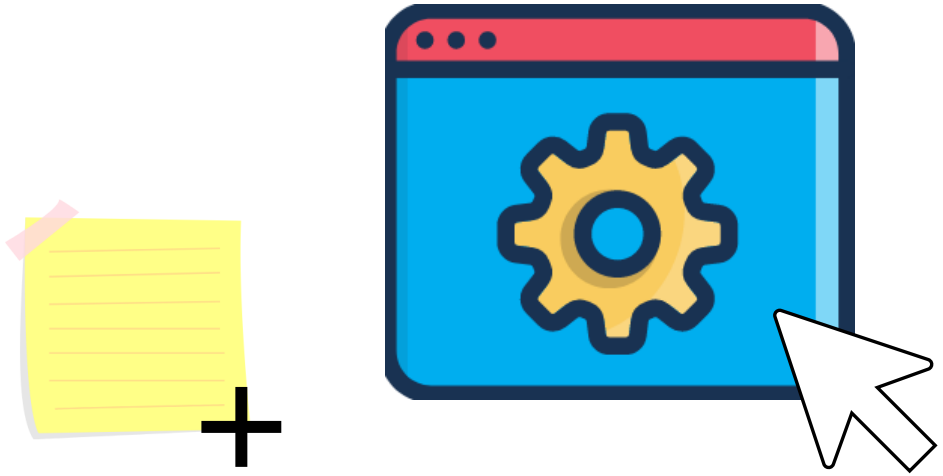
Installation and setup

VS code installation and setup

- We will need to use a **source-code editor** to write our code
- There are many source-code editors, such as: VS Code, Sublime Text and Atom
- It is highly recommended that you use VS Code for the purpose of this course

Practical

- Install and setup VS code



SECTION:

CREATE A BASIC DJANGO WEB APP

Django project setup

Practical

- Setup a virtual environment for our project
- Install and setup our Django project
- Test our server

Django app setup

Practical

- Create a Django app
- Configure our Django app

Templates, URL's and Views

Practical

- Configure Django to render templates
- Setup our URLs and views

Configure static files

Practical

- Configure static files in Django
- Configure and connect - CSS and JavaScript files
- Configure and connect images

Styling our web app

Practical

- Add a free bootstrap theme **from bootswatch** to style our web application

User registration

Practical

- Create a model form to register users
- Register a user

SECTION:

RECAPTCHA

reCAPTCHA

reCAPTCHA

- A **reCAPTCHA** is a free service provided by Google that is used to protect websites against spam and abuse
- This is typically used on registration / sign up forms
- A ‘CAPTCHA’ is effectively a ‘**Turing test**’ that is used to tell the difference between a robot/bot against a human
- There are multiple versions of reCAPTCHA available

Practical

- Setup a free reCAPTCHA in Google Cloud

Practical

- Add a reCAPTCHA to our Django web app

SECTION:

TWO-FACTOR AUTHENTICATION

Two-Factor Authentication

Two-Factor Authentication (2FA)

- Two-Factor Authentication or ‘2FA’ is an extra layer of security that is added in conjunction with a username and password
- Usually, it is in the form of a physical or virtual device - (phone)
- Users can utilize 2FA with an authenticator app or by mobile SMS’s
- Authenticator apps provide users with a **random token** every 30 seconds or so, which is used to login to their account

2FA - authenticator apps...

Google authenticator

Authy

Microsoft Authenticator

An important note…

An important note!

- **Integrating 2FA in Django is challenging!**
- So please be sure to watch each step carefully and try not to skip ahead because there is a lot involved!
- But don’t worry, we will do everything slowly and in the easiest way possible

Practical

- Integrate Two-Factor Authentication 2FA - Part 1

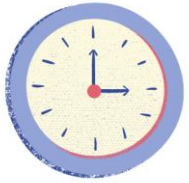
SECTION:

SESSION TIMEOUT

Session timeout

Session timeout

- Users often forget to log out of their accounts, hence leaving their account idle for several hours
- If they forget to logout in a public environment, anyone can simply use their computer and make devastating changes
- Therefore, users should be logged out automatically if they remain idle for too long



Practical

- Add a session timeout

SECTION:

MANAGE BRUTE FORCE ATTACKS

Manage brute force attacks

Managing brute force attacks

- A ‘brute-force’ attack is when another user attempts to login to your account by trying out multiple username and password combinations in the hope of guessing correctly
- These attackers use a ‘trial-and-error’ method

Practical

- Create an ‘account-locked’ page

Practical

- Manage brute force attacks

SECTION:

CREATING ENVIRONMENT VARIABLES

Creating environment variables

Environment variables

- An environment variable is a variable whose value is set outside of a program
- It is important to utilize environment variables in order to keep our sensitive data from our application code
- **NEVER!** Deploy your application without setting environment variables for your sensitive data

Practical

- Create an environment variable

SECTION:

PASSWORD MANAGEMENT

Password management

Password management

- Your users will at some point either forget their password or want to change it, therefore it is important that they are able to change it themselves
- It is **good practice** to choose a strong password and change it every 90 days on rotation

Practical

- Allow your users to reset their passwords - Part 1

SECTION:

PRE-DEPLOYMENT SETTINGS

Pre-deployment settings

Pre-deployment settings

- Django has built-in security settings that we need to configure before we deploy our Django web application
- These settings range from enforcing **CSRF protection**, remembering to **turn off debug** straight to **preventing XSS attacks**

Practical

- Configure additional security settings

SECTION:

FILE HANDLING

File handling

File handling

- File handling is a large topic on its own and is concerned with managing the files that your user's upload according to a set of principles

File handling...

- Good file management principles to research and to integrate in your Django web application would include to:

- Only allow registered users to upload files
- Limit the **number of characters** in an uploaded file's name
- Limit the **size** of an uploaded file
- Rename a **user's** file name upon upload
- Validate the **file's extension** (.pdf) to ensure it isn't a type of malware

SECTION:

ADDITIONAL TIPS AND ADVICE

Additional tips and advice

Additional tips

- After you deploy your Django web application, please be sure to check its safety rating at the following websites:

Mozilla observatory

- Mozilla observatory allows you to paste your website URL into its vulnerability scanner
- Once the scan has been completed you will receive a score on an A - F rating with recommendations for improvements

- So, please be sure to check it out at:

https://observatory.mozilla.org/

DJ Checkup

- DJ checkup allows you to paste your website URL into its vulnerability scanner
- It is specifically designed to **check Django applications** for vulnerabilities and flaws

- So, please be sure to check it out at:

https://djcheckup.com/

SSL Trust

- SSL Trust allows you to paste your website URL into its vulnerability scanner
- It provides us with a detailed SSL security test and tests for a wide variety of typical web issues

- So, please be sure to check it out at:

https://www.ssltrust.com.au/ssl-tools/website-security-check

Securi SiteCheck

- Securi SiteCheck allows you to paste your website URL into its vulnerability scanner
- One of the most popular open-source web security tools out there, that checks for:

- Common website errors, malware and out-of-date software among other things

- So, please be sure to check it out at:

https://sitecheck.sucuri.net/

Research, research and research...

- Also be sure to do your own research and read blogs, and articles from individuals who have experience in deploying their Django web applications
- Just remember that no matter how much effort you put into securing your Django web application it will never be 100% secure, you should do at least the minimum and do as much as you can to ensure that your website is secure

SECTION:

THANK YOU!