

# Django

Copyright © 2025 by  
Robert M. Dondero, Ph.D  
Princeton University

# Objectives

- The lecture will provide *a taste of ...*
- The **Django** server-side web app framework
  - As a more powerful alternative to Flask
  - As a contrast with Flask

# Goal

- Goal of this lecture
  - Give you enough information about Django to:
    - Help you decide if you want to use it
    - Help you get started with it

# Agenda

- **Django overview**
- An example app
- Setting up
- Creating files and directories
- Creating the database
- Running the app
- The admin interface
- Concluding observations

# Django Overview

**Jean “Django”  
Reinhardt**  
1910-1952

Jazz guitarist  
and composer



# Django Overview

- **Who:** Adrian Holovaty & Jacob Kaplan-Moss (Benevolent Dictators for Life)
- **When:** 1995
- **What:** A heavyweight Python server-side web framework
- **Why:** Help web developers do more, use good style, and avoid security vulnerabilities



Holovaty

Kaplan-Moss

# Django Overview

## Django characteristics:

“Django lets you build deep, dynamic, interesting sites in an extremely **short time**. Django is designed to let you focus on the **fun**, interesting parts of your job while easing the pain of the repetitive bits. In doing so, it provides **high-level abstractions** of common Web-development patterns, shortcuts for frequent programming tasks, and clear conventions on how to solve problems. At the same time, Django tries to **stay out of your way**, letting you work outside the scope of the framework as needed.”

-- Holovaty and Kaplan-Moss

# Django Overview

- Why study Django?
  - Popular
  - Different
  - Reasonable for COS 333 project



# Agenda

- Django overview
- **An example app**
- Setting up
- Creating files and directories
- Creating the database
- Running the app
- The admin interface
- Concluding observations

# An Example App

- See **PennyDjango** app

```
$ python runserver.py 55555
```

```
After running the server, browse to this URL:
```

```
http://localhost:55555/pennyapp
```

```
The site administrator is rdondero with password xxxxxxxx
```

```
Watching for file changes with StatReloader
```

```
Performing system checks...
```

```
System check identified no issues (0 silenced).
```

```
August 20, 2024 - 12:05:02
```

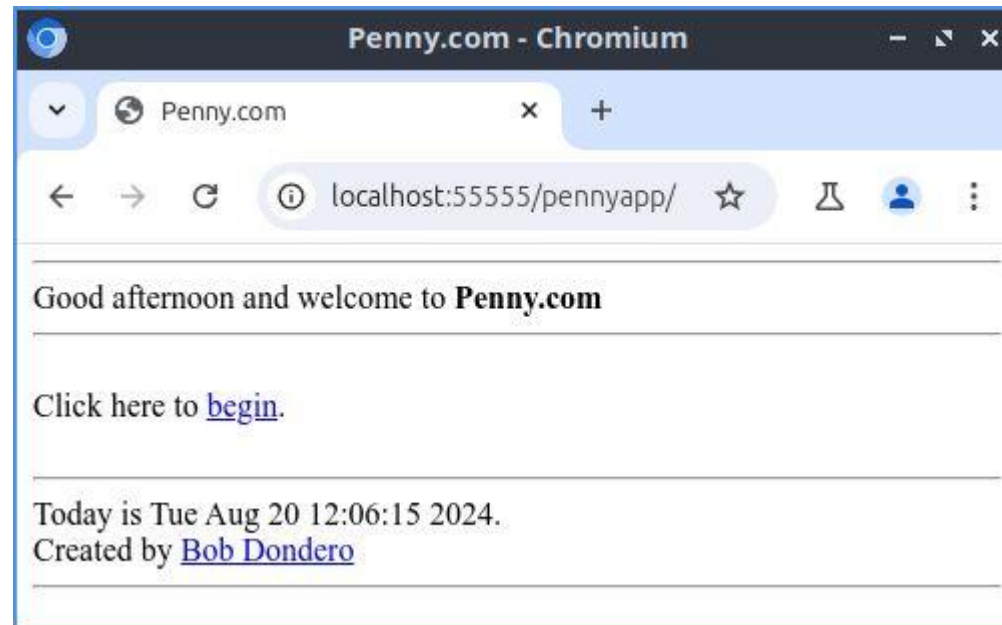
```
Django version 5.1, using settings 'pennyproj.settings'
```

```
Starting development server at http://0.0.0.0:55555/
```

```
Quit the server with CONTROL-C.
```

# An Example App

- See **PennyDjango** app



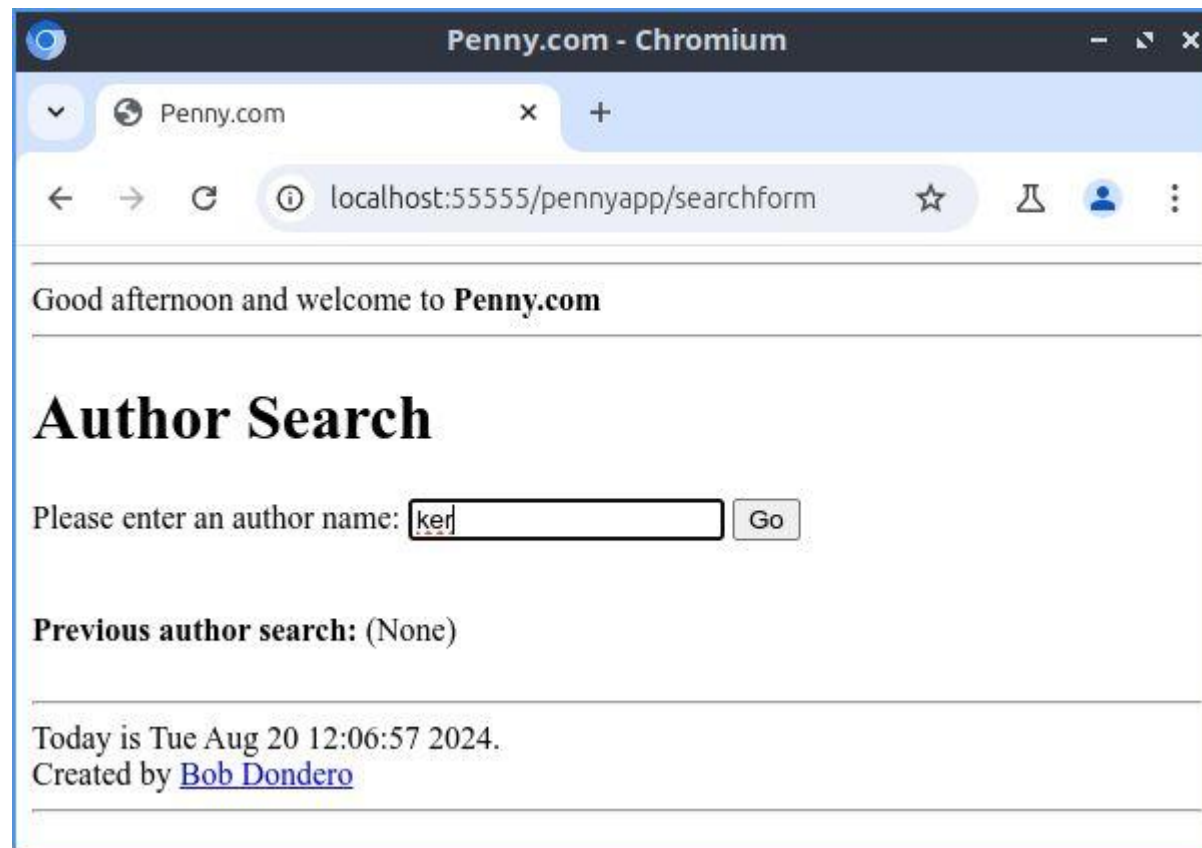
# An Example App

- See **PennyDjango** app (cont.)



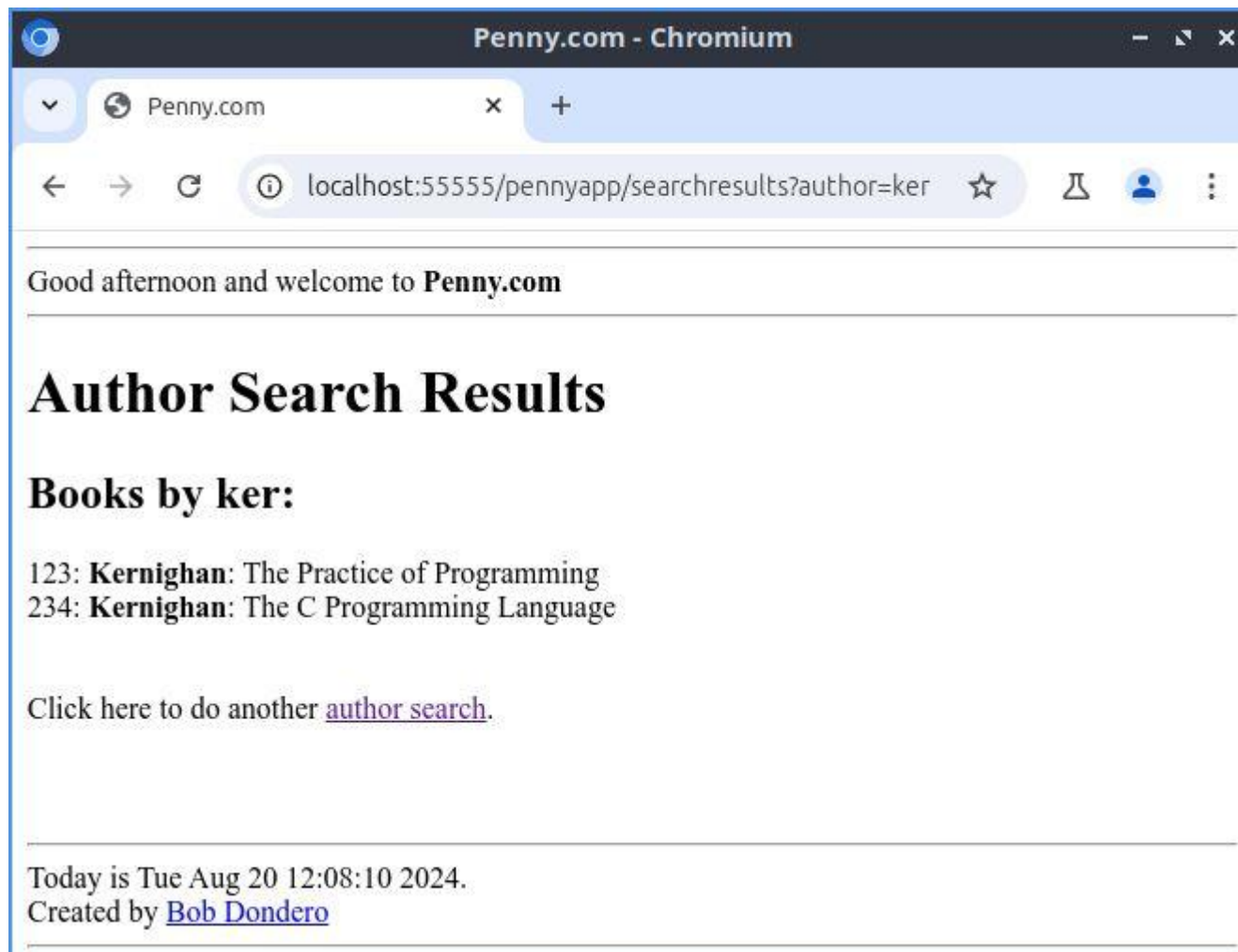
# An Example App

- See **PennyDjango** app (cont.)



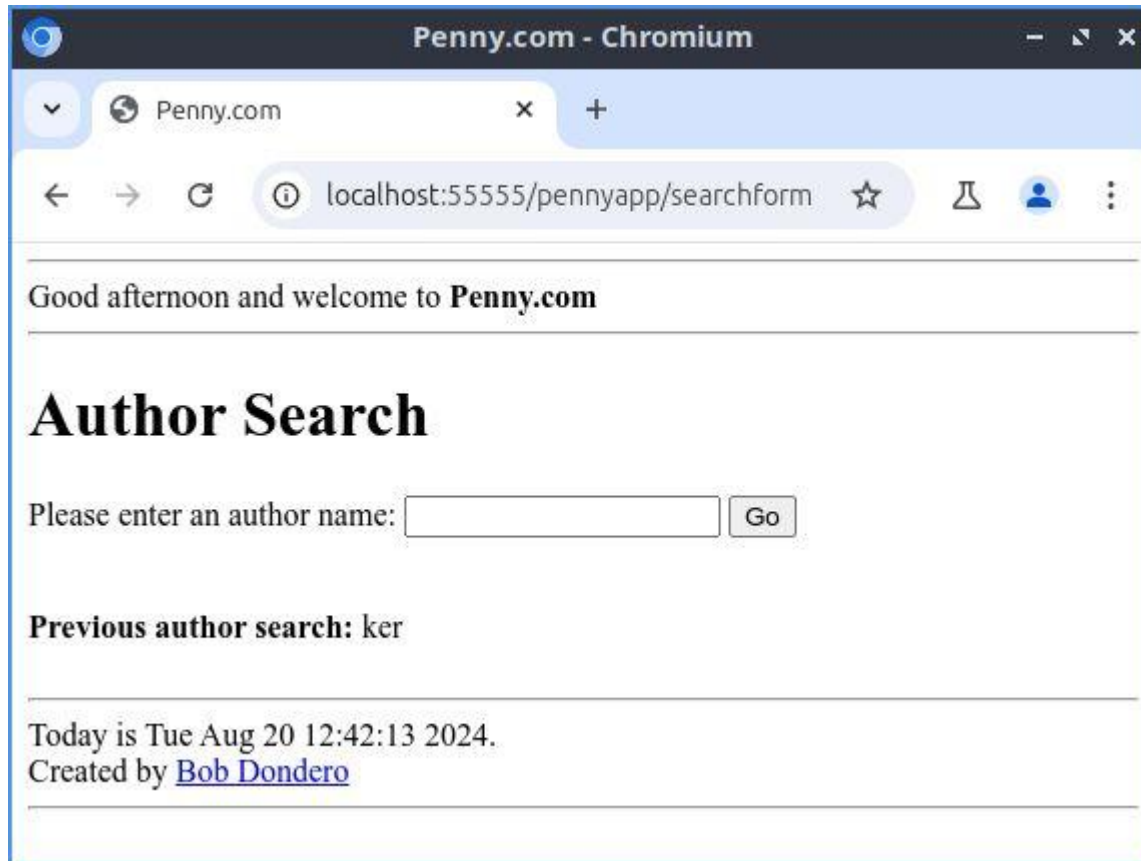
# An Example App

- See **PennyDjango** app (cont.)



# An Example App

- See **PennyDjango** app (cont.)



# Agenda

- Django overview
- An example app
- **Setting up**
- Creating files and directories
- Creating the database
- Running the app
- The admin interface
- Concluding observations



# Setting Up

- **Activate an appropriate Python virtual environment**
  - (OS specific)
- **Install Django**
  - `$ python -m pip install django`
- **Create and cd to some reasonable directory**
  - `$ mkdir PennyDjango`
  - `$ cd PennyDjango`

# Setting Up

- **Create a Django project**
  - `$ django-admin startproject pennyproj`
- **cd to your project directory**
  - `$ cd pennyproj`
- **Create an application within the project**
  - `$ python manage.py startapp pennyapp`

# Agenda

- Django overview
- An example app
- Setting up
- **Creating files and directories**
- Creating the database
- Running the app
- The admin interface
- Concluding observations

# Creating Files & Directories

Create/edit these files and directories:

```
PennyDjango/  
  pennyproj/  
    manage.py  
    penny.sql  
  pennyapp/  
    __init__.py  
    admin.py  
    apps.py  
    models.py  
    templates/  
      pennyapp/  
        footer.html  
        header.html  
        index.html  
        searchform.html  
        searchresults.html  
  
    tests.py  
    urls.py  
    views.py
```

```
PennyDjango/ (cont.)  
  pennyproj/ (cont.)  
    pennyproj/  
      __init__.py  
      asgi.py  
      settings.py  
      urls.py  
      wsgi.py  
      runserver.py
```

Boldface =>

Created or edited by Dondero

# Creating Files & Directories

```
PennyDjango/  
  pennyproj/  
    manage.py  
    penny.sql  
    pennyapp/  
      __init__.py  
      admin.py  
      apps.py  
      models.py  
      templates/  
        pennyapp/  
          footer.html  
          header.html  
          index.html  
          searchform.html  
          searchresults.html  
      tests.py  
      urls.py  
      views.py
```

```
PennyDjango/ (cont.)  
  pennyproj/ (cont.)  
    pennyproj/  
      __init__.py  
      asgi.py  
      settings.py  
      urls.py  
      wsgi.py  
      runserver.py
```

Created by Django

Makes Django management  
functionality available

# Creating Files & Directories

```
PennyDjango/  
  pennyproj/  
    manage.py  
    penny.sql  
    pennyapp/  
      __init__.py  
      admin.py  
      apps.py  
      models.py  
      templates/  
        pennyapp/  
          footer.html  
          header.html  
          index.html  
          searchform.html  
          searchresults.html  
      tests.py  
      urls.py  
      views.py
```

```
PennyDjango/ (cont.)  
  pennyproj/ (cont.)  
    pennyproj/  
      __init__.py  
      asgi.py  
      settings.py  
      urls.py  
      wsgi.py  
      runserver.py
```

Created by Dondero

SQL script

Used to populate DB

# Creating Files & Directories

```
PennyDjango/  
  pennyproj/  
    manage.py  
    penny.sql  
    pennyapp/  
      __init__.py  
      admin.py  
      apps.py  
      models.py  
      templates/  
        pennyapp/  
          footer.html  
          header.html  
          index.html  
          searchform.html  
          searchresults.html  
      tests.py  
      urls.py  
      views.py
```

```
PennyDjango/ (cont.)  
  pennyproj/ (cont.)  
    pennyproj/  
      __init__.py  
      asgi.py  
      settings.py  
      urls.py  
      wsgi.py  
      runserver.py
```

Created by Dondero

So need not remember  
how to start server!

# Creating Files & Directories

```
PennyDjango/  
  pennyproj/  
    manage.py  
    penny.sql  
    pennyapp/  
      __init__.py  
      admin.py  
      apps.py  
      models.py  
      templates/  
        pennyapp/  
          footer.html  
          header.html  
          index.html  
          searchform.html  
          searchresults.html  
      tests.py  
      urls.py  
      views.py
```

```
PennyDjango/ (cont.)  
  pennyproj/ (cont.)  
    pennyproj/  
      __init__.py  
      asgi.py  
      settings.py  
      urls.py  
      wsgi.py  
      runserver.py
```

Created by Django

Empty file  
pennyproj is a package



# Creating Files & Directories

```
PennyDjango/  
  pennyproj/  
    manage.py  
    penny.sql  
    pennyapp/  
      __init__.py  
      admin.py  
      apps.py  
      models.py  
      templates/  
        pennyapp/  
          footer.html  
          header.html  
          index.html  
          searchform.html  
          searchresults.html  
  
    tests.py  
    urls.py  
    views.py
```

```
PennyDjango/ (cont.)  
  pennyproj/ (cont.)  
    pennyproj/  
      __init__.py  
      asgi.py  
      settings.py  
      urls.py  
      wsgi.py  
      runserver.py
```

Created by Django

Unused

# Creating Files & Directories

```
PennyDjango/  
  pennyproj/  
    manage.py  
    penny.sql  
    pennyapp/  
      __init__.py  
      admin.py  
      apps.py  
      models.py  
      templates/  
        pennyapp/  
          footer.html  
          header.html  
          index.html  
          searchform.html  
          searchresults.html  
      tests.py  
      urls.py  
      views.py
```

```
PennyDjango/ (cont.)  
  pennyproj/ (cont.)  
    pennyproj/  
      __init__.py  
      asgi.py  
      settings.py  
      urls.py  
      wsgi.py  
      runserver.py
```

Created by Django  
Edited (2 lines) by Dondero

Specifies settings for pennyproj,  
Esp. DB settings

# Creating Files & Directories

```
PennyDjango/  
  pennyproj/  
    manage.py  
    penny.sql  
    pennyapp/  
      __init__.py  
      admin.py  
      apps.py  
      models.py  
      templates/  
        pennyapp/  
          footer.html  
          header.html  
          index.html  
          searchform.html  
          searchresults.html  
      tests.py  
      urls.py  
      views.py
```

```
PennyDjango/ (cont.)  
  pennyproj/ (cont.)  
    pennyproj/  
      __init__.py  
      asgi.py  
      settings.py  
      urls.py  
      wsgi.py  
      runserver.py
```

Created by Django  
Edited (1 line) by Dondero

Maps pennyproj URLs to  
pennapp URLs

# Creating Files & Directories

```
PennyDjango/  
  pennyproj/  
    manage.py  
    penny.sql  
    pennyapp/  
      __init__.py  
      admin.py  
      apps.py  
      models.py  
      templates/  
        pennyapp/  
          footer.html  
          header.html  
          index.html  
          searchform.html  
          searchresults.html  
      tests.py  
      urls.py  
      views.py
```

```
PennyDjango/ (cont.)  
  pennyproj/ (cont.)  
    pennyproj/  
      __init__.py  
      asgi.py  
      settings.py  
      urls.py  
      wsgi.py  
      runserver.py
```

Created by Django

Connects project to WSGI

# Creating Files & Directories

```
PennyDjango/  
  pennyproj/  
    manage.py  
    penny.sql  
    pennyapp/  
      __init__.py  
      admin.py  
      apps.py  
      models.py  
      templates/  
        pennyapp/  
          footer.html  
          header.html  
          index.html  
          searchform.html  
          searchresults.html  
      tests.py  
      urls.py  
      views.py
```

```
PennyDjango/ (cont.)  
  pennyproj/ (cont.)  
    pennyproj/  
      __init__.py  
      asgi.py  
      settings.py  
      urls.py  
      wsgi.py  
      runserver.py
```

Created by Django

Empty file

pennyapp is a Python package

# Creating Files & Directories

```
PennyDjango/  
  pennyproj/  
    manage.py  
    penny.sql  
    pennyapp/  
      __init__.py  
      admin.py  
      apps.py  
      models.py  
      templates/  
        pennyapp/  
          footer.html  
          header.html  
          index.html  
          searchform.html  
          searchresults.html  
      tests.py  
      urls.py  
      views.py
```

```
PennyDjango/ (cont.)  
  pennyproj/ (cont.)  
    pennyproj/  
      __init__.py  
      asgi.py  
      settings.py  
      urls.py  
      wsgi.py  
      runserver.py
```

Created by Django  
Edited by Dondero

Indicates that admin can  
manipulate Book DB table

# Creating Files & Directories

```
PennyDjango/  
  pennyproj/  
    manage.py  
    penny.sql  
    pennyapp/  
      __init__.py  
      admin.py  
      apps.py  
      models.py  
      templates/  
        pennyapp/  
          footer.html  
          header.html  
          index.html  
          searchform.html  
          searchresults.html  
  
      tests.py  
      urls.py  
      views.py
```

```
PennyDjango/ (cont.)  
  pennyproj/ (cont.)  
    pennyproj/  
      __init__.py  
      asgi.py  
      settings.py  
      urls.py  
      wsgi.py  
      runserver.py
```

Created by Django

Defines app name

# Creating Files & Directories

```
PennyDjango/  
  pennyproj/  
    manage.py  
    penny.sql  
    pennyapp/  
      __init__.py  
      admin.py  
      apps.py  
      models.py  
      templates/  
        pennyapp/  
          footer.html  
          header.html  
          index.html  
          searchform.html  
          searchresults.html  
  
      tests.py  
      urls.py  
      views.py
```

```
PennyDjango/ (cont.)  
  pennyproj/ (cont.)  
    pennyproj/  
      __init__.py  
      asgi.py  
      settings.py  
      urls.py  
      wsgi.py  
      runserver.py
```

Created by Django  
Edited heavily by Dondero

Defines DB schema



# Creating Files & Directories

```
PennyDjango/  
  pennyproj/  
    manage.py  
    penny.sql  
    pennyapp/  
      __init__.py  
      admin.py  
      apps.py  
      models.py  
      templates/  
        pennyapp/  
          footer.html  
          header.html  
          index.html  
          searchform.html  
          searchresults.html  
      tests.py  
      urls.py  
      views.py
```

```
PennyDjango/ (cont.)  
  pennyproj/ (cont.)  
    pennyproj/  
      __init__.py  
      asgi.py  
      settings.py  
      urls.py  
      wsgi.py  
      runserver.py
```

Created by Dondero

Define HTML templates  
Note Django templating  
language

# Creating Files & Directories

```
PennyDjango/  
  pennyproj/  
    manage.py  
    penny.sql  
    pennyapp/  
      __init__.py  
      admin.py  
      apps.py  
      models.py  
      templates/  
        pennyapp/  
          footer.html  
          header.html  
          index.html  
          searchform.html  
          searchresults.html  
  
      tests.py  
      urls.py  
      views.py
```

```
PennyDjango/ (cont.)  
  pennyproj/ (cont.)  
    pennyproj/  
      __init__.py  
      asgi.py  
      settings.py  
      urls.py  
      wsgi.py  
      runserver.py
```

Created by Django

Unused

# Creating Files & Directories

```
PennyDjango/  
  pennyproj/  
    manage.py  
    penny.sql  
    pennyapp/  
      __init__.py  
      admin.py  
      apps.py  
      models.py  
      templates/  
        pennyapp/  
          footer.html  
          header.html  
          index.html  
          searchform.html  
          searchresults.html  
  
      tests.py  
      urls.py  
      views.py
```

```
PennyDjango/ (cont.)  
  pennyproj/ (cont.)  
    pennyproj/  
      __init__.py  
      asgi.py  
      settings.py  
      urls.py  
      wsgi.py  
      runserver.py
```

Created by Django  
Edited by Dondero

Maps pennyapp URLs  
to functions

# Creating Files & Directories

```
PennyDjango/  
  pennyproj/  
    manage.py  
    penny.sql  
    pennyapp/  
      __init__.py  
      admin.py  
      apps.py  
      models.py  
      templates/  
        pennyapp/  
          footer.html  
          header.html  
          index.html  
          searchform.html  
          searchresults.html  
  
      tests.py  
      urls.py  
      views.py
```

```
PennyDjango/ (cont.)  
  pennyproj/ (cont.)  
    pennyproj/  
      __init__.py  
      asgi.py  
      settings.py  
      urls.py  
      wsgi.py  
      runserver.py
```

Created by Django  
Edited heavily by Dondero

Defines functions to handle  
HTTP requests

# Agenda

- Django overview
- An example app
- Setting up
- Creating files and directories
- **Creating the database**
- Running the app
- The admin interface
- Concluding observations

# Creating the Database

- **Create the database**
  - \$ `python manage.py makemigrations`
  - \$ `python manage.py migrate`
  - \$ `sqlite3 db.sqlite3`
  - `sqlite> .read penny.sql`
  - `sqlite> .quit`

# Creating the Database

- (Optional) Examine the database

```
$ sqlite3 db.sqlite3
SQLite version 3.45.1 2024-01-30 16:01:20
Enter ".help" for usage hints.
sqlite> .tables
auth_group                django_admin_log
auth_group_permissions    django_content_type
auth_permission           django_migrations
auth_user                 django_session
auth_user_groups         pennyapp_book
auth_user_user_permissions
sqlite> .schema pennyapp_book
CREATE TABLE IF NOT EXISTS "pennyapp_book" ("isbn" varchar(128)
NOT NULL PRIMARY KEY, "author" varchar(128) NOT NULL, "title"
varchar(128) NOT NULL);
sqlite> select * from pennyapp_book;
123|Kernighan|The Practice of Programming
234|Kernighan|The C Programming Language
345|Sedgewick|Algorithms in C
sqlite>
```

Note: DB has extra tables

# Agenda

- Django overview
- An example app
- Setting up
- Creating files and directories
- Creating the database
- **Running the app**
- The admin interface
- Concluding observations



# Running the App

- Run the test server
  - `$ python runserver.py 55555`
- Run the application
  - Browse to  
`http://localhost:55555/pennyapp`

# Agenda

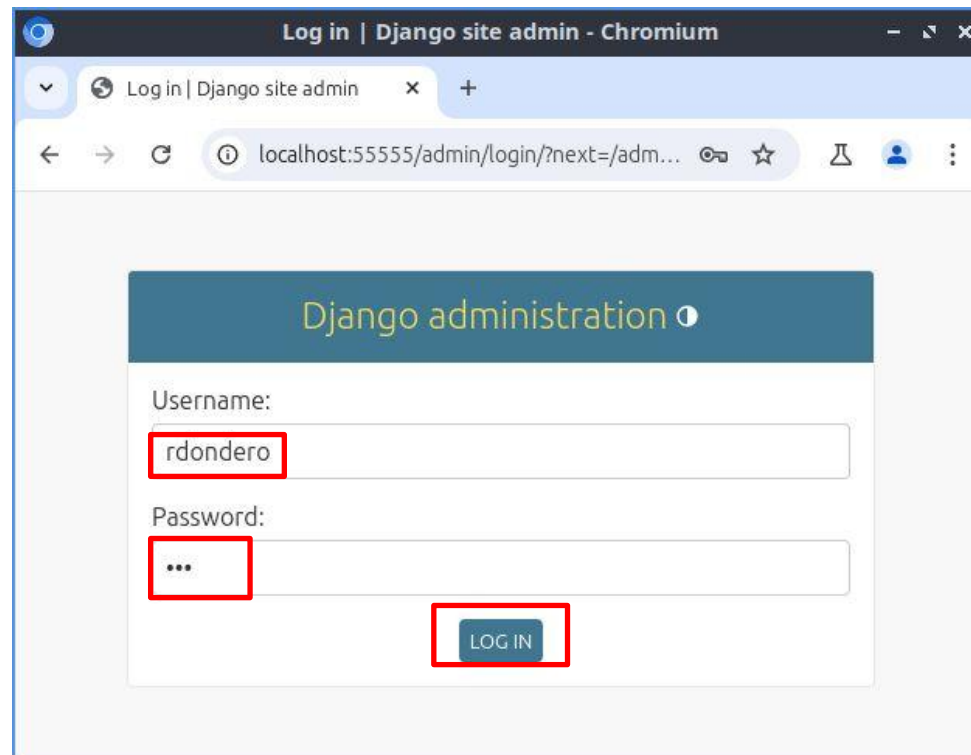
- Django overview
- An example app
- Setting up
- Creating files and directories
- Creating the database
- Running the app
- **The admin interface**
- Concluding observations

# The Admin Interface

- Create a superuser
  - \$ `python manage.py createsuperuser`
    - Enter a username and password

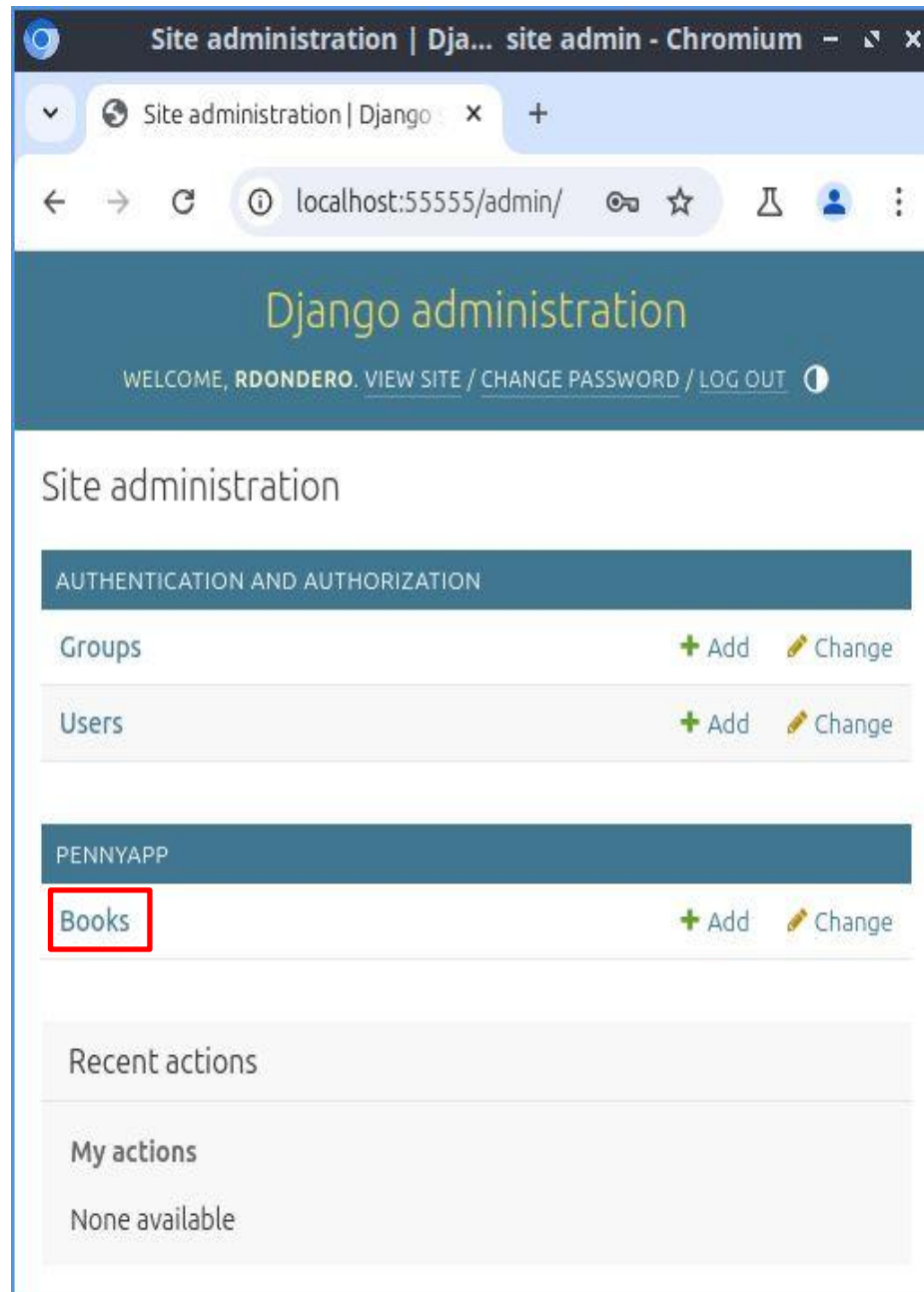
# The Admin Interface

- Run the *admin interface*
  - Browse to  
`http://localhost:55555/admin`



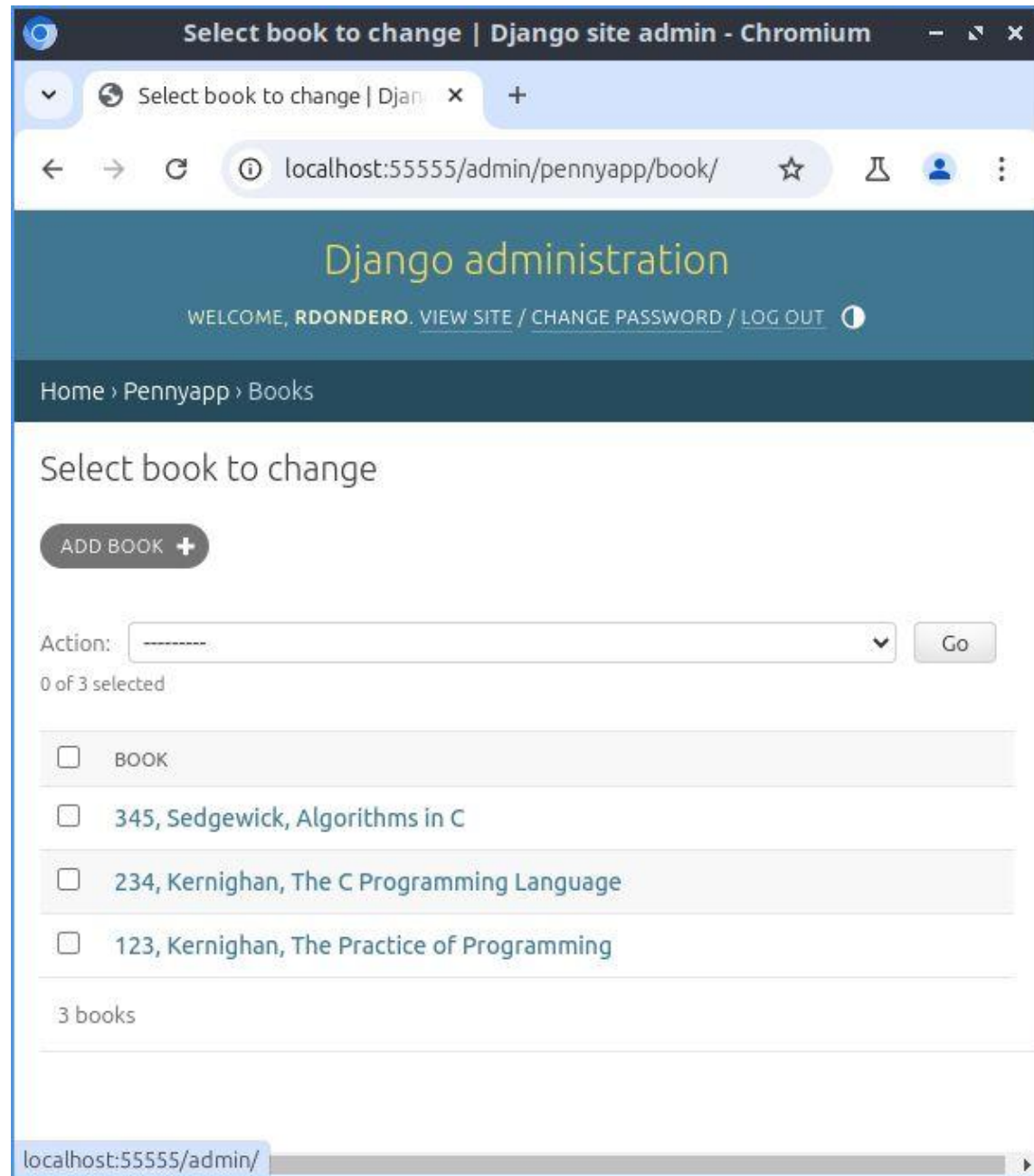
# The Admin Interface

- Run the *admin interface* (cont.)



# The Admin Interface

- Run the *admin interface* (cont.)



# Agenda

- Django overview
- An example app
- Setting up
- Creating files and directories
- Creating the database
- Running the app
- The admin interface
- **Concluding observations**

# Concluding Observations

- Django:
  - Separates URL from file name
  - Provides a template engine
  - Provides an object-relational mapper (ORM)
  - Implements MVC architecture
  - Automatically provides an admin interface



# Summary

- The lecture has provided *a taste of...*
  - The **Django** server-side web app framework
    - As a more powerful alternative to Flask
    - As a contrast with Flask