

# Django 4: Template Inheritance, Static Files & CDNs

**IN608: Intermediate Application Development Concepts**

**Kaiako: Tom Clark & Grayson Orr**

# Last Session's Content

- Forms
- Class-based views

# Today's Content

- Template inheritance
- Static files
- CDNs (Content delivery networks)

# Template Inheritance

# Template Inheritance

- In the `polls/templates/polls/` directory, create an `.html` file called `base.html`
- Allows you to a base “skeleton” template
- Defines blocks that child template can override
- The child template’s job is to fill the empty blocks with content

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>{% block title %}Polls{% endblock %}</title>
</head>
<body>
  <div id="content">
    {% block content %}{% endblock %}
  </div>
</body>
</html>
```

# Template Inheritance

- polls/templates/polls/index.html
- {% extends %} must be the first template tag

```
{% extends "polls/base.html" %}

{% block content %}
{% if latest_question_list %}
    <ul>
        {% for question in latest_question_list %}
            <li><a href="{% url 'polls:detail' question.id %}">{{ question }}</a></li>
        {% endfor %}
    </ul>
{% else %}
    <p>No polls are available.</p>
{% endif %}
{% endblock %}
```

# Template Inheritance

- polls/templates/polls/detail.html

```
{% extends "polls/base.html" %}

{% block title %}Polls | Detail{% endblock %}

{% block content %}
<h1>{{ question }}</h1>

{% if error_message %}<p><strong>{{ error_message }}</strong></p>{% endif %}

<form action="{% url 'polls:vote' question.id %}" method="post">
{% csrf_token %}
{% for choice in question.choice_set.all %}
    <input type="radio" name="choice" id="choice{{ forloop.counter }}" value="{{ choice.id }}">
    <label for="choice{{ forloop.counter }}">{{ choice }}</label><br>
{% endfor %}
<input type="submit" value="Vote">
</form>
{% endblock %}
```

# Template Inheritance

- polls/templates/polls/results.html

```
{% extends "polls/base.html" %}

{% block title %}Polls | Results{% endblock %}

{% block content %}
<h1>{{ question }}</h1>

<ul>
{% for choice in question.choice_set.all %}
  <li>{{ choice }} -- {{ choice.votes }} vote{{ choice.votes|pluralize }}</li>
{% endfor %}
</ul>

<a href="{% url 'polls:detail' question.id %}">Vote again?</a>
{% endblock %}
```



# Static Files

# Static Files

- In the `polls/` directory, create a directory called `static/`
- In the `static/` directory, create a directory called `polls/`
- In the `polls/` directory, create a `.css` file called `style`
- App structure:

```
polls/  
  __init__.py  
  admin.py  
  apps.py  
  migrations/  
  __init__.py  
  models.py  
  static/  
    polls/  
      style.css  
  templates/  
    polls/  
      base.html  
      detail.html  
      index.html  
      results.html  
  tests.py  
  urls.py  
  views.py
```

# Static Files

- polls/templates/polls/base.html
- {% load static %} - generates the absolute URL
- Resource: <https://docs.djangoproject.com/en/3.0/howto/static-files/>

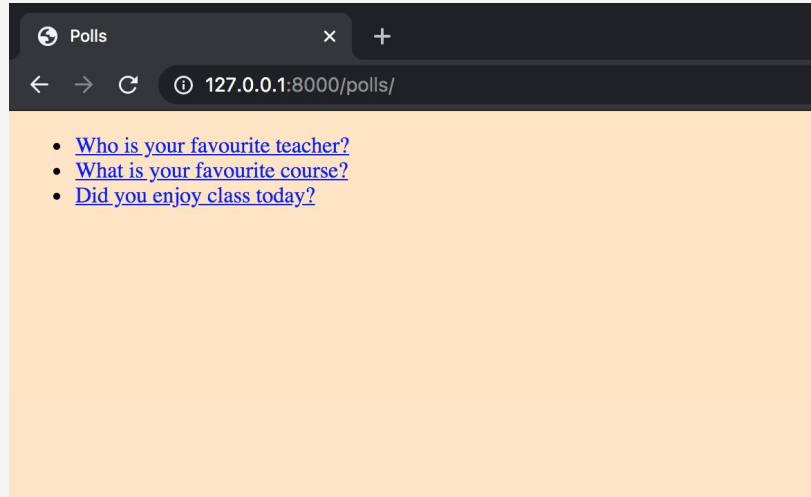
```
{% load static %}

<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <link rel="stylesheet" type="text/css" href="{% static 'polls/style.css' %}">
  <title>{% block title %}Polls{% endblock %}</title>
</head>
<body>
  <div id="content">
    {% block content %}{% endblock %}
  </div>
</body>
</html>
```

# Static Files

- polls/static/polls/style.css

```
body {  
    background-color: #FFE4C4; /* bisque */  
}
```



# CDNs

# CDNs

- Content delivery networks
- Distributed network of servers designed to load your site as quickly as possible
- Allows for quick transfer of assets, i.e., HTML, JS, CSS, image & video files
- Advantages:
  - Improved site load times
  - Reduced bandwidth costs
  - Increased content availability & redundancy
  - Improved site security
- Bootstrap CDN - <https://getbootstrap.com/docs/4.0/getting-started/download/#bootstrapcdn>
- TailWind CDN - <https://tailwindcss.com/docs/installation/#using-tailwind-via-cdn>

# Practical

# Programming Activity

- Checkout to master - `git checkout master`
- Create a new branch called 10-practical - `git checkout -b 10-practical`
- Copy 10-practical.ipynb from the course materials repository into your practicals repository
- Open up the Anaconda Prompt (it should be install on all lab computers) & `cd` to your practicals repository
- Run the following command: `jupyter notebook`