

Ziel

Erstelle eine Django-View, die prüft, ob heute **Mittwoch** ist.

- Wenn ja, steht fett: **YES**
- Wenn nein, steht fett: **NO**
 - Darunter in klein: *Today is Thursday* (oder ein anderer Wochentag)

Relevanter Python-Teil: `datetime`

Wie finde ich mit Python heraus, welcher Wochentag heute ist?

Schritt 1: `datetime` importieren

```
from datetime import datetime
```

`datetime` ist ein Modul in der Python-Standardbibliothek. Damit kann man mit Datum und Uhrzeit arbeiten.

Schritt 2: Aktuelles Datum holen

```
today = datetime.today()
```

Diese Zeile erzeugt ein `datetime`-Objekt, das das **aktuelle Datum und die aktuelle Uhrzeit** enthält.

Schritt 3: Den Wochentag herausfinden

```
weekday = today.strftime('%A')
```

`.strftime('%A')` wandelt das Datum in einen **voll ausgeschriebenen englischen Wochentag** um (zB. `'Monday'`, `'Tuesday'`, ...).

Django-View-Code (in `views.py`)

```
from django.shortcuts import render
from datetime import datetime

def is_it_wednesday(request):
    ...
    return render(request, 'check_wednesday.html', context)
```

Was passiert hier?

Codezeile	Bedeutung
<code>today = datetime.today()</code>	Holt das heutige Datum und die Uhrzeit
<code>weekday = today.strftime('%A')</code>	Wandelt das Datum in den ausgeschriebenen Wochentag um
<code>is_wednesday = (weekday == "Wednesday")</code>	Vergleicht den Tag mit „Wednesday“
<code>context = {...}</code>	Übergibt die Daten an das Template

Template: `check_wednesday.html`

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <title>Is it Wednesday?</title>
</head>
<body style="text-align: center; font-family: sans-serif; margin-top: 5rem;">
  <h1 style="font-size: 6rem;">
    ...
  </h1>
  <p style="font-size: 1.5rem; color: gray;">
    ...
  </p>
</body>
</html>
```

Letzter Schritt: `urls.py` einbinden (Beispiel)

```
from django.urls import path
from . import views

urlpatterns = [
    ...
]
```