

Позначити як виконано

Завдання 1

Складіть файл `urls.py`, який:

- направляє URL з `home/` у метод `views.home` і задає ім'я для цього URL як `'home-view'`;
- направляє URL з `'book/{назва глави}/` у метод `views.Book` разом з назвою розділу як аргумент `title` та задає ім'я для цього URL як `'book'`;
- передає запити, що починаються з `'lesson_2/'` до модуля `'lesson_2.urls'` разом із залишком URL.

Завдання 2

Складіть файл `urls.py`, який:

- § направляє URL з `'index/'` у метод `views.index` і задає ім'я для цього URL як `'index-view'`.
- § направляє URL з `'bio/{ім'я користувача}/` у метод `views.bio` разом з ім'ям користувача як аргумент `username` та задає ім'я для цього URL як `'bio'`.
- § передає запити, що починаються з `'lesson_1/'` в модуль `'lesson_1.urls'` разом із залишком URL.

Завдання 3

1. Register on [Weather Open API](#) website.
2. Get your APPID.
3. Make a test request and get a weather in your city, use [GUIDE](#) for it.
4. Add a `/weather/?city=...` route to your Django app.
5. Show a django template with this info to the user.
6. If city doesn't exist, show warning to the current user using `HttpResponse` with following text:

```
f'<script>alert("City {city} does not exist!");</script>'
```

You need to show following information to the current user.

Your user can be from any country like Ukraine, USA, Canada, Italy e.t.c.

Let's imagine that frontend sends us some information about user's city in the query params and we have to make a request using

"OpenWeather" API to get detailed info:

- * Country: ...
- * City: ...
- * Coords: ... (lon, lat)
- * Weather: ... (like Clouds e.t.c)
- * Temp: ... (**convert it into Celsius**)

EXAMPLE:

The following request must print information like the bellow one into the console.

(Replace `YOUR_APPID` with APPID from the site)

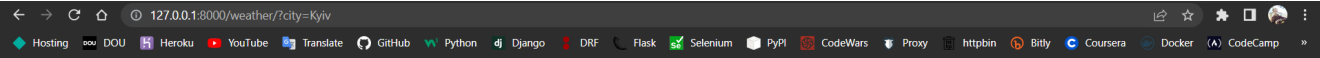
```
from requests import get

city = 'Kyiv'
# GET IT FROM https://home.openweathermap.org/api\_keys
key = "YOUR_APPID" # #####
# GET IT FROM https://home.openweathermap.org/api\_keys
weather_api = f'https://api.openweathermap.org/data/2.5/weather?q={city}&APPID={key}'
response = get(weather_api)
print(response.json())
```

```
# Expected output
```

{'coord': {'lon': 30.5167, 'lat': 50.4333}, 'weather': [{'id': 804, 'main': 'Clouds', 'description': 'overcast clouds', 'icon': '04n'}], 'base': 'stations', 'main': {'temp': 278.32, 'feels_like': 278.32, 'temp_min': 276.29, 'temp_max': 278.55, 'pressure': 1011, 'humidity': 93}, 'visibility': 10000, 'wind': {'speed': 0.45, 'deg': 127, 'gust': 1.79}, 'clouds': {'all': 99}, 'dt': 1667681413, 'sys': {'type': 2, 'id': 2003742, 'country': 'UA', 'sunrise': 1667624097, 'sunset': 1667658483}, 'timezone': 7200, 'id': 703448, 'name': 'Kyiv', 'cod': 200}

Expected template



Country: UA

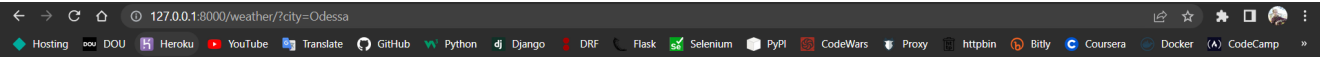
City: Kyiv

Longitude: 30.5167

Latitude: 50.4333

Weather: Clouds

Temperature: 5.4



Country: UA

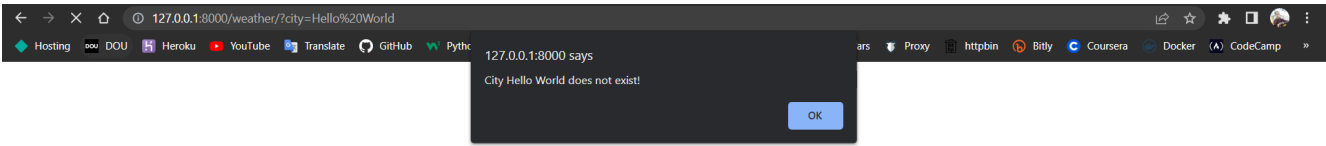
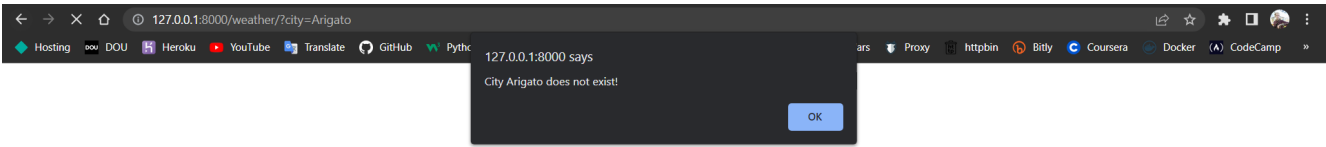
City: Odessa

Longitude: 30.7326

Latitude: 46.4775

Weather: Clouds

Temperature: 11.97



Здати роботу

Статус роботи

Статус роботи	Подання ще не подано
Статус оцінення	Не оцінено

 [Зверніться до служби підтримки сайту](#) 

Ви зашли під ім'ям [Morozov Denys](#) (Вийти)
[Підсумок збереження даних](#)