

Praca domowa 6

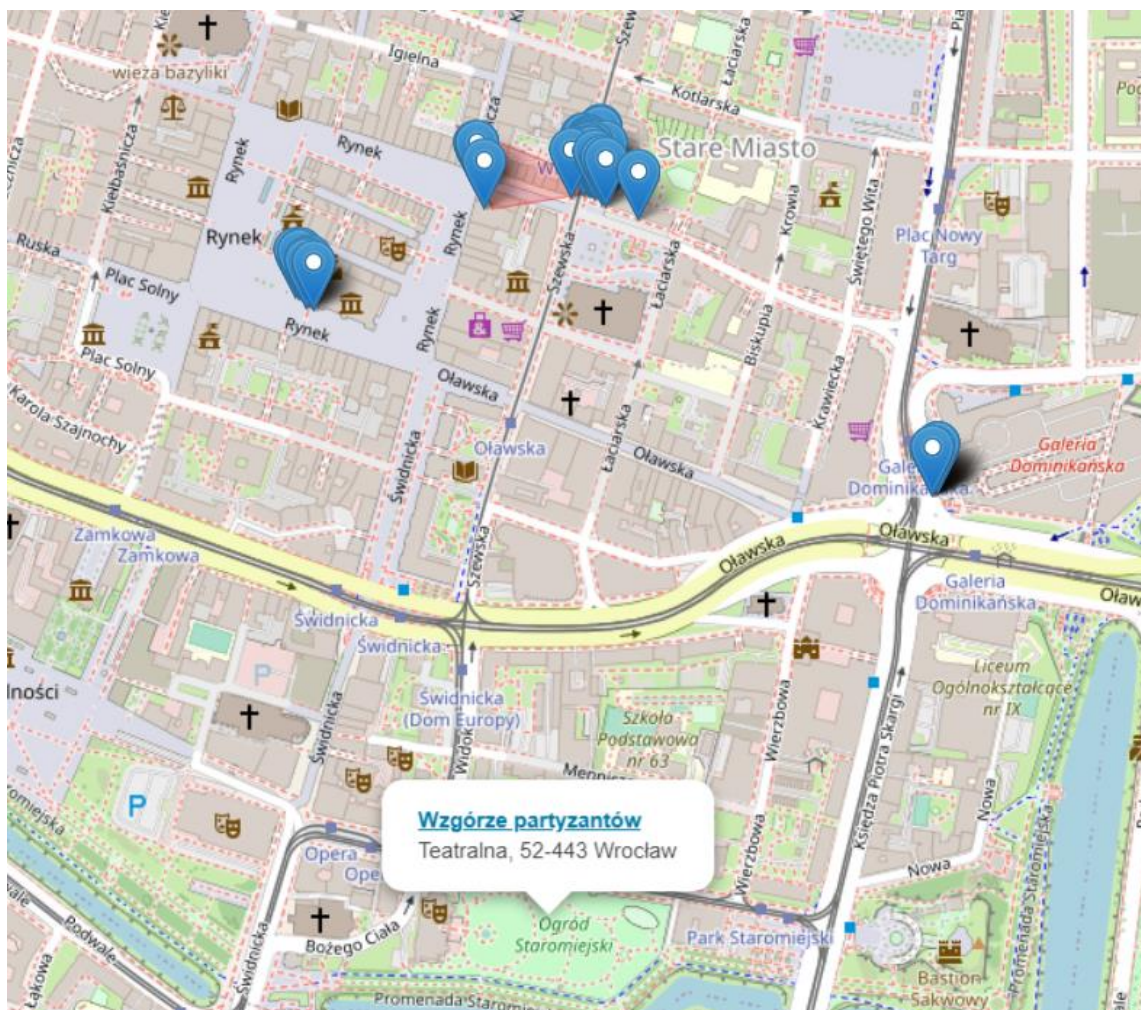
Franciszek Szczepaniak

Opis pakietu:

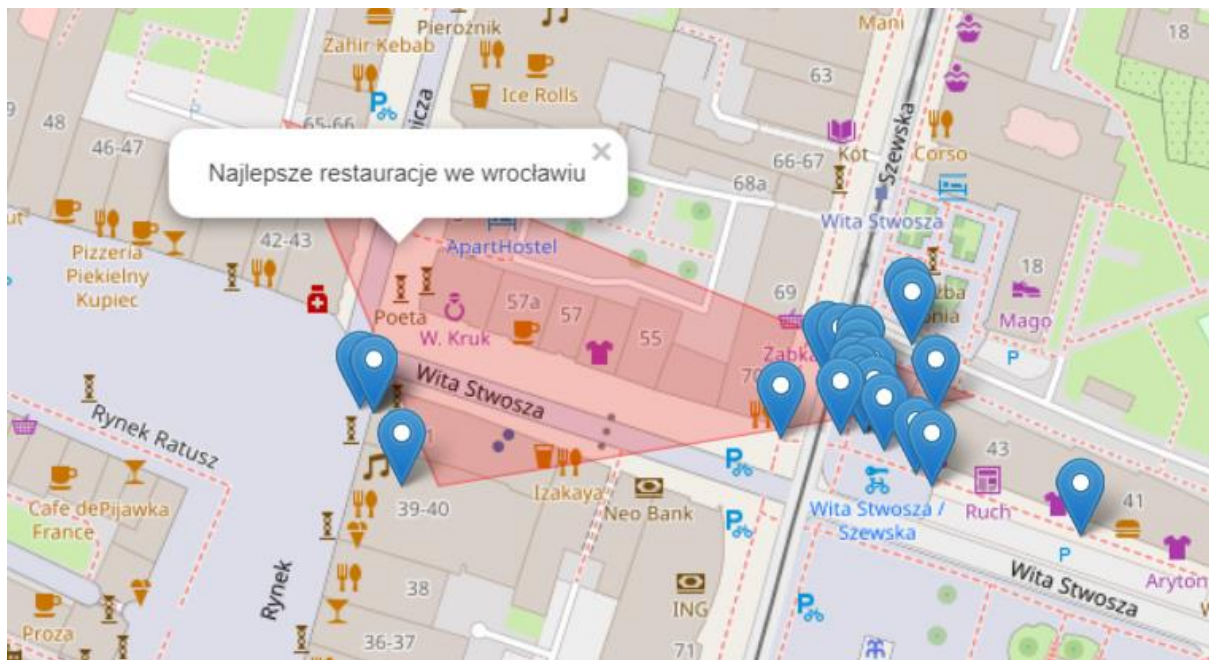
Pakiet, który chcę zaprezentować to pakiet leaflet. Służy od do tworzenia podstawowych map, wykorzystując współrzędne geograficzne. Do tego istnieje możliwość dodania markerów, które zaznaczają punkty na mapie np. wykorzystując dane z ramki danych. Makrerom tym można przypisać tytuły. Dodatkowo istnieje możliwość zaznaczenia pewnych punktów, za pomocą „chmurki” z tekstem. Do tekstów tych można dodać między dwoma punktami.

Stworzone wizualizacje:

Do stworzenia wizualizacji użyłem własnych zdjęć z wrocławia. Za pomocą pakietu „exifread” oraz „pandas” w pythonie, pobrałem metadane tych zdjęć, zmieniłem format współrzędnych geograficznych na standardowe i stworzyłem z nich tabelkę csv zawierającą długość i szerokość geograficzną zdjęć. Oto wyniki:



Niebieski tekst „Wzgórze partyzantów” po kliknięciu przekierowuje na stronę wikipedii zawierającą informacje o wzgórzu partyzantów.



Klikając na zaznaczony obszar wyświetli się tekst. Tak samo będzie w przypadku markerów.

Kod:

```
1 import exifread
2 import pandas as pd
3 import os
4 def main():
5     def _get_if_exist(data, key):
6         if key in data:
7             return data[key]
8
9         return None
10
11     def _convert_to_degress(value):
12         """
13         Helper function to convert the GPS coordinates stored in the EXIF to degress in float format
14
15         :param value:
16         :type value: exifread.utils.Ratio
17         :rtype: float
18         """
19         d = float(value.values[0].num) / float(value.values[0].den)
20         m = float(value.values[1].num) / float(value.values[1].den)
21         s = float(value.values[2].num) / float(value.values[2].den)
22
23         return d + (m / 60.0) + (s / 3600.0)
```

```

25 def get_exif_location(exif_data):
26     """
27     Returns the latitude and longitude, if available, from the provided exif_data (obtained through get_exif_data ab
28     """
29     lat = None
30     lon = None
31
32     gps_latitude = _get_if_exist(exif_data, 'GPS GPSLatitude')
33     gps_latitude_ref = _get_if_exist(exif_data, 'GPS GPSLatitudeRef')
34     gps_longitude = _get_if_exist(exif_data, 'GPS GPSLongitude')
35     gps_longitude_ref = _get_if_exist(exif_data, 'GPS GPSLongitudeRef')
36
37     if gps_latitude and gps_latitude_ref and gps_longitude and gps_longitude_ref:
38         lat = _convert_to_degree(gps_latitude)
39         if gps_latitude_ref.values[0] != 'N':
40             lat = 0 - lat
41
42         lon = _convert_to_degree(gps_longitude)
43         if gps_longitude_ref.values[0] != 'E':
44             lon = 0 - lon
45
46     return lat, lon

```

```

48 def get_exif_data(image_file):
49     with open(image_file, 'rb') as f:
50         exif_tags = exifread.process_file(f)
51     return exif_tags
52
53     # Create an empty data frame
54     lista = [0,0]
55     df = pd.DataFrame()
56
57     # Iterate through the photos in the folder
58     file_directory = "D:/zdjecia/wroclaw"
59     for file in os.listdir(file_directory):
60         # Open the file and extract the metadata
61         with open(f"D:/zdjecia/wroclaw/{file}", 'rb') as f:
62             tags = exifread.process_file(f)
63             lat, long = get_exif_location(tags)
64             lista[0] = long
65             lista[1] = lat
66
67
68
69     # Create a dictionary with the metadata
70     data = {i: lista[i] for i in range(2)}
71     # Add the metadata to the data frame
72     df = df.append(data, ignore_index=True)
73

```

```

74     # Display the data frame
75
76     df.drop([0], axis=0, inplace=True)
77     df.columns = ['long', 'lat']
78     df.to_csv('wspolrzedne.csv')
79
80     print(df)
81
82
83 if __name__ == "__main__":
84     main()

```

```

1 df <- as.data.frame(read.csv("D:/podstawy programowania/laby z poprzednich lat/wspolrzedne.csv"))
2
3
4 df<- df %>% select(long,lat)
5
6 library(leaflet)
7
8 content <- paste(sep = "<br/>",
9                 "<b><a href='https://pl.wikipedia.org/wiki/Park_Miko%C5%82aja_Kopernika_we_Wroc%C5%82awiu'"
10                 ">Wzgórze partyzantów</a></b>",
11                 "Teatralna, 52-443 Wrocław"
12             )
13
14 m <- leaflet(df) %>%
15   addTiles() %>%
16   addMarkers(~ long,~ lat, popup = "Wycieczka do Wrocławia") %>%
17   addPopups( 17.03441388231875, 51.10503537102607, content,
18             options = popupOptions(closeButton = FALSE)
19         )
20
21
22 map <- addPolygons(m, lng = c(17.03332145477918, 17.033736184490586, 17.035181773985375),
23                          lat = c(51.110886977024805, 51.11026667667644, 51.110418779424116),
24                          color = "red", fillColor = "red",
25                          opacity = 0.5, weight = 1,
26                          popup = "Najlepsze restauracje we Wrocławiu")
27 map
28
29

```