

Adresy a mapy

Návrh řešení:

1. Stažení dat – pomocí knihovny Scrapy jsem si stáhla data ze stránky:
https://www.izus.cz/kontakt/seznam_zakladnich_umeleckych_skol_v_cr/ ,
které jsem následně rozdělila. Z dat jsem si pomocí css selektorů vybrala název školy a adresu. Pomocí příkazu „scrapy crawl zus -O zus.json“ jsem si do souboru zus.json uložila potřebná data. Využila jsem tutoriál vytvořený přímo pro tuto knihovnu. Lze najít na adrese:
<https://docs.scrapy.org/en/latest/intro/tutorial.html>
2. Přepis adresy do geokódu – procházím jednotlivé adresy, které musím převést z textové formy do souřadnic. Na to využiji Google Maps API, dle návodu
<https://github.com/googlemaps/google-maps-services-python>
3. Vytvoření mapy s body – vytvořím pomocí knihovny folium mapu, střed nastavím na přibližný střed ČR.
(<https://python-visualization.github.io/folium/quickstart.html>) Na mapu poté přidám body podle souřadnic. Výsledek uložím do souboru s příponou html.
4. Vytvoření heat mapy dle krajů – Jako první krok bylo potřeba rozdělit pomyslnou mapu na kraje, tuto mapu jsem vytvořila na adrese: <https://mapshaper.org/> a postupovala jsem podle návodu na adrese:
<https://discourse.metabase.com/t/sharing-czech-geojson-maps/10476>. Druhým krokem byl výpočet počtů ZUŠ v jednotlivých krajích a následně vše zobrazit na mapě, dle návodu
<https://python-visualization.github.io/folium/quickstart.html#Choropleth-maps>

Výsledky a fakta:

```

1 import scrapy
2
3
4 class ZusScraper(scrapy.Spider):
5     name = "zus"
6
7     def start_requests(self):
8         url = 'https://www.izus.cz/kontakt/seznam_zakladnich_umeleckych_skol_v_cr/'
9         yield scrapy.Request(url=url, callback=self.parse) #stažení dat z adresy a zavolání metody parse
10
11     def parse(self, response):
12         for zus in response.css('tr'): #pro kazdy radek v tabulce
13             yield {
14                 'nazev_skoly': str(zus.css('zkraceny_nazev::text').get()).replace(u'\xa0', ' '),
15                 'adresa_sidla_skoly': str(zus.css('adresa_sidla_skoly > a::text').get()).replace(u'\xa0', ' '),
16             }
17
18     def parse(self, response):
19         for zus in response.css('tr')

```

Terminal: Local x +

Try the new cross-platform PowerShell <https://aka.ms/powershell>

PS C:\Users\simko\PycharmProjects\mapa_zus> cd .\zusScraper\
PS C:\Users\simko\PycharmProjects\mapa_zus\zusScraper> scrapy crawl zus -o zus.json

Stažení dat ze stránky pomocí knihovny Scrapy

```

[{"nazev_skoly": "None", "adresa_sidla_skoly": "None"},
{"nazev_skoly": "ZUŠ Adamov", "adresa_sidla_skoly": "Ronovská 281/12, 67904 Adamov"},
{"nazev_skoly": "ZUŠ Aš", "adresa_sidla_skoly": "Kostelní 42/12, 35201 Aš"},
{"nazev_skoly": "ZUŠ Bečov nad Teplou", "adresa_sidla_skoly": "Školní 338, 36464 Bečov nad Teplou"},
{"nazev_skoly": "ZUŠ V. Pichla, Bechyně", "adresa_sidla_skoly": "Klášteří 39, 39165 Bechyně"},
{"nazev_skoly": "ZUŠ Benátky nad Jizerou", "adresa_sidla_skoly": "Smetanova 21/5, 29471 Benátky nad Jizerou"},
{"nazev_skoly": "ZUŠ Benešov", "adresa_sidla_skoly": "Žižkova 471, 25601 Benešov"},
{"nazev_skoly": "SZUŠ Dr. Lidinské, Beroun", "adresa_sidla_skoly": "Tovární 66, 26601 Beroun"},
{"nazev_skoly": "ZUŠ V. Talicha Beroun", "adresa_sidla_skoly": "Husovo nám. 77/5, 26601 Beroun"},
{"nazev_skoly": "ZUŠ Bezručice", "adresa_sidla_skoly": "Školní 183, 34953 Bezručice"},
{"nazev_skoly": "ZUŠ G. Waltera, Bílina", "adresa_sidla_skoly": "Mírové náměstí 21/16, 41801 Bílina"},
{"nazev_skoly": "ZUŠ Bílovec", "adresa_sidla_skoly": "Pivovarská 124/24, 74301 Bílovec"},
{"nazev_skoly": "SZUŠ Blansko", "adresa_sidla_skoly": "Hybešova 240/1, 67801 Blansko"},
{"nazev_skoly": "ZUŠ Blansko", "adresa_sidla_skoly": "Zámek 3/3, 67801 Blansko"},
{"nazev_skoly": "ZUŠ Blatná", "adresa_sidla_skoly": "tř. J. P. Koubka 4, 38801 Blatná"},
{"nazev_skoly": "ZUŠ Blatná", "adresa_sidla_skoly": "Husova 222, 33601 Blatná"},
{"nazev_skoly": "ZUŠ Bohumín - Nový Bohumín", "adresa_sidla_skoly": "Žižkova 620, 73581 Bohumín - Nový Bohumín"},

```

Soubor zus.json po stažení dat

```

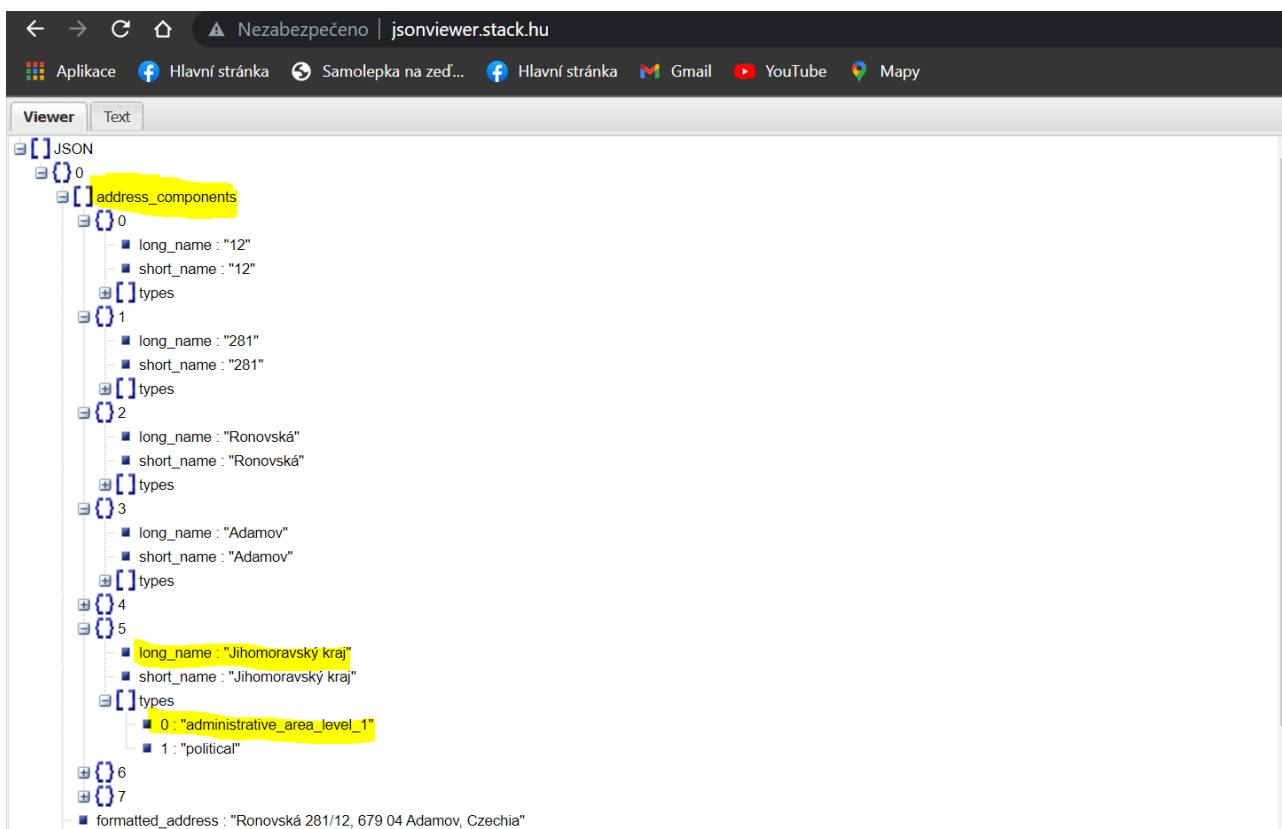
def vsechnyadresydoGeokodu():
    with open("./zusScraper/zus.json", encoding='utf-8') as jsonFile:
        jsonObject = json.load(jsonFile)
        jsonFile.close()

    seznamzus = []
    for zus in jsonObject:
        adresa = zus['adresa_sidla_skoly']
        longt, latt, kraj = geodekodovani(adresa)
        print(zus['nazev_skoly'])
        zus['longt'] = longt
        zus['latt'] = latt
        zus['kraj'] = kraj
        seznamzus.append(zus)

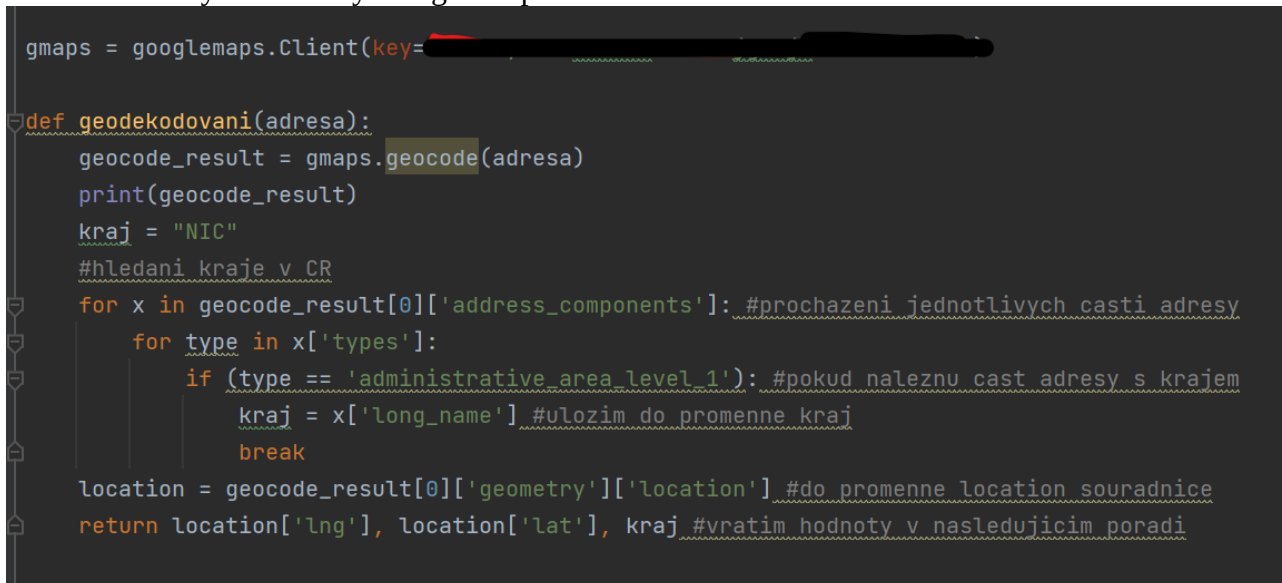
    with open('zusadresa.json', 'w', encoding='utf-8') as outfile:
        json.dump(seznamzus, outfile, indent=4, ensure_ascii=False)

```

Získání souřadnic z adresy



Struktura adresy z knihovny Google Maps API



Rozšíření adresy o další informace (souřadnice + název kraje)

```
ZusScraper.py x main.py x pocetVKraji.csv x SPH_KRAJ.json x zus.json x zusadresa.json x
3453 {
3454     "nazev_skoly": "ZUŠ Vracov",
3455     "adresa_sidla_skoly": "náměstí Míru 48, 69642 Vracov",
3456     "longt": 17.217736,
3457     "latt": 48.97956360000001,
3458     "kraj": "Jihomoravský kraj"
3459 },
3460 {
3461     "nazev_skoly": "ZUŠ Vrané nad Vltavou",
3462     "adresa_sidla_skoly": "Nádražní 144, 25246 Vrané nad Vltavou",
3463     "longt": 14.3840361,
3464     "latt": 49.9392375,
3465     "kraj": "Středočeský kraj"
3466 },
```

Soubor zus.json doplněný o data souřadnic a název kraje

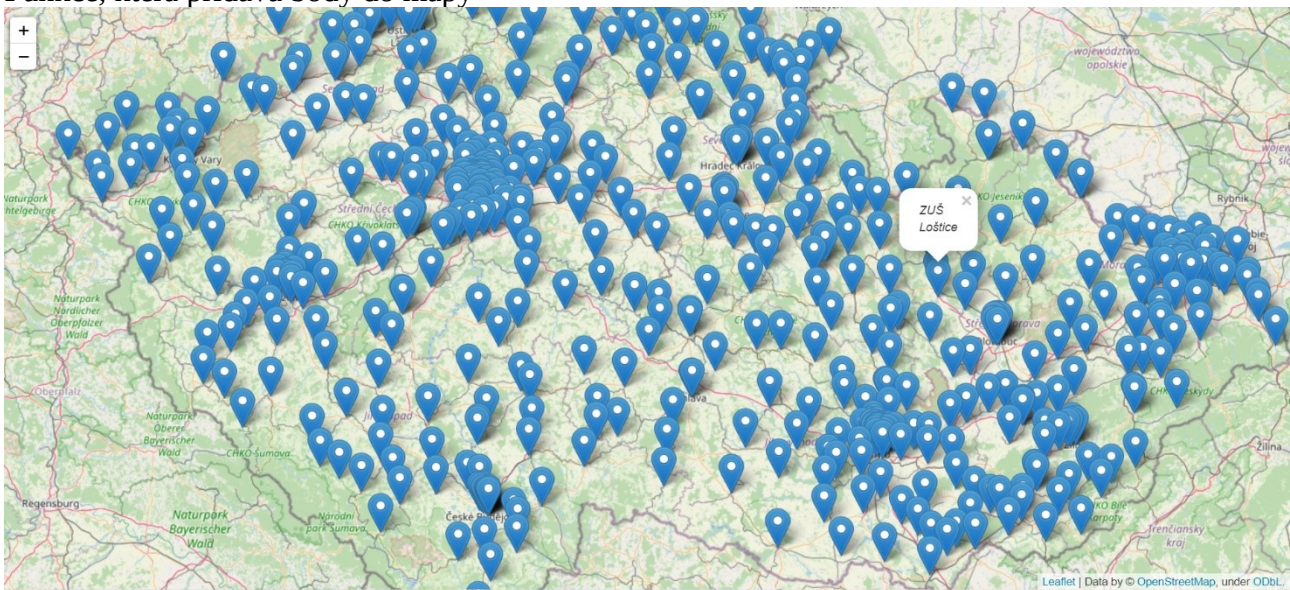
```
if __name__ == '__main__':
    #vsechnyadresydogeokodu() #staci spustit 1x
    with open("zusadresa.json", encoding='utf-8') as jsonFile: #otevře soubor zusadresa.json
        jsonObject = json.load(jsonFile)
        jsonFile.close()

    mapa = folium.Map(location=[49.7437572, 15.3386383], zoom_start=8) #nastaví střed zobrazení mapy
    pridatbodydomapy()
    mapa.save("index.html")
```

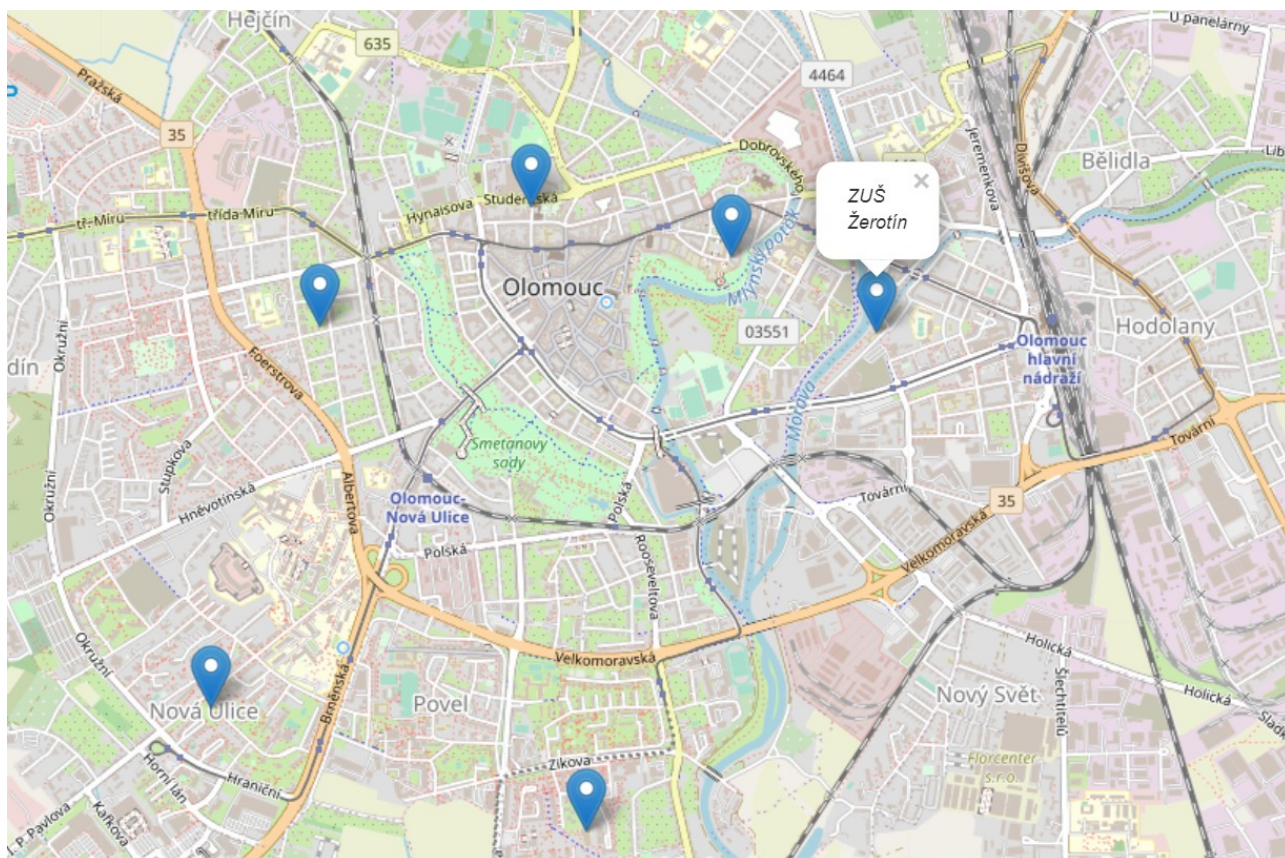
Hlavní funkce , která jednou zavolá převod adres. Dále vytvoří novou mapu a přidá body a uloží vše

```
def pridatbodydomapy():
    for zus in jsonObject:
        folium.Marker(
            [zus['latt'], zus['longt']], popup=f"<i>{zus['nazev_skoly']}</i>"
        ).add_to(mapa)
```

Funkce, která přidává body do mapy



Výsledná mapa



Mapa přiblížená

```

if __name__ == '__main__':
    #vsechny adresy dle geo kodu() #staci spustit 1x
    with open("zusadresa.json", encoding='utf-8') as jsonFile: #otevře soubor zusadresa.json
        jsonObject = json.load(jsonFile)
        jsonFile.close()

    mapa = folium.Map(location=[49.7437572, 15.3386383], zoom_start=8) #nastaví střed zobrazení mapy
    #přidat body do mapy()
    heatmapa()
    mapa.save("index.html")

```

Hlavní funkce, která vytvoří mapu. Zavolá funkci heatmapa a uloží

```

def heatmapa():
    pocet_zus_nakraj = { #seznam kraje s poctem zus
        'Hlavní město Praha': 0,
        'Středočeský kraj': 0,
        'Jihočeský kraj': 0,
        'Plzeňský kraj': 0,
        'Karlovarský kraj': 0,
        'Ústecký kraj': 0,
        'Liberecký kraj': 0,
        'Královéhradecký kraj': 0,
        'Pardubický kraj': 0,
        'Kraj Vysočina': 0,
        'Jihomoravský kraj': 0,
        'Olomoucký kraj': 0,
        'Moravskoslezský kraj': 0,
        'Zlínský kraj': 0
    }

    for zus in jsonObject: #pro každou zus - zvýším počet v kraji, kde se nachází
        if (zus['kraj'] != 'NIC'):
            pocet_zus_nakraj[zus['kraj']] += 1

    data = pd.DataFrame(list(pocet_zus_nakraj.items()), #přidání záhlaví do tabulky
                        columns=['kraj', 'pocet'])

```

Vytvoření pole s počty ZUŠ v krajích a naplnění

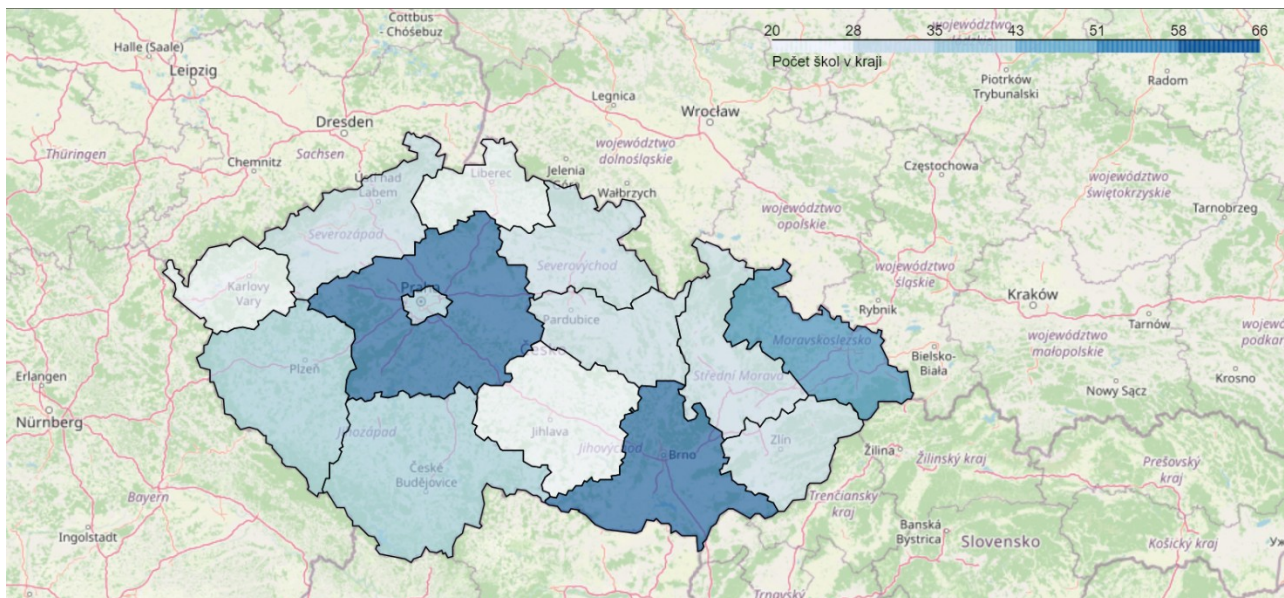
```

with open("./SPH_KRAJ.json", encoding='utf-8') as jsonFile:
    jsonKraje = json.load(jsonFile)
    jsonFile.close()

folium.Choropleth(
    geo_data=jsonKraje, #použiji načtený soubor SPH_KRAJ.json
    data=data, #tabulka s počty v jednotlivých krajích
    columns=['kraj', 'pocet'], #sloupce z tabulky
    key_on="feature.properties.NAZEV_NUTS", #klíč podle kterého se mapují data na tabulku
    legend_name="Počet škol v kraji",
).add_to(mapa) #přidání do mapy

```

Využití funkce z knihovny folium na vytvoření mapy



Výsledná heatmapa

Závěr:

Během práce na tomto projektu jsem první část zvládla bez potíží, druhá část, konkrétně heatmapa, mi však již trvala déle. Při geokódování jsem chtěla původně využít knihovnu <https://geocode.xyz/>, ta však již nezvládla ve volné verzi objem všech potřebných adres. Proto jsem se rozhodla pro geokódování za pomoci Google Maps API, kde již velký objem dat nebyl problém.

Myslím si, že jsem však své zadání splnila. Výsledkem jsou tedy 2 mapy, jedna mapa přibližitelná se značkami, kde se jednotlivé ZUŠ nachází a druhá heatmapa.