Homework

Alexander Heß

November 23, 2022

1 Task 1: Setup 0 points

Question: State the tag, image id and size of your modified Angular container. You can check this information with the command docker images.

Answer:

```
gis_assignment_3_frontend   latest     e630f6bbf901   19 hours ago   636MB
```

2 Task 2: Bringing it all together 15 points

2.1 Database

1. Write a SQL query that, given an amenity type, returns the name, latitude, and longitude of all amenities of the given type in the city of Konstanz. Inside server.py following query

```
    WITH konstanz AS (
    SELECT way
    FROM planet_osm_polygon
    WHERE admin_level='8' and name = 'Konstanz'
)
SELECT points.name, ST_Y(points.way) as latitude , ST_X(points.way) as longitude
from planet_osm_point points join konstanz on st_contains(konstanz.way, points.way)
where points.amenity='{query_name}'
""".format(query_name=amenity)
```

2.2 Backend

1. Create a new endpoint in the backend, that takes an amenity type as a POST request in JSON format (see: https://flask.palletsprojects.com/en/1.1.x/api/#flask.Request.json on how to get this information) and uses the query from 1.) to return all amenities of that type in the city of Konstanz.

```
    @app.route('/get-amenity', methods=["POST"])
def getAmenity():
    print("HELLO ")
    print(request.get_json(force=True))

    content = request.get_json(force=True)
    amenity = content.get('amenity')

    query = """WITH konstanz AS (
    SELECT way
    FROM planet_osm_polygon
    WHERE admin_level='8' and name = 'Konstanz'
)
SELECT points.name, ST_Y(points.way) as latitude , ST_X(points.way) as longitude
from planet_osm_point points join konstanz on st_contains(konstanz.way, points.way)
```

```
where points.amenity='{query_name}'
""".format(query_name=amenity)

    with psycopg2.connect(host="database", port=5432, dbname="gis_db", user="gis_user", password="gis
        with conn.cursor() as cursor:
            cursor.execute(query)
            results = cursor.fetchall()

    return jsonify([{'name': r[0], 'latitude': r[1], 'longitude': r[2]} for r in results]), 200
```

2.3 Frontend

1. Add a text input field and a button to the Settings component,in which the user can enter an amenity type and submit the query.

```
    <div class="amenity">
        <label for="amenity">Amenity: </label>
        <input
            id="amenity"
            name="amenity"
            type="text"
            placeholder="Restaurant"
            \#getAmenity
        />
        <button class="button"  (click)="addgetAmenity(getAmenity.value)">Submit</button>
  </div>
```

2. Propagate the button click and the entered amenity type to the App component

```
    // Function in the settings component
      addgetAmenity(name: string) {
        this.getAmenity.emit(name);
    }
    // Declare an Output for passing it to the parent Component with the getAmenity and define a
     @Output()
         getAmenity: EventEmitter<string> = new EventEmitter<string>();

    // extend the app-setting tag in the html of parent component
      <app-settings (markerAdded)="onAddMarker($event)" (pubsAdded)="onPubsAdded()" (getAmenity)

    // define a function in the app.component.ts, we have the value here now and use it to query
      onGetAmenityAdded(amenity: string) {
      console.log("INSIDE THE PARENT COMPONENT", amenity)
  }
```

3. Write a new method in the DataService to request the information about the specified amenities from the backend.

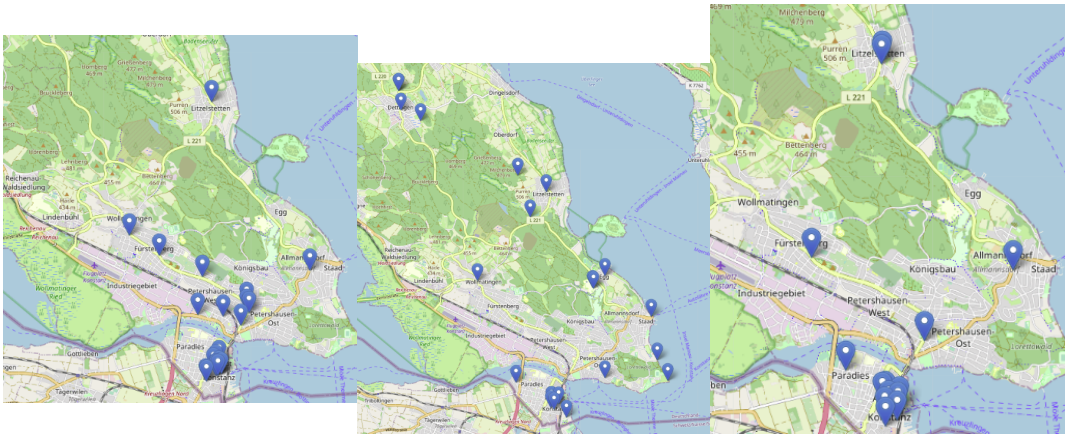
```
          /**
   * Get Specific Amenity from Backend
   */
   public getAmenity(amenity: string): Observable<
   any
  > {
   const body = `{"amenity":"${amenity}"}`
```

```
    console.log(body)
    const headers = { 'content-type': 'application/json'}
    const url = 'http://localhost:5000/get-amenity';
    return this.http.post(url, body, headers);
  }
```

4. Push the retrieved data into the Map component and add a marker for each requested amenity. in app component update the method as:

```
onGetAmenityAdded(amenity: string) {
// send to the backend
this.dataservice.getAmenity(amenity).subscribe((amenity) => {
  console.log(amenity)
  this.amenities = amenity
})
```

The result of the three queries from the exercise are displayed in the images below



References