Exploring the Connection Between Pedestrianisation and City Bike Usage Through Postcards

Creative Making: Big Data

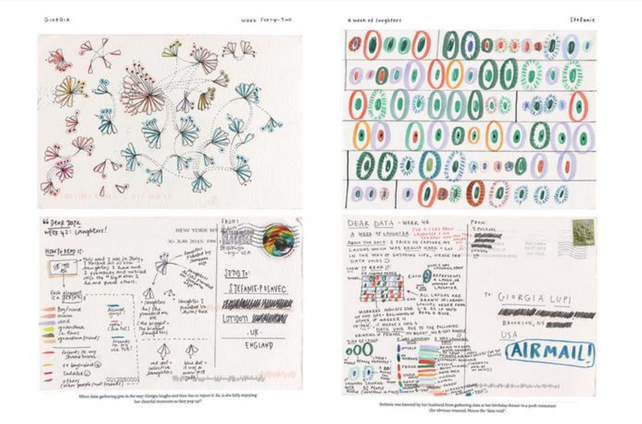
Table of contents:

Samia Tirike

ABSTRACT

My project will highlight the usage of city bikes worldwide and see if there's a connection between infrastructure and the use of bicycles within a city. I want to display if pedestrianisation equals more bike usage within the city. Pedestrianisation is creating an area where it is converted to areas for walking use. For the outcome, I will create hand-drawn postcards showing visual data from each city with the most bike usage to display my idea.

The idea of using postcards stuck with me because people use postcards to share experiences and information by posting them. And allows the reader to stop and experience. In contrast, biking allows you to experience the world around you physically. I decided to base my idea on city bikes and postcards because they both use movement to connect with their surroundings and build an appreciation of the world around them. Postcards enable the transmission of personal information without relying on digital means, which aligns with my view about being present.



Inspo: *Dear Data*, about two designers who pass on personal information about themselves every week on a postcard as visual datasets.

Visual representation of pedestrianisation

RESEARCH

Pedestrianisation is vital in creating an urban environment that contributes to sustainability. Creating livable cities is essential and impacts sustainability by reducing carbon emissions and allowing citizens to be healthy by walking everywhere. In the long term, it creates a better quality of life for locals and cities worldwide. They have implemented pedestrianisation in Copenhagen by making no-car zones and investments in cycling and public transport infrastructure (Yassin,2019).
By creating these enforcements, Copenhagen has lowered emissions and made areas more pedestrianisation friendly. This was a driving factor for my work because it started spaces for citizens to be more environmentally sustainable and efficient mobility.

More pedestrianised areas also mean increased use of bikes, which will raise bike-sharing usage. Bike sharing can also help locals get to other closer transportation links, e.g., buses, without using cars. Biking also has health benefits since it is a physical activity. Investments in bike sharing not only helps users around the local area but keep areas empty for pedestrians (DeMario,2009). Using bicycles and pedestrians on the road can signify how well a city's infrastructure is set up. By looking at city bike data, you can see how well a city is built for mobilisation.

DeMaio, P., 2009. Bike-sharing: History, impacts, models of provision, and future. Journal of public transportation, 12(4), pp.41-56.

Yassin, H.H., 2019. Livable city: An approach to pedestrianization through tactical urbanism. Alexandria engineering journal, 58(1), pp.251-259.

1) Printing data within network

```
In [45]: import requests
         import json
         import pybikes
         import numpy as np
         import pandas

In [2]: headers = {'Accept': 'application/json'}
        r = requests.get('http://api.citybik.es/v2/networks', headers=headers)

        # Convert data to JSON format
        data = r.json()
        #client = citybikes.Client()


        # Extract the info and records
        networks = data['networks']
        location = data ['networks']

In [3]: print(data)
```


{'networks': [{'company': ['ЗАО «СитиБайк»'], 'href': '/v2/networks/velobike-moscow', 'id': 'velobike-moscow', 'location': {'city': 'Moscow', 'country': 'RU', 'latitude': 55.75, 'longitude': 37.616667}, 'name': 'Velobike'}, {'company': ['Urban Infrastructure Partner'], 'href': '/v2/networks/baerum-bysykkel', 'id': 'baerum-bysykkel', 'location': {'city': 'Bærum', 'country': 'NO', 'latitude': 59.89455, 'longitude': 10.546343}, 'name': 'Bysykkel'}, {'company': ['Comunicare S.r.l.'], 'href': '/v2/networks/bicincitta-siena', 'id': 'bicincitta-siena', 'location': {'city': 'Siena', 'country': 'IT', 'latitude': 43.3186, 'longitude': 11.3306}, 'name': 'Bicincittà', 'source': 'https://www.bicincitta.com/frmLeStazioni.aspx?ID=202'}, {'company': ['Cyclopolis Systems'], 'href': '/v2/networks/cyclopolis-maroussi', 'id': 'cyclopolis-maroussi', 'location': {'city': 'Maroussi', 'country': 'GR', 'latitude': 38.0568722388, 'longitude': 23.8083299536}, 'name': 'Cyclopolis'}, {'company': ['Groundwork', 'Slough Borough Council', 'ITS'], 'href': '/v2/networks/cycle-hire-slough', 'id': 'cycle-hire-slough', 'location': {'city': 'Slough', 'country': 'GB', 'latitude': 51.51135, 'longitude': -0.591562}, 'name': 'Cycle Hire'}, {'company': ['Cyclopolis Systems'], 'href': '/v2/networks/cyclopolis-nafplio', 'id': 'cyclopolis-nafplio', 'location': {'city': 'Nafplio', 'country': 'GR', 'latitude': 37.5639397319, 'longitude': 22.8093402872}, 'name': 'Cyclopolis'}, {'company': ['Comunicare S.r.l.'], 'href': '/v2/networks/bicincitta-parco-dei-colli-di-bergamo', 'id': 'bicincitta-parco-dei-colli-di-bergamo', 'location': {'city': 'Parco dei Colli di Bergamo', 'country': 'IT', 'latitude': 45.72295637032245, 'longitude': 9.649230073016383}, 'name': 'Bicincittà', 'source': 'https://www.bicincitta.com/frmLeStazioni.aspx?ID=203'}, {'company': ['Cyclopolis Systems'], 'href': '/v2/networks/cyclopolis-aigialeia', 'id': 'cyclopolis-aigialeia', 'location': {'city': 'Aigialeia', 'country': 'GR', 'latitude': 38.2511101325, 'longitude': 22.0821570196}, 'name': 'Cyclopolis'}, {'company': ['Cyclopolis Systems'], 'href': '/v2/networks/cyclopolis-marathon', 'id': 'cyclopolis-marathon', 'location': {'city': 'Marathon', 'country': 'GR', 'latitude': 38.0855680022, 'longitude': 23.9776389963}, 'name':

figure 1

```
In [6]: for networks in data['networks']:
            print(networks['location'])
```


{'city': 'Moscow', 'country': 'RU', 'latitude': 55.75, 'longitude': 37.616667}
{'city': 'Bærum', 'country': 'NO', 'latitude': 59.89455, 'longitude': 10.546343}
{'city': 'Siena', 'country': 'IT', 'latitude': 43.3186, 'longitude': 11.3306}
{'city': 'Maroussi', 'country': 'GR', 'latitude': 38.0568722388, 'longitude': 23.8083299536}
{'city': 'Slough', 'country': 'GB', 'latitude': 51.51135, 'longitude': -0.591562}
{'city': 'Nafplio', 'country': 'GR', 'latitude': 37.5639397319, 'longitude': 22.8093402872}
{'city': 'Parco dei Colli di Bergamo', 'country': 'IT', 'latitude': 45.72295637032245, 'longitude': 9.649230073016383}
{'city': 'Aigialeia', 'country': 'GR', 'latitude': 38.2511101325, 'longitude': 22.0821570196}
{'city': 'Marathon', 'country': 'GR', 'latitude': 38.0855680022, 'longitude': 23.9776389963}
{'city': 'Mantova', 'country': 'IT', 'latitude': 45.15866370764634, 'longitude': 10.783718973016335}
{'city': 'Nea Smyrni', 'country': 'GR', 'latitude': 37.9381, 'longitude': 23.7126}
{'city': 'Moschato-Tavros', 'country': 'GR', 'latitude': 37.9538991434, 'longitude': 23.6824746661}
{'city': 'Arxaia Olympia', 'country': 'GR', 'latitude': 37.6419700934, 'longitude': 21.6247265179}
{'city': 'Kιato', 'country': 'GR', 'latitude': 38.0132674508, 'longitude': 22.7493819902}
{'city': 'Assemini', 'country': 'IT', 'latitude': 39.28956306007679, 'longitude': 9.006601555294765}
{'city': 'Rhodes', 'country': 'GR', 'latitude': 36.4509118, 'longitude': 28.2246966}
{'city': 'Florina', 'country': 'GR', 'latitude': 40.8016111, 'longitude': 21.4245222}
{'city': 'Tortolì', 'country': 'IT', 'latitude': 39.92664489852196, 'longitude': 9.654953570777934}
{'city': 'Limnos', 'country': 'GR', 'latitude': 39.8747861, 'longitude': 25.0586722}

figure 2

For the data scraping, I used the Python libraries: requests, JSON, numpy, counter, and Excel to sort values. And to collect data, I used the CityBikes API http://api.citybik.es/v2/ to manage my scrape my data on bike-sharing within different cities.

My initial thought process was printing all the data out within ['networks'] to show what I could work with (figure 1). I found fields such as id, country, company and name that would all be useful later on. Using the requests, I print out all the city bike data worldwide. Then, I processed within networks the locations of the stations (figure 2). What helped me massively was using for loops to produce data constantly. I used this code to filter out the results so I can only see the enclosed bracket of location, which outputs the city, country, alpha-2 code and coordinates of the countries' locations.

2) Counting countries and sorting values

```
In [8]: from collections import Counter

        # Sort the networks by country
        networks_by_country = sorted(data["networks"], key=lambda x: x["location"]["country"])

        # Print the networks alphabetically by country
        for network in networks_by_country:
            print(network["location"]["country"], network["location"]["city"])

            #1.sort out the data alphabetically  /done
            #2.see from data how many bike stations within that country /done
            #see if theres a correlation between how many bike stations and better infrastructure within that city

        #AE,AR,AT,AU,BA,BE,BR,CA,CH,CL,CØ,CY,CZ,DE,ES,FI,FR,GB,GR,HR,HU,IE,IL,IN,IS,IT,JP,KZ,LB,LT,LU,LV,MC,ME.MT,MX,N
        #top four germany,Czech Republic,Switzerland,france

        #collected data that has the most bike networks within that country
        #going to look at the most countries with the most networks
        #im going to illustrate the data of the cities within the countries

        AE Abu Dhabi
        AR Buenos Aires
        AR Rosario
        AT Neusiedler See
        AT St.Pölten
        AT Mödling
        AT Wachau
        AT Tulln
        AT Thermenregion
        AT Wr.Neustadt
        AT 10vorWien
        AT Innsbruck
        AT Hollabrunn
        AT WienerWald
        AT Serfaus
        AT Groß Enzersdorf
        AT Klagenfurt
        AT Wien
        AT Linz
        AT Kufstein

In [9]: z = ["AE", "AR", "AR", "AT", "AT", "AT", "AT", "AT", "AT", "AT", "AT", "AT", "AT", "AT", "AT", "AT", "AT", "AT", "AT"
```

figure 3

```
In [10]: Counter(z) #shows the highest usage of bikes in a country.
         #italy (IT) -123
         # germany (DE) - 73
         # Poland (PL) 55
         # United States (50) (US) - Country
         #suprised at the US data since the US doesn't priortise bikes and paedestinisation

Out[10]: Counter({'AE': 1,
                  'AR': 2,
                  'AT': 18,
                  'AU': 3,
                  'BA': 3,
                  'BE': 4,
                  'BR': 17,
                  'CA': 6,
                  'CH': 18,
                  'CL': 2,
                  'CO': 1,
                  'CY': 3,
                  'CZ': 26,
                  'DE': 73,
                  'ES': 26,
                  'FI': 2,
                  'FR': 44,
                  'GB': 11,
                  'GE': 1,
                  'GR': 26,
                  'HR': 28,
                  'HU': 1,
                  'IE': 4,
                  'IL': 1,
                  'IN': 1,
                  'IS': 1,
                  'IT': 123,
                  'JP': 1,
                  'KZ': 3,
                  'LB': 1,
                  'LT': 1,
                  'LU': 2,
                  'LV': 1,
                  'MC': 1,
                  'ME': 1,
                  'MT': 1,
                  'MX': 4,
                  'NL': 2,
                  'NO': 5,
                  'NZ': 2,
                  'PL': 55,
```

figure 4

```
In [88]: c = collections.Counter(z)
         print(c)
         print(c.most_common()) #sorts countries from highest to lowest

         Counter({'IT': 123, 'DE': 73, 'PL': 55, 'US': 50, 'FR': 44, 'HR': 28, 'CZ': 26, 'ES': 26, 'GR': 26, 'AT': 18, 'CH'
         : 18, 'BR': 17, 'GB': 11, 'SI': 9, 'CA': 6, 'NO': 5, 'SK': 5, 'BE': 4, 'IE': 4, 'MX': 4, 'RO': 4, 'TR': 4, 'UA': 4
         , 'AU': 3, 'BA': 3, 'CY': 3, 'KZ': 3, 'RU': 3, 'SE': 3, 'AR': 2, 'CL': 2, 'FI': 2, 'LU': 2, 'NL': 2, 'NZ': 2, 'AE'
         : 1, 'CO': 1, 'GE': 1, 'HU': 1, 'IL': 1, 'IN': 1, 'IS': 1, 'JP': 1, 'LB': 1, 'LT': 1, 'LV': 1, 'MC': 1, 'ME': 1, '
         MT': 1, 'PT': 1, 'TH': 1, 'UK': 1})
         [('IT', 123), ('DE', 73), ('PL', 55), ('US', 50), ('FR', 44), ('HR', 28), ('CZ', 26), ('ES', 26), ('GR', 26), ('AT
         ', 18), ('CH', 18), ('BR', 17), ('GB', 11), ('SI', 9), ('CA', 6), ('NO', 5), ('SK', 5), ('BE', 4), ('IE', 4), ('MX
         ', 4), ('RO', 4), ('TR', 4), ('UA', 4), ('AU', 3), ('BA', 3), ('CY', 3), ('KZ', 3), ('RU', 3), ('SE', 3), ('AR', 2
         ), ('CL', 2), ('FI', 2), ('LU', 2), ('NL', 2), ('NZ', 2), ('AE', 1), ('CO', 1), ('GE', 1), ('HU', 1), ('IL', 1), (
         'IN', 1), ('IS', 1), ('JP', 1), ('LB', 1), ('LT', 1), ('LV', 1), ('MC', 1), ('ME', 1), ('MT', 1), ('PT', 1), ('TH
         , 1), ('UK', 1)]
```

figure 5

I scraped all the countries mentioned in the API's JSON. I decided to look into the countries so I could look at the individual station of their countries. I used the counter library in Python to get a bigger idea of what countries have bike stations and how many cities within that countries have bike stations. I also sorted the information by using the collection and counters modules which allowed me to sort my values from highest to lowest from a list. From what I've scraped, I discovered that Italy, Germany, Poland and the United States are the top four countries mentioned the most within the CityBikes API. What shocked me was that the United States was in the top four because they need to prioritise pedestrianisation. I also did this to improve my knowledge and explore connections between pedestrianisation and bike usage. So I decided to scrape more data and find more evidence on the highest countries with bike usage.

3) Looking at biking companies

```
In [12]: for networks in data['networks']:
             print(networks['name']) #shows dictionary

         #'Bicincittà': 41,'Cyclopolis': 11,
```

```
Velobike
Bysykkel
Bicincittà
Cyclopolis
Cycle Hire
Cyclopolis
Bicincittà
Cyclopolis
Cyclopolis
Bicincittà
Cyclopolis
Cyclopolis
Cyclopolis
Cyclopolis
Bicincittà
Cyclopolis
Cyclopolis
Bicincittà
Cyclopolis
Bicincittà
Bicincittà
Bicincittà
Coca-Cola Zero® Bikes
Coca-Cola Zero® Bikes
Pun Pun Bike Share
Coca-Cola Zero® Bikes
BBBike
Bicincittà
Bydgoski rower aglomeracyjny
Santander Cycles
WE-cycle
Bicincittà
WOW Citybike
StadtRAD Hamburg'
Tigullionbike
```

figure 6

```
In [13]: a = ["Velobike", "Bysykkel", "Bicincittà", "Cyclopolis", "Cycle Hire", "Cyclopolis
```

```
In [14]: Counter(a)
```

```
Out[14]: Counter({'Velobike': 4,
                  'Bysykkel': 5,
                  'Bicincittà': 41,
                  'Cyclopolis': 11,
                  'Cycle Hire': 1,
                  'Coca-Cola Zero® Bikes': 3,
                  'Pun Pun Bike Share': 1,
                  'BBBike': 1,
                  'Bydgoski rower aglomeracyjny': 1,
                  'Santander Cycles': 2,
                  'WE-cycle': 1,
                  'WOW Citybike': 1,
                  "StadtRAD Hamburg'": 1,
                  'Tigullionbike': 1,
                  'ARbike': 1,
                  "Vélib' Métropole": 1,
                  'Ascoli Piceno': 1,
                  'BiGi': 1,
                  'Giracleta': 1
```

figure 7

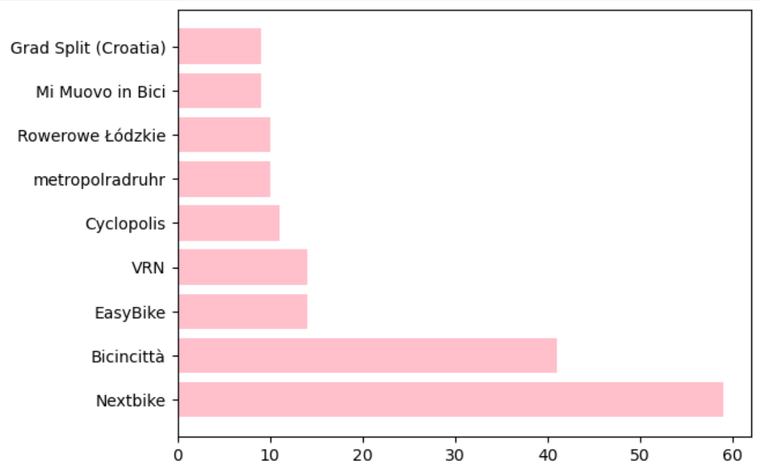


figure 8

```
In [82]: import collections


         c = collections.Counter(a)
         print(c)
         print(c.most_common()) # to sort values from highest to lowest
```

```
Counter({'Nextbike': 59, 'Bicincittà': 41, 'EasyBike': 14, 'VRN': 14, 'Cyclopolis': 11, 'metropolradruhr': 10, '
Rowerowe Łódzkie': 10, 'Mi Muovo in Bici': 9, 'Grad Split (Croatia)': 9, 'Bysykkel': 5, 'Velobike': 4, 'Coca-Col
a Zero® Bikes': 3, 'Baksi': 3, 'Mi Muovo in bici': 3, 'Lovesharing (Canary Islands)': 3, 'nextbike Ostrava': 3,
'Santander Cycles': 2, 'SoBi': 2, 'B-cycle': 2, 'VéloCité': 2, 'e.motion': 2, 'NomagoBikes': 2, 'Płocki Rower Mi
ejski (PRM) Poland': 2, 'nextbike Brno': 2, 'nextbike Praha': 2, 'YOY – San Luis Potosí': 2, 'nextbike Mladobole
slavsko': 2, 'MV-Rad': 2, 'VVT REGIORAD Tirol': 2, 'Cycle Hire': 1, 'Pun Pun Bike Share': 1, 'BBBike': 1, 'Bydgo
ski rower aglomeracyjny': 1, 'WE-cycle': 1, 'WOW Citybike': 1, "StadtRAD Hamburg'": 1, 'Tigullionbike': 1, 'ARbi
ke': 1, "Vélib' Métropole": 1, 'Ascoli Piceno': 1, 'BiGi': 1, 'Girocleta': 1, 'Alba': 1, 'BikeMi': 1, 'Biella':
1, 'Chivasso': 1, 'Bizi': 1, 'EcoBici': 1, 'Bici in Busto': 1, 'Velo Antwerpen': 1, 'Cuneo': 1, 'Malmö by bike':
1, 'Bici las Condes': 1, 'BiciMAD': 1, 'FRee Bike sharing': 1, 'lecce': 1, 'Manerba in Bici': 1, 'Mar e Bici': 1
, 'Novara': 1, 'ThessBike': 1, 'AthensBikes': 1, 'GoodBike': 1, 'CicloSampa': 1, 'Bicicoruña': 1, 'BiciFerrolTer
ra': 1, 'Curtin Bike Share': 1, 'Monash BikeShare': 1, 'Bixi': 1, 'Bike Share Toronto': 1, 'ArborBike': 1, 'Pavi
a in bici': 1, 'Bici Perugia': 1, 'Austin B-cycle': 1, 'Bike Chattanooga': 1, 'Pinerolo': 1, 'BIKETOWN': 1, 'BRi
teBikes': 1, 'Boise GreenBike': 1, 'Boulder B-cycle': 1, 'Broward B-cycle': 1, 'Bublr Bikes': 1, 'Reddy Bike Sha
re': 1, 'Capital BikeShare': 1, 'Charlotte B-cycle': 1, 'Saluzzo': 1, 'Cincy Red Bike': 1, 'Savigliano': 1, 'Cit
i Bike': 1, 'Coast Bike Share': 1, 'Terni': 1, 'CoGo': 1, 'Denver B-cycle': 1, 'Divvy': 1, 'Tirano in Bici': 1,
'El Paso B-cycle': 1, 'Fort Worth Bike Sharing': 1, 'Udine': 1, 'Great Rides Bike Share': 1, 'GREENbike': 1, 'Gr
id Bike Share': 1, 'Heartland B-cycle': 1, 'Houston B-cycle': 1, 'Blue Bikes': 1, 'Ustica': 1, 'Indego': 1, 'Ind
iana Pacers Bikeshare': 1, 'Juice': 1, 'BICI&CO': 1, 'Link': 1, 'Metro Bike Share': 1, 'Mountain Rides Bike Shar
e': 1, 'Nice Ride': 1, 'Speziainbici': 1, 'SWell Cycle': 1, 'Share-a-Bull Bikes': 1, 'Venezia': 1, 'Spartanburg
```

figure 9

Then, I decided to look at the names of the Citybike API. I realised the names of networks were companies(figure 6) , and most of the companies were only based in one country. So, I looked into the companies of the networks and counted them using the counter module(figure 7), which allowed me to count up arrays and see which company was mentioned the most. Lastly, I put the results from the counter, sorted them through the most common and put it into a graph using Matplotlib (figure 8, 9). The top four companies are Nextbike, Bicincittà, EasyBike and VRN. From this information, I searched where the companies are based, and all of them are found in the same countries they operate from.

3) Looking at indivdual stations within cities

```
url = "https://api.citybik.es/v2/networks/to-bike"  #Turin,Italy

response = requests.get(url)

if response.status_code == 200:
    data = response.json()
    stations = data['network']['stations']
   #bikes = []
    #empty_slots = []

    for station in stations:

        bikes.append(station['free_bikes'])  #append = pushing the intergers of each bi
        empty_slots.append(station['empty_slots'])
    print(f"No. of Stations: {len(stations)}") #len = grabs the length number of statio
    print(f"Mean No. of Bikes: {np.mean(bikes)}") #grabs the mean
    print(f"Mean  No. of Empty Slots: {np.mean(empty_slots)}") #grabs the mean of stati
```

figure 9

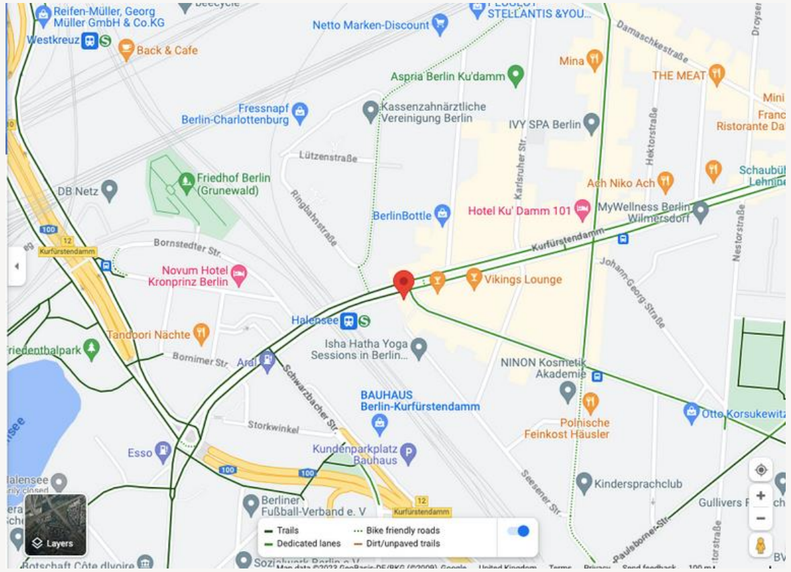
```
In [183]: url = "https://api.citybik.es/v2/networks/nextbike-berlin"

          response = requests.get(url)

          if response.status_code == 200:
              data = response.json()
              stations = data['network']['stations']

              print(stations[1])

{'empty_slots': 0, 'extra': {'bike_uids': ['19236', '17933', '17087', '13120', '13786'], 'number': '1670', 'slots':
4, 'uid': '72228'}, 'free_bikes': 5, 'id': 'c32093893e4e32afee18894c7c0c1ba4', 'latitude': 52.496986, 'longitude': 1
3.29121, 'name': 'virtuell - EDEKA Schmitt (S Halensee)', 'timestamp': '2023-05-19T18:49:30.736000Z'}
```

figure 10

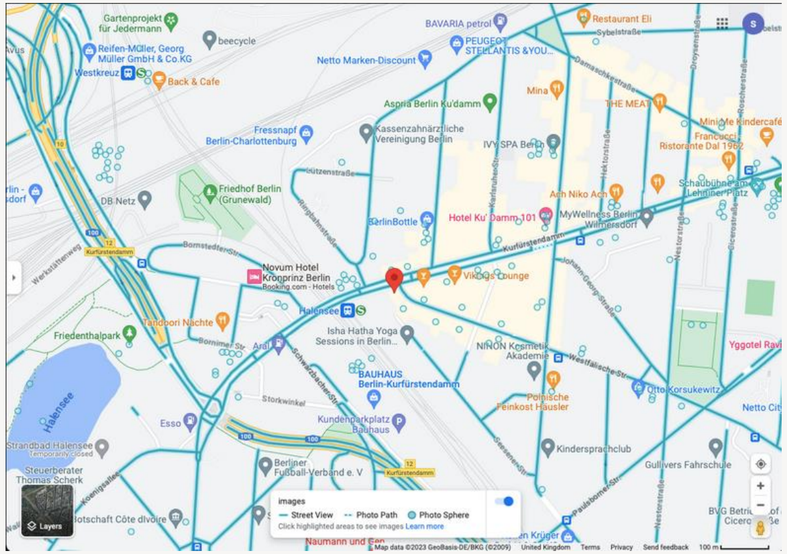
Using this code, I printed the station data off companies, e.g. nextbike-berlin. The output prints out all the data found within the station-enclosed brackets. Then, using the URL and data within the enclosed station data, I looked at the number of stations, the number of stations, the mean of bikes and the mean of empty slots(figure 9). This data can give me a good idea of what stations can be found within Berlin. After I found data within Berlin, I looked into specific stations and their data. With station[1], from the data that I've scraped, you can see the id of the station, the free bikes, the location and the time stamp(figure 10). I decided to look at the mean to get the average f the overall number of bikes and empty slots.

4) Looking at Padestrinisation

Hamberg , Germany 52.497000, 13.291222



Green Lines = Bike Lanes figure 11



Blue Lines = Streets figure 12

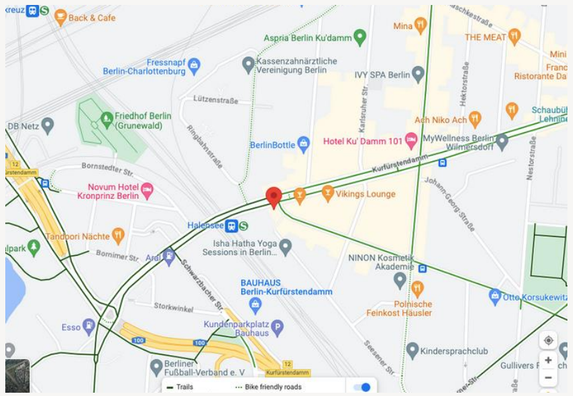
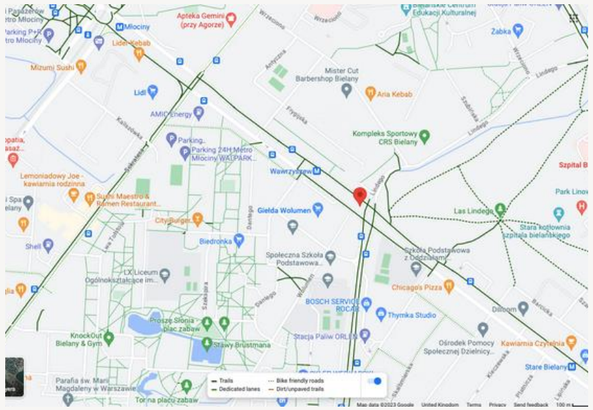
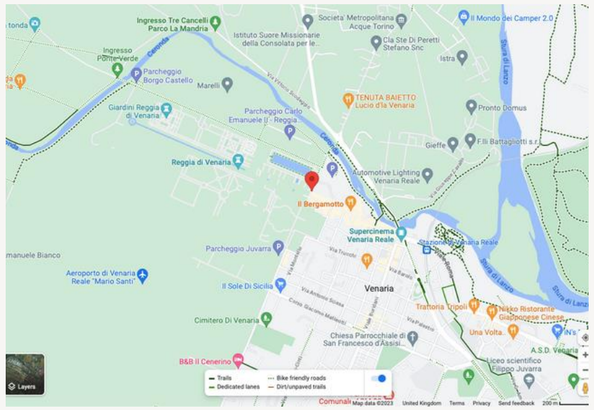
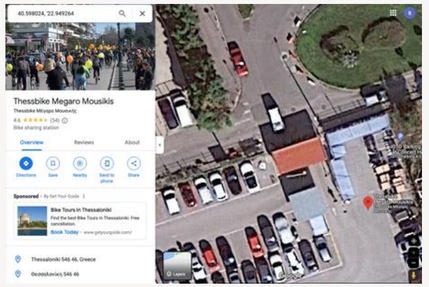
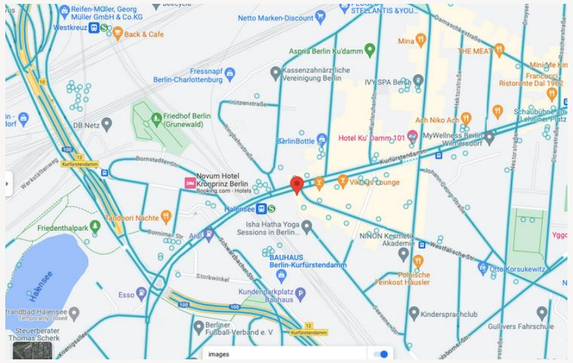
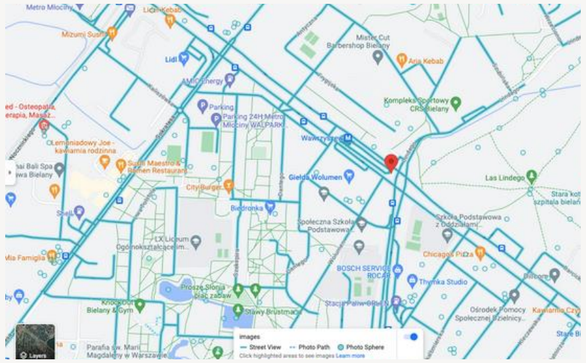
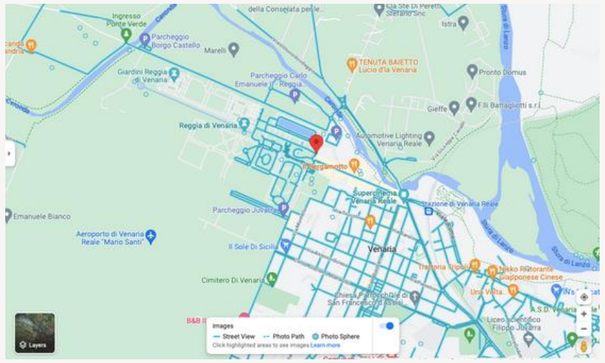
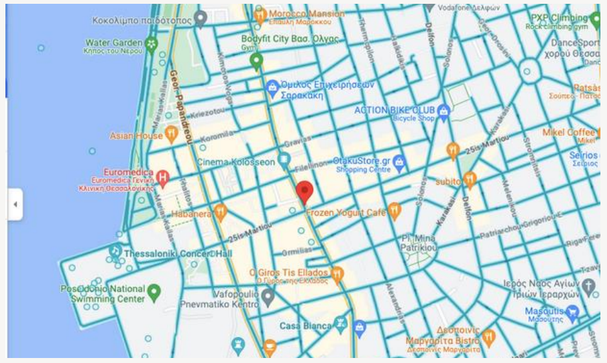
Looking more into specific stations, I looked at google maps to see if my argument on having a city bike share system means a better infrastructure for pedestrianisation. This station is the next-bike station in Berlin. Figure 11 shows the green lines within that area. The area is well-connected to bike lanes. Even in the green area, there are bike lanes implemented. The bike station is conveniently located with other transport links, supermarkets and places to eat.

In Figure 12, the maps display the streets highlighted in blue. The area is connected and has plenty of places to walk around. The minor road is connected. Looking back at the data for this, there were only a few empty slots and five free bikes to take. This can imply that the bike station is full of bikes. However, looking at the timestamp, it was Friday the 19th of May at 6:49, so people had slotted their bikes in for the day and headed home or to nearby restaurants. So I reload the data to give me the latest on that same station.

```
In [4]: url = "https://api.citybik.es/v2/networks/nextbike-berlin"

        response = requests.get(url)

        if response.status_code == 200:
            data = response.json()
            stations = data['network']['stations']

            print(stations[1])
```

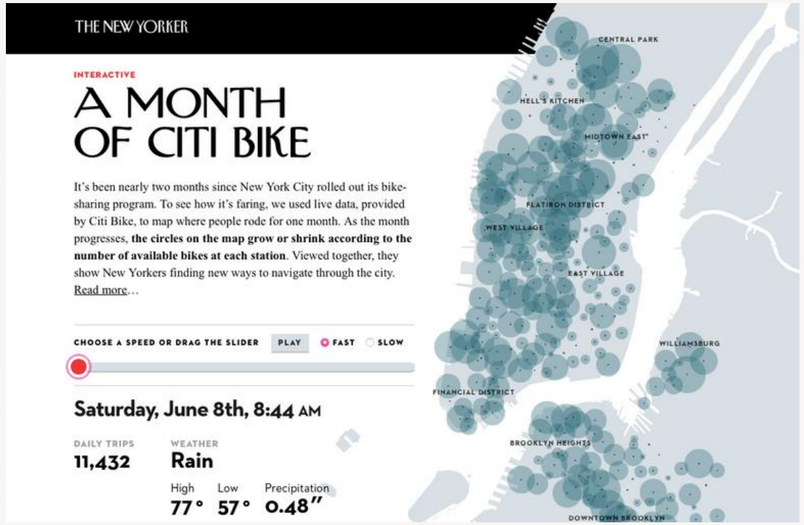
```
{'empty_slots': 1, 'extra': {'bike_uids': ['17845', '13120', '13878'], 'number': '1670', 'slots': 4, 'uid': '72228
'}, 'free_bikes': 3, 'id': 'c32093893e4e32afee18894c7c0c1ba4', 'latitude': 52.496986, 'longitude': 13.29121, 'name
': 'virtuell - EDEKA Schmitt (S Halensee)', 'timestamp': '2023-05-24T15:00:04.403000Z'}
```

The new data shows one empty slot to put back in the bikes and three bikes on the 24th of May at 3 pm. This new data shows that the station gets used by people around the area from looking at empty slots and the number of bikes in total. In my final project, I would like to show the mean of bike usage, the longitude and latitude and the comprehensive data in the city and real-time data I have taken for all four cities and create a visualisation of it on a postcard.

	Germany	Poland	Italy	Greece
Green Lines = Bike Lanes				
Blue Lines = Streets				

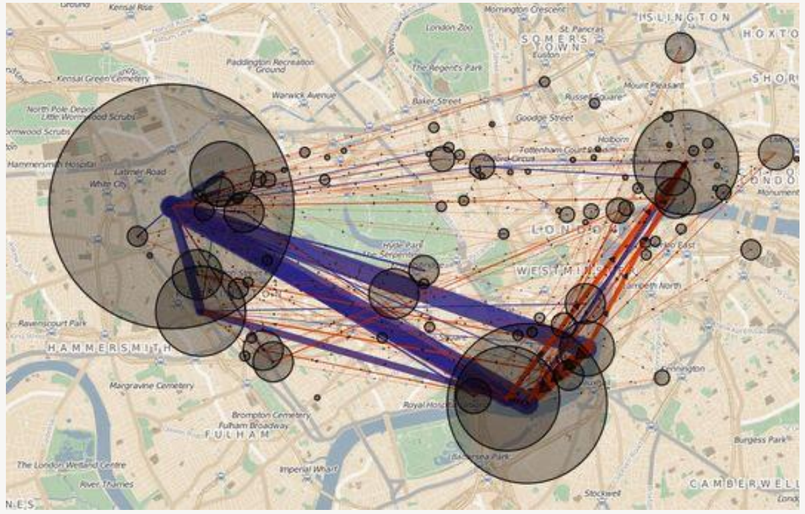
To look at the layout of these cities, I decided to go to Google Maps to look at the street and bike lanes around one of the stations in the city. I found the data of the location through my data and looked at the first available station's coordinates. Looking at all of the stations from a map viewpoint, Germany has the best infrastructure for pedestrianisation due to its amount of bike lanes and street lanes. Around the area, it's found that shops like supermarkets are close to the bike station. Italy had many street lanes but only a few bike lanes within the park/green areas. This station might be used often to ride in the parks because it is close to a prominent landmark, La Venaria Reale. Greece had plenty of street lanes but no bike lanes on Google Maps. This can be mainly because people ride along the coast, which I found through the reviews for the bike station. Poland has many streets and bike lanes, but the bike lanes are mainly on the main road and nearby places like schools. I realised that in the more pedestrianised areas, the bike stations are found within high street areas or highly populated areas where locals live. Unlike less pedestrianised areas, bike stations are in touristy areas, parks for leisure.

5) Similar Visualisations using bike sharing data



A Month of Citi Bike, The New Yorker

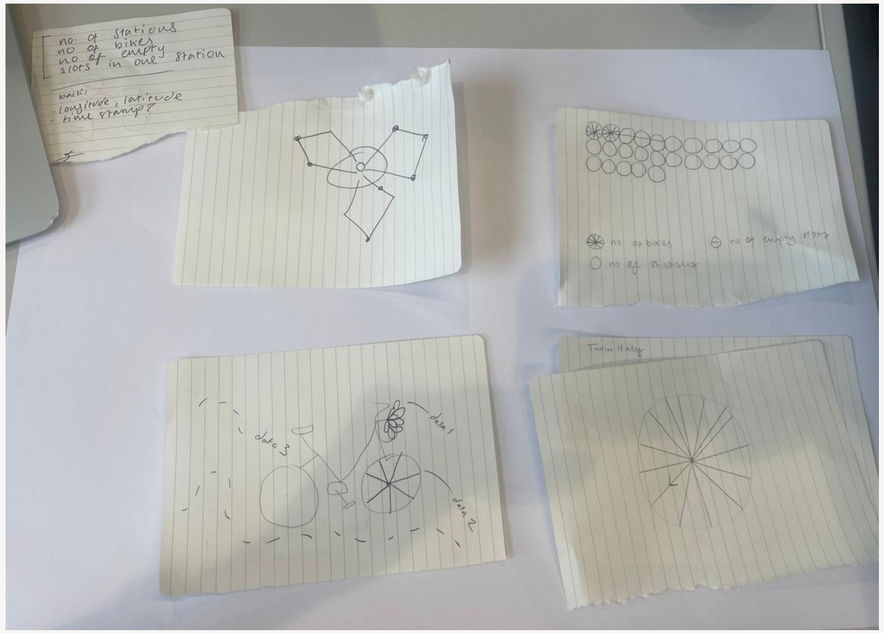
In this project, the new yorker times recorded the available bikes around New York in June. This interactive project also includes the day when the bikes are accessible in the station, temperature and weather. As you drag the slider, you see the circles on the map changing and getting bigger and smaller from how many bikes are available. This project is great because it shows how the data changes over time and can be affected by the weather and time.



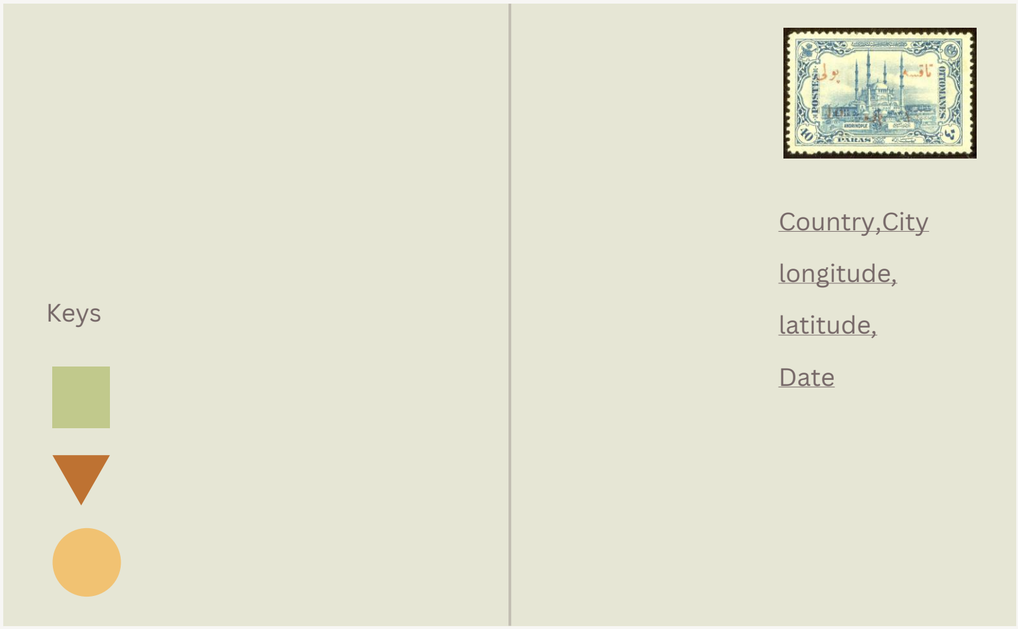
I Know Where You Were Last Summer, James Siddle

In this project, James Siddle looks into the usage of London individual cyclists within six months. The dataset used was the TfL (Transport for London). The dataset included the usage of cyclists using the Bike shares, time and end of a bike ride. Using this data, Siddle created an interactive map with keys determining what each journey meant. For example, the purple lines indicate that the trip was from both directions, and red-lined journies told the route was taken on an early morning. The project is interesting because he uses shapes and colours to understand how Londoners use bike share.

6) Visualising the data



FRONT OF POSTCARD (DRAFT)



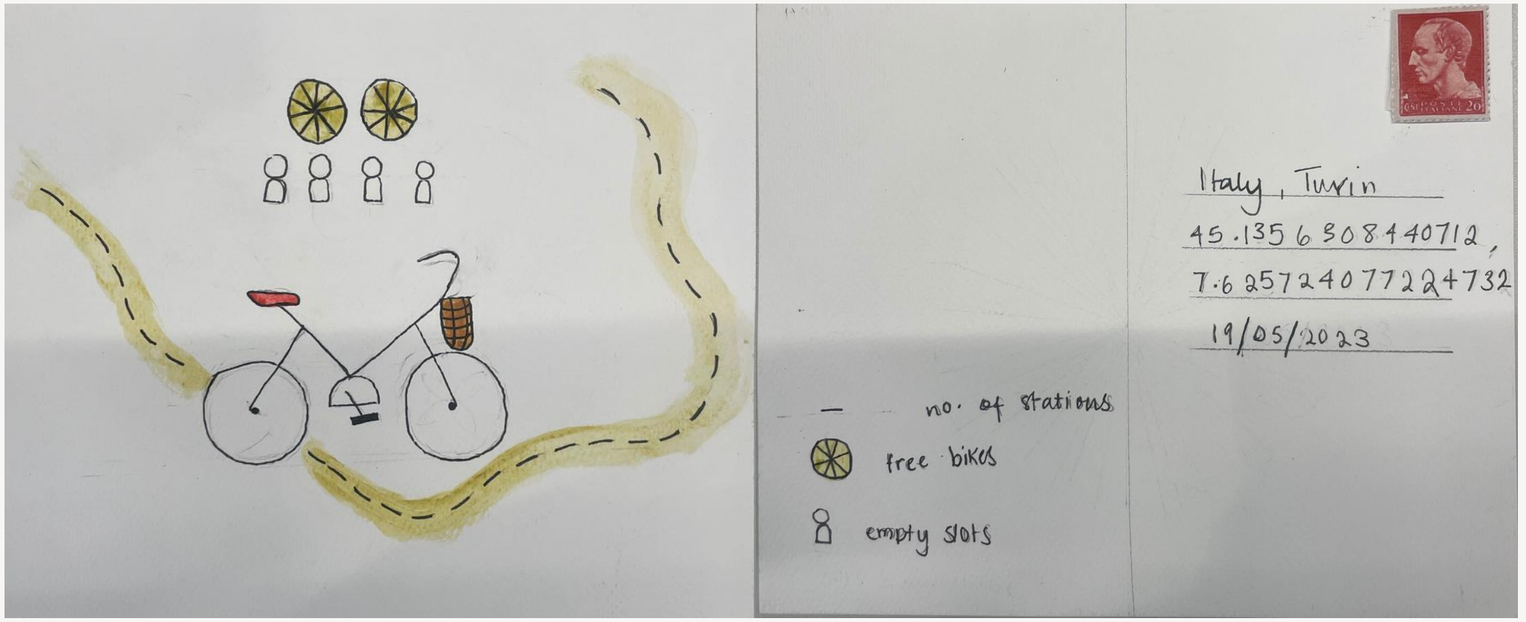
BACK OF POSTCARD LAYOUT

To start visualising, I started drawing how I would visualise the data. I wanted to keep the theme of padestrinisation, bikes and wheels. I also want to incorporate various geometric shapes and different sizes to have distinctive keys. Having distinctive keys can help present mass data. I kept the overall theme of the postcards, but I decided to change how I designed the front.

I chose to place the visualisations at the front and the keys linked to the city bike data at the back. For example, for the bike on the bottom left of the data visualisation draft, the number of dashes in the trail shows the no. of stations. I also included additional data like the country, city, longitude, latitude and date on the right of the postcard where the address that your sending it to would be. And for visual purposes, I included a stamp of the countries to make it seem realistic and that the postcard is moving from one place to another.

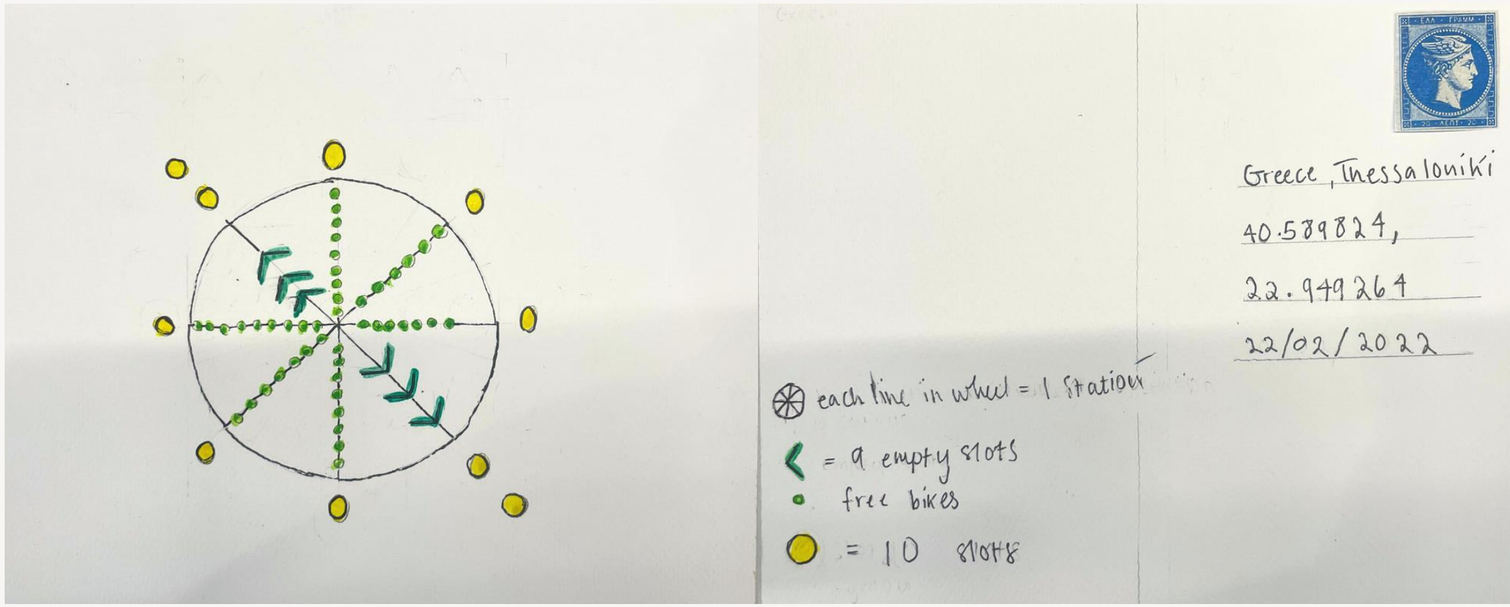
7) Final Project

For Turin, I decided to use trails to show data. The wheels show the number of free bikes in that specific station, and the people icons show the empty slots for the bike stations. For visual purposes, I decided to keep the theme of pedestrianisation by showing the trails of movement, wheels,bikes and people.



For Berlin, I decided to show data representation of the no. of stations as buildings. I chose buildings because they link to infrastructure and planning of the cities. The tandem bike shows the number of free bikes available in that station. I drew a tandem bike to use the bike seats to indicate the free bikes available. The people icons were another way to show the bike slots total of the station.

For Thessaloniki, I decided to draw a big wheel. The keys inside the wheel show the empty slots and free bikes. They remind me of bike wheel light attachments. I decided to group my data in empty slots because it'll be easier for the reader to interpret.



For Łódź, I drew a "map" of an ideal pedestrianised area. The map includes a park, bike lanes and a place like a cafe for people to congregate. The rest of the map is a negative space that is pedestrianised. The free bikes key are bike stations within the area, and the dashes are the number of stations.

CHALLENGES

- One of the challenges was getting information within enclosed brackets. It was hard to access mass data like 'slots'. I resolved it by experimenting with closed brackets in my code and declaring the data within stations. Also, creating lists using append and finding out the no. of stations using len helped a lot.
- I also found that visualising the data to be very hard. I wanted my visualisation postcard to look like a traditional postcard, and the design aspect was challenging since I had to think about what the keys meant. I solved this by planning out my visualisations and having a clear picture of what the data means.
- Another challenge was the data of specific stations being real-time. I resolved this by looking at particular dates and visualising the postcards from there.

TO IMPROVE FURTHER:

- To improve further, I could have looked at data from an even larger scale, focusing on continents or countries to see if the environment does show how much bikes are being used. Using a more extensive data set, it could be more accurate and make my argument stronger.
- Another way to improve data is creating visualisations based on the change of real-time data in these four locations for comparisonbased on time day,week, month to see how actively these stations are utilised.

REFERENCES

DeMaio, P., 2009. Bike-sharing: History, impacts, models of provision, and future. Journal of public transportation, 12(4), pp.41-56.

Yassin, H.H., 2019. Livable city: An approach to pedestrianization through tactical urbanism. Alexandria engineering journal, 58(1), pp.251-259.