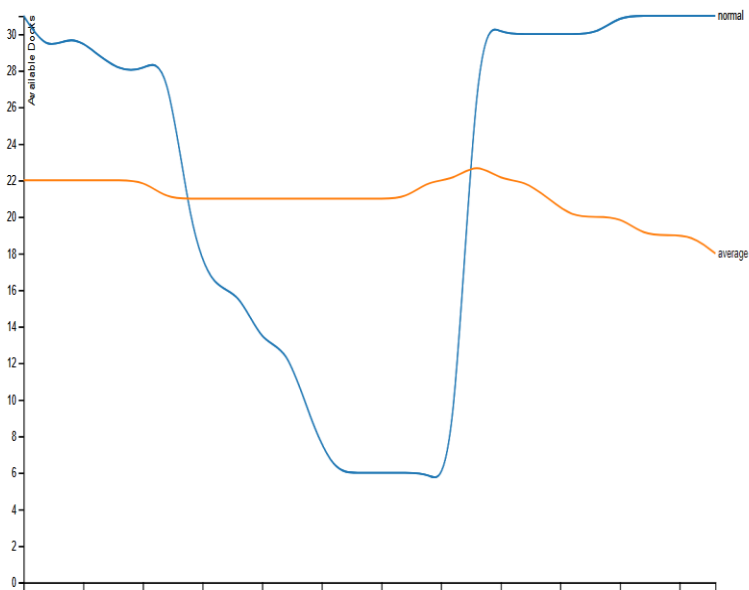


Time Range ☒ Last 1 Hour SMA ☐ Last 24 Hour SMA

Line Chart for last 1 hour(s)



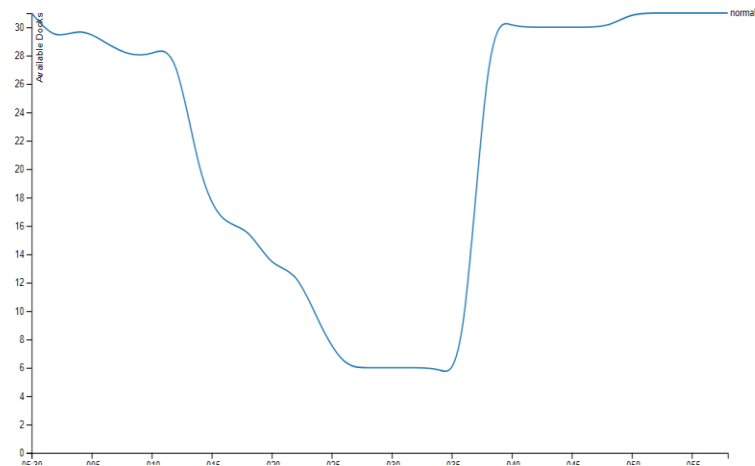


Home

Go Back

Time Range ☐ Last 1 Hour SMA ☐ Last 24 Hour SMA

Line Chart for last 1 hour(s)

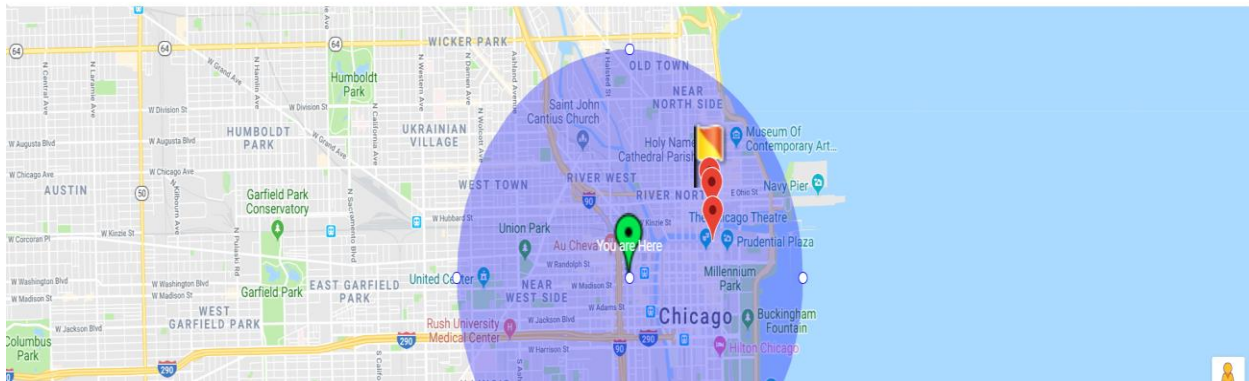


Find Another Place 🔍

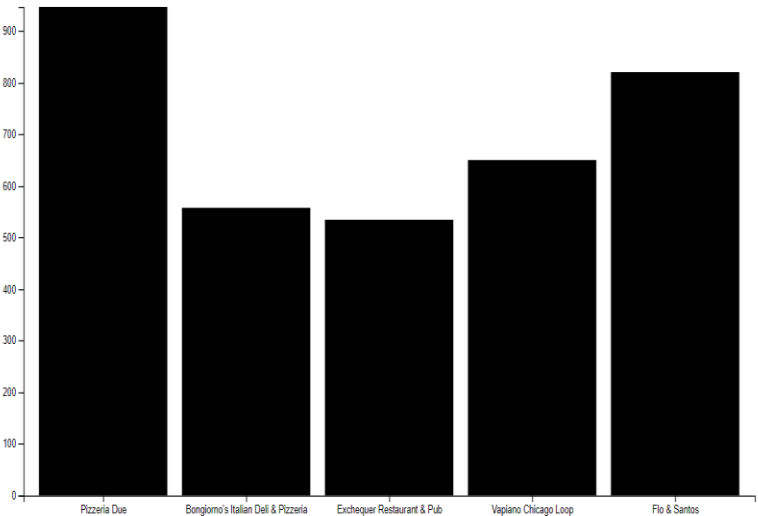
Divvy HeatMap

← Go Back

ID	stationName	availableBikes	availableDocks	is_renting	lastCommunicationTime	latitude	longitude	status	totalDocks	Action
199	Wabash Ave & Grand Ave	20	11	true	2019-05-03, 4:46:20 pm	41.891466	-87.626761	IN_SERVICE	31	Real Time Chart 📊
125	Rush St & Hubbard St	12	11	true	2019-05-03, 4:46:10 pm	41.890173	-87.626185	IN_SERVICE	23	Real Time Chart 📊
194	Wabash Ave & Wacker Pl	10	9	true	2019-05-03, 4:48:00 pm	41.886875	-87.62603	IN_SERVICE	19	Real Time Chart 📊



Bar Chart



localhost:4200/list_of_places

Chicago Social Hub

Find Another Place

Bar Chart

Name	Phone	Address	Closed	Rating	Review Count	
Pizzeria Due	(312) 943-2400	619 N Wabash Ave	false	3.5	947	<div>Divvy Near by</div>
Bongiorno's Italian Deli & Pizzeria	(312) 755-1255	405 N Wabash Ave	false	4	558	<div>Divvy Near by</div>
Exchequer Restaurant & Pub	(312) 939-5633	226 S Wabash Ave	false	3.5	535	<div>Divvy Near by</div>
Vapiano Chicago Loop	(312) 384-1960	44 S Wabash Ave	false	3.5	650	<div>Divvy Near by</div>
Flo & Santos	(312) 566-9817	1310 S Wabash Ave	false	4	821	<div>Divvy Near by</div>

localhost:4200/find

Chicago Social Hub

Find

pizza

Restaurants, Bars, Cafe, ...

Where

wabash

Near me, ...

Go

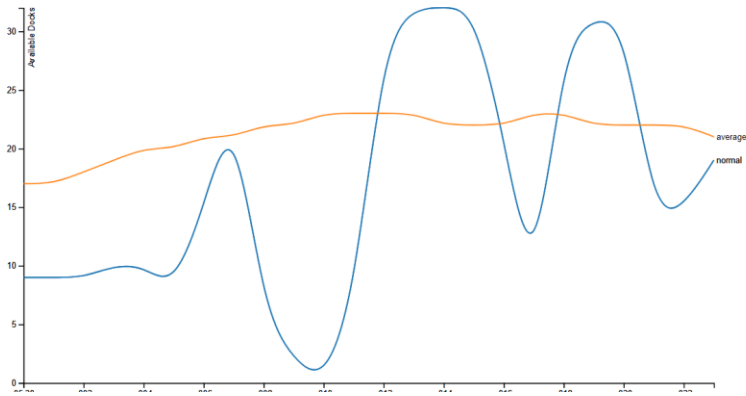
Time Range

Last 24 Hours

☐ Last 1 Hour SMA

☒ Last 24 Hour SMA

Line Chart for last 24 hour(s)



```
# Extract the body of the reply
response_body = response.read()

# Decode the format in json format
stations_json = json.loads(response_body.decode("utf-8"))

stations_json = stations_json['stationBeanList']

for item in stations_json:
    if item['city'] == "Chicago ":
        item['city'] = "Chicago"

list_divvy_documents = []

for stations in stations_json:
    index_stations = {
        "_index": "divvy_stations_logs1",
        "_type": "logs",
        "_source": stations
    }
    list_divvy_documents.append(index_stations)

output_file = open('divvyToES.json', 'w', encoding='utf-8')

for dic in list_divvy_documents:
    json.dump(dic, output_file)
    output_file.write("\n")

es1 = Elasticsearch()

helpers.bulk(es1, list_divvy_documents)

# Sleep for 125 seconds; divvy updates its stations status every 2 minutes
print('Sent Heartbeat to Divvy Servers and Going to sleep for 125 seconds now ...')
time.sleep(125)
continue
```