In [4]:
```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
from plotly import tools
# import plotly.plotly as py
import chart_studio.plotly
from plotly.offline import init_notebook_mode, iplot
init_notebook_mode(connected=True)
import plotly.graph_objs as go
import plotly.figure_factory as ff
from IPython.display import HTML, Image
import plotly.express as px
px.set_mapbox_access_token(open(".mapbox_token").read())
import datetime
import math
import matplotlib.pyplot as plt
```

In [5]:
```
import pandas as pd
birds = pd.read_csv("bird_tracking.csv")
```

In [6]:
```
birds.head()
```

Out[6]:

	altitude	date_time	device_info_serial	direction	latitude	longitude	speed_2d	bird_name
0	71	2013-08-15 00:18:08+00	851	-150.469753	49.419859	2.120733	0.150000	Eric
1	68	2013-08-15 00:48:07+00	851	-136.151141	49.419880	2.120746	2.438360	Eric
2	68	2013-08-15 01:17:58+00	851	160.797477	49.420310	2.120885	0.596657	Eric
3	73	2013-08-15 01:47:51+00	851	32.769360	49.420359	2.120859	0.310161	Eric
4	69	2013-08-15 02:17:42+00	851	45.191230	49.420331	2.120887	0.193132	Eric

In [7]:
```
birds['year'] = pd.to_datetime(birds['date_time']).dt.year
birds['month']= pd.to_datetime(birds['date_time']).dt.month
birds['hour'] = pd.to_datetime(birds['date_time']).dt.hour
birds['log_altitude'] = np.log2(birds['altitude'].clip(lower=1))
```

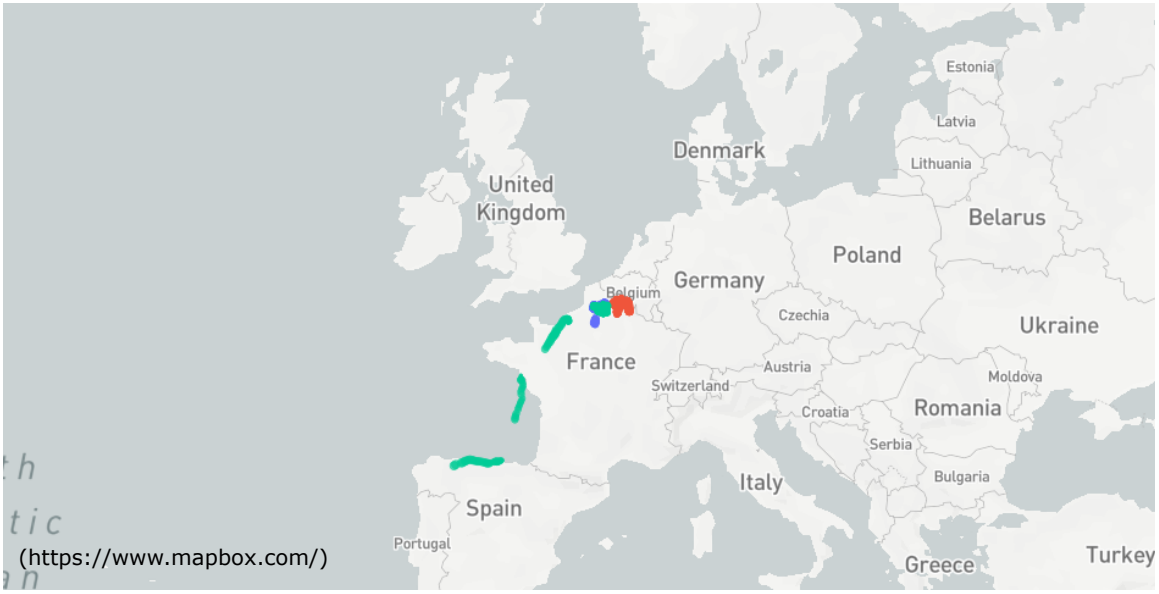
In [8]:
```
birds['day or night'] = birds.apply(lambda d: 'night' if pd.to_datetime(d['date_time']).hour >= 18 or pd.to_
datetime(d['date_time']).hour <=  5 else 'day', axis = 1)
birds_night = birds[birds['day or night']== 'night']
birds_day = birds[birds['day or night']== 'day']
```

In [9]:
```
august = birds[birds['month']==8]
august.set_index('date_time').groupby('bird_name')['log_altitude'].mean().to_frame().reset_index()
```

Out[9]:

	bird_name	log_altitude
0	Eric	6.562098
1	Nico	6.706361
2	Sanne	6.075402

In [10]:
```
day_aug = birds_day[birds_day['month'] == 8]
px.set_mapbox_access_token(open(".mapbox_token").read())
fig = px.scatter_mapbox(day_aug, lat="latitude", lon="longitude", color = "bird_name",
                  color_continuous_scale=px.colors.cyclical.IceFire,size = 'log_altitude',size_max= 5, zoom=
3)
fig.show()
```



Overview¶ The data focuses on three birds named Nico Sanne and Eric, it tracks their migration from late 2013 to early 2014 Headers explain the data we are looking at in particular Questions we are asking are posed before the code and output After the output there are explainations/analysis of results These explainations determine differences/characteristics of the three birds in terms of the presented data and form some conclusions The following is based on data from August¶

In [12]:
```
august = birds[birds['month']==8]
august.set_index('date_time').groupby('bird_name')['log_altitude'].mean().to_frame().reset_index()
```

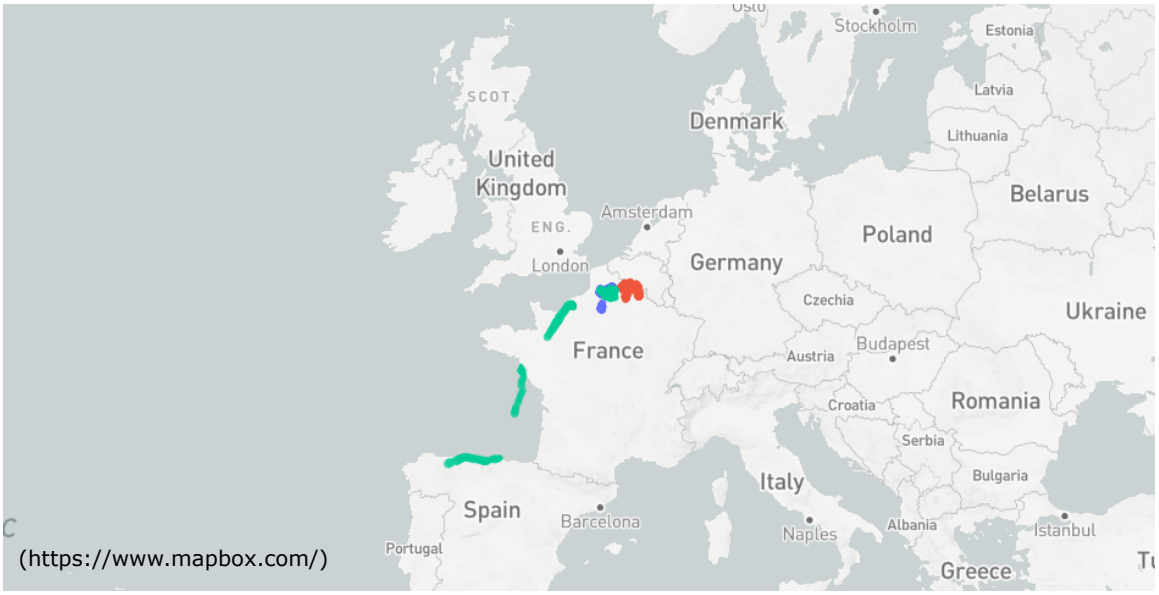
Out[12]:

	bird_name	log_altitude
0	Eric	6.562098
1	Nico	6.706361
2	Sanne	6.075402

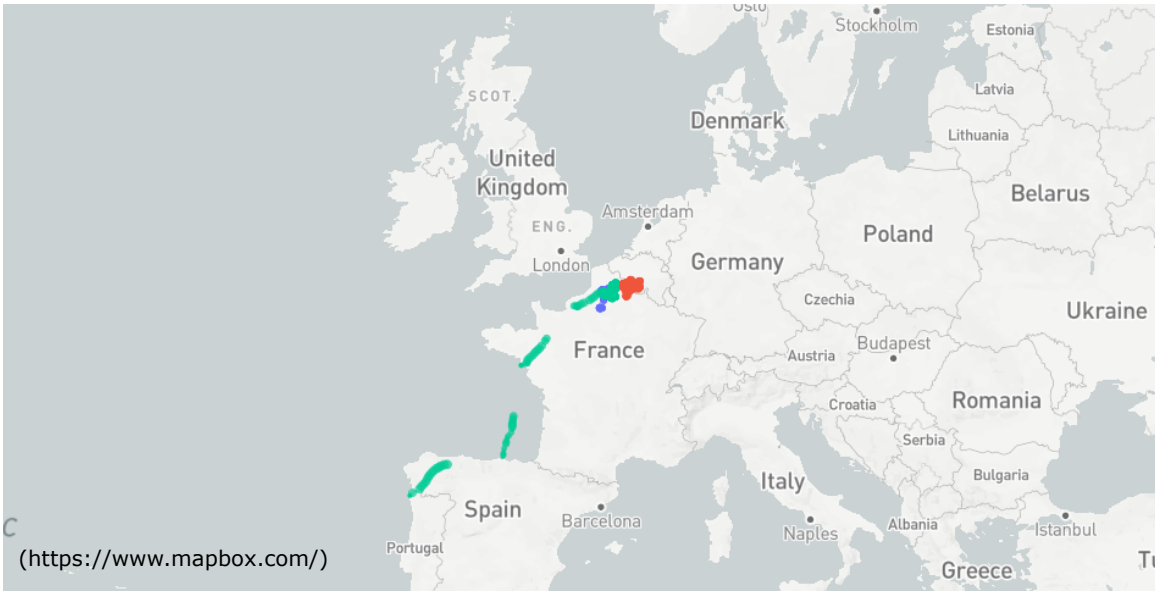
It appears that during the month of August Nico flies the highest on average. This could mean that the terrain he is flying over is higher above sea level then that of Eric and Sanne. This is especially true when compared to Sanne who has the lowest altitude on average in August. This would mean that in all likelihood Sanne flies closer to sea level on average in August as her average altitude is significantly lower.

Is there a difference in flight patterns and activity between the three birds in both the day and night time? NOTE: The gaps in the map are from it swtiched from day to night and vice versa if we looked at August as a whole there would be no gaps in flight paths

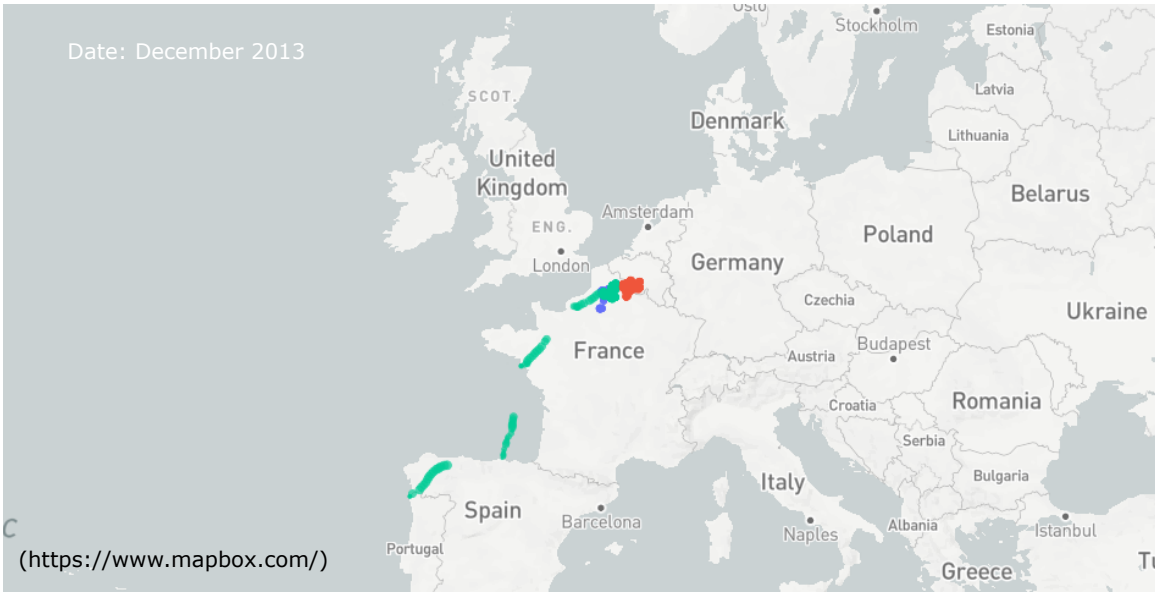
In [13]:
```
day_aug = birds_day[birds_day['month'] == 8]
px.set_mapbox_access_token(open(".mapbox_token").read())
fig = px.scatter_mapbox(day_aug, lat="latitude", lon="longitude", color = "bird_name",
                  color_continuous_scale=px.colors.cyclical.IceFire,size = 'log_altitude',size_max= 5, zoom=
3)
fig.show()
```



In [14]:
```
night_aug = birds_night[birds_night['month'] == 8]
px.set_mapbox_access_token(open(".mapbox_token").read())
fig = px.scatter_mapbox(night_aug, lat="latitude", lon="longitude", color = "bird_name",
                  color_continuous_scale=px.colors.cyclical.IceFire,size = 'log_altitude',size_max= 5, zoom=
3)
fig.show()
```



In [15]:
```
fig.add_annotation(
    x=0.05,
    y=0.95,
    text="Date: December 2013",
    showarrow=False,
    font=dict(size=14, color="white")
)
```



In [ ]:
```
bird_data = birds
bird_data["log_altitude"] = np.log2(bird_data["altitude"].clip(lower=1))
bird_fall_winter = bird_data[(bird_data['date_time'] >= '2013-12-01') & (bird_data['date_time'] <= '2013-12-
31')]

px.set_mapbox_access_token(open(".mapbox_token").read())
fig = px.scatter_mapbox(bird_fall_winter, lat="latitude", lon="longitude", color="bird_name", size="log_alti
tude",
                        color_continuous_scale=px.colors.cyclical.IceFire, size_max=5, zoom=5)
fig.show()
```

In [ ]:
```
fig = px.scatter_mapbox(
    bird_fall_winter,
    lat="latitude",
    lon="longitude",
    color="bird_name",
    size="log_altitude",
    color_continuous_scale=px.colors.cyclical.IceFire,
    size_max=5,
    zoom=5,
    custom_data=["bird_name", "log_altitude", "date_time"]
)
```

In [19]:
```
fig.update_traces(
    hovertemplate = "<br>".join([
        "Latitude: %{lat}",
        "Longitude: %{lon}",
        "Bird: %{customdata[0]}",
        "Log altitude: %{customdata[1]:.2f}",
        "Date: %{customdata[2]}"
    ])
)
```