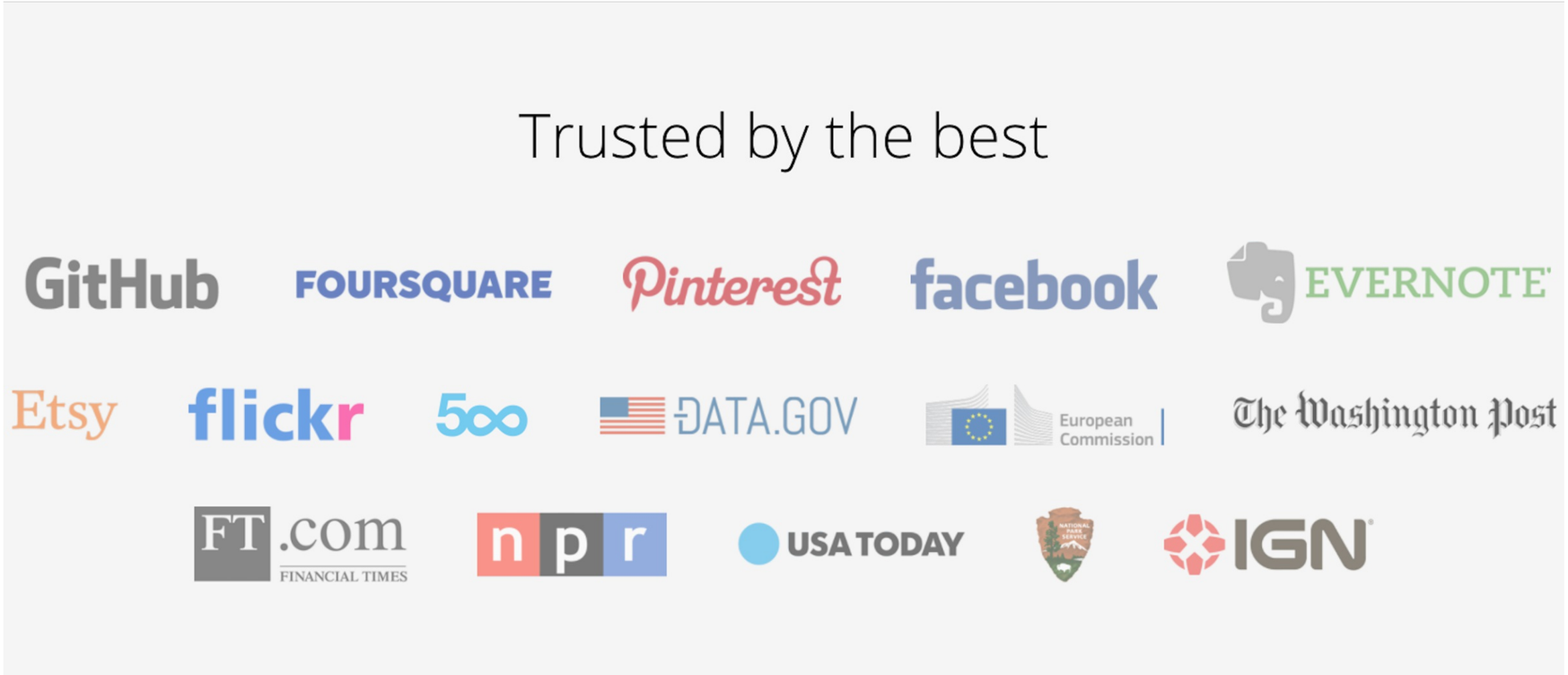
INTERACTIVE MAPS WITH LEAFLET IN R

Introduction to leaflet

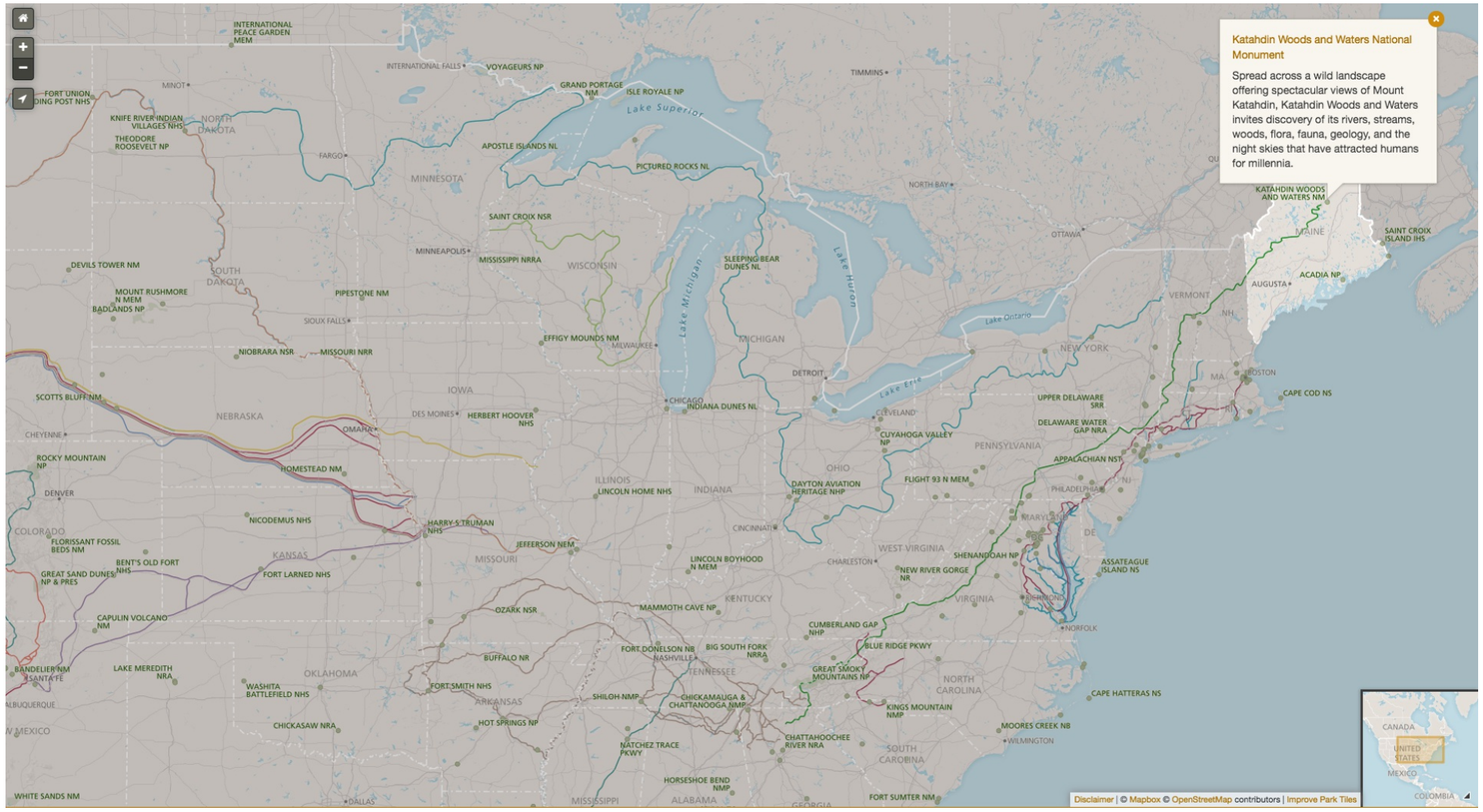
Rich Majerus
Assistant Vice President, Colby College

leaflet

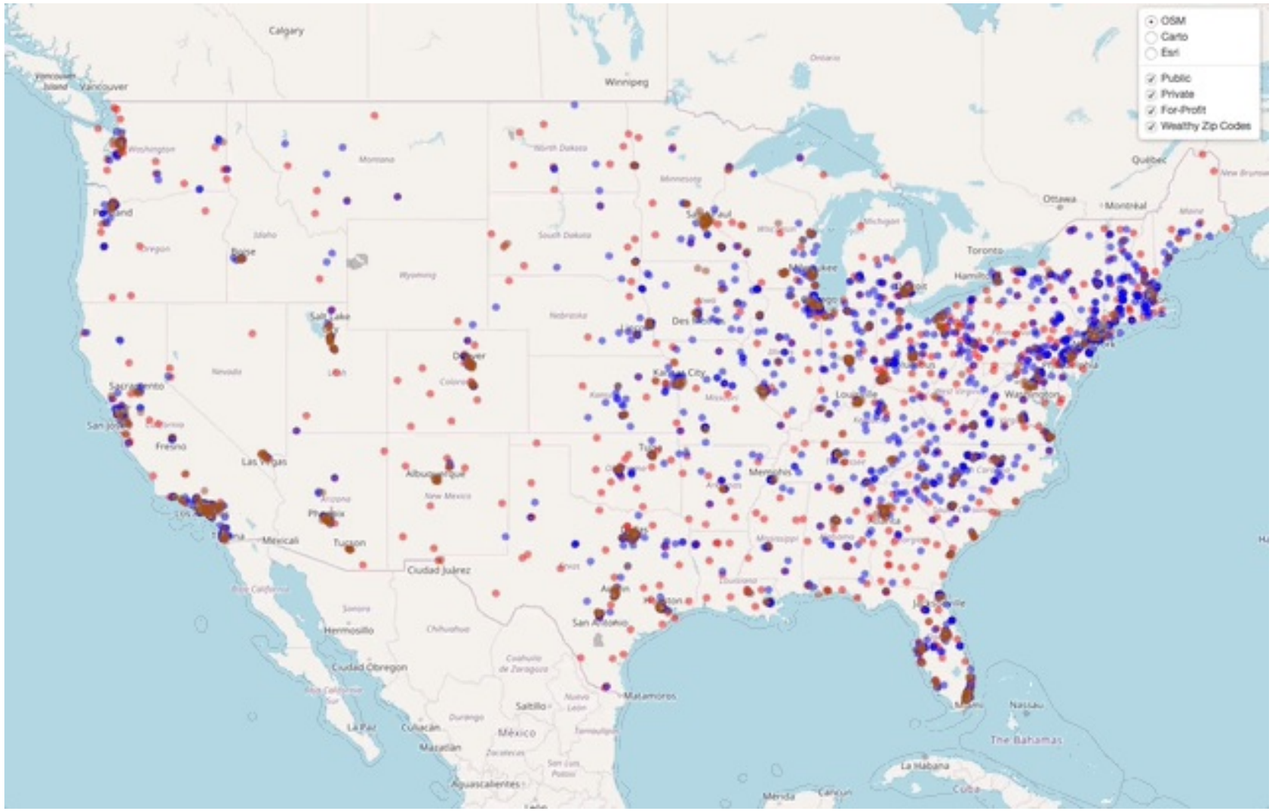
- Open-source JavaScript library
- Popular option for creating interactive mobile-friendly maps
- Can be created using only R code via the htmlwidgets package



leaflet Example: National Parks Service

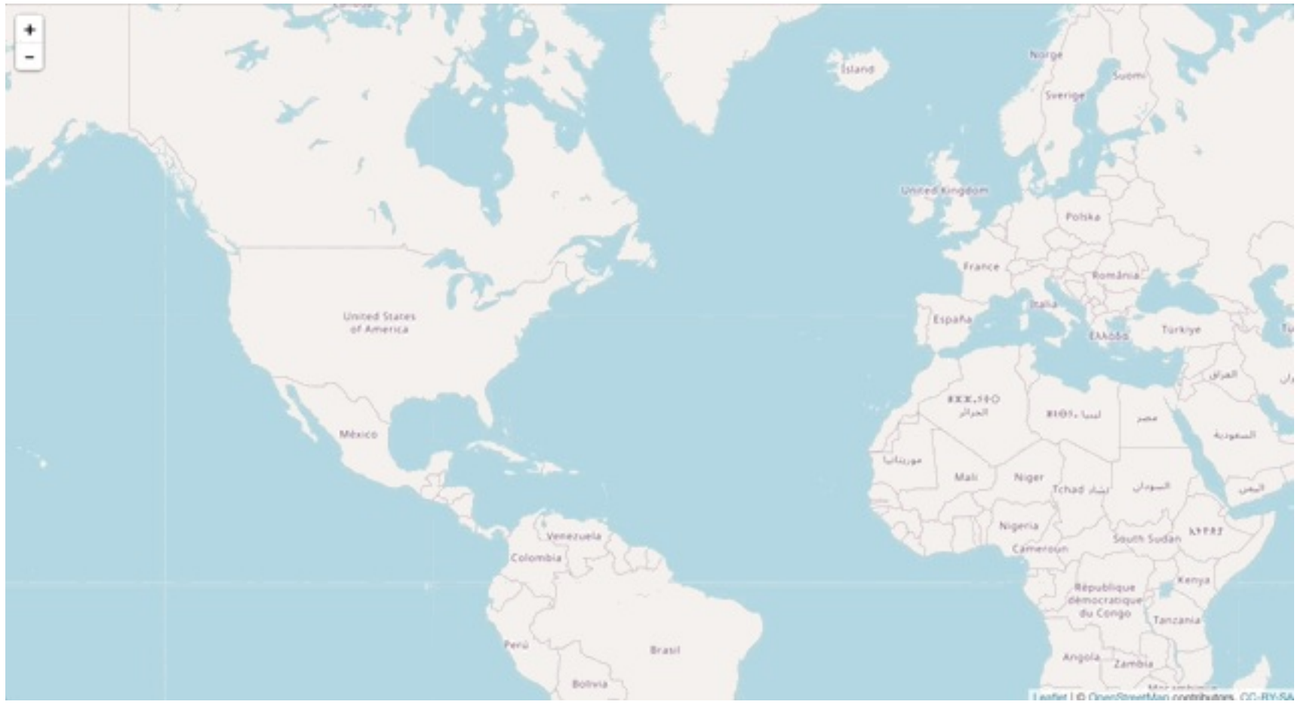


What We are Working Toward



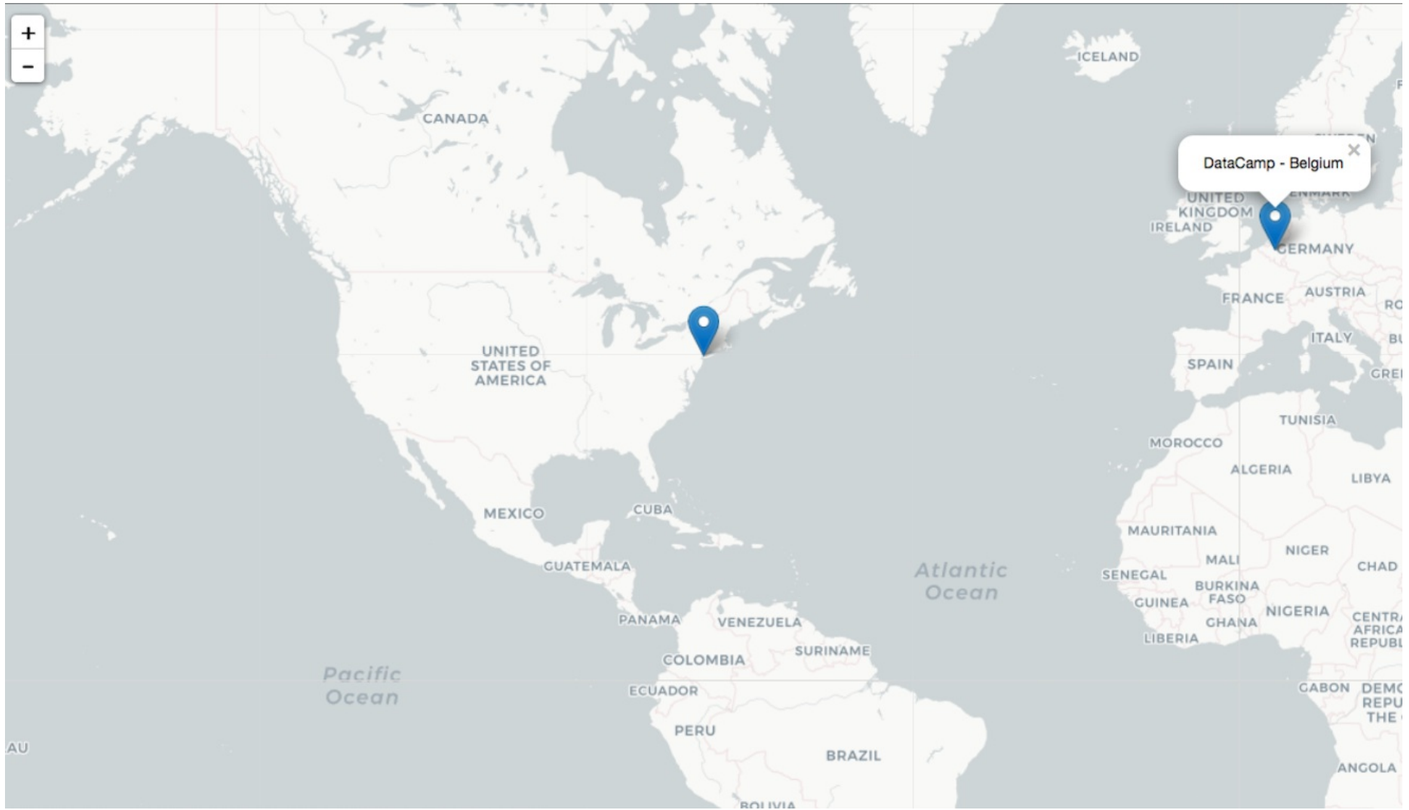
Creating our First leaflet Map

```
library(leaflet)

leaflet() %>%
   addTiles()
```



Where We are Going in Chapter 1

```
leaflet() %>%
   addProviderTiles("CartoDB") %>%
   addMarkers(lng = dc_hq$lon,
              lat = dc_hq$lat,
              popup = dc_hq$hq)
```



INTERACTIVE MAPS WITH LEAFLET IN R

Let's practice!

INTERACTIVE MAPS WITH LEAFLET IN R

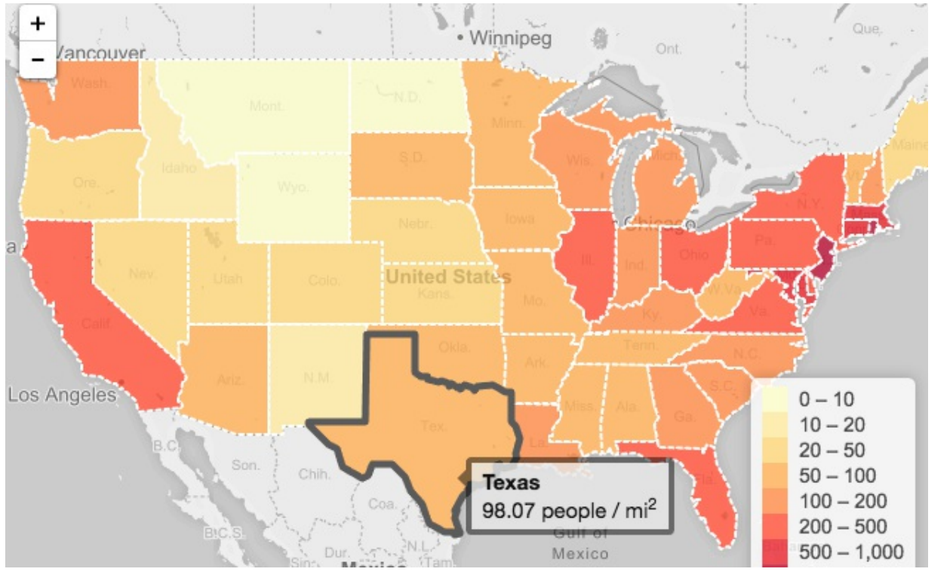
Provider Tiles

Rich Majerus
Assistant Vice President, Colby College

Selecting a Base Map

- Why are you making this map?
  - Exploratory analysis
  - Deliverable product
  - Just for fun!
- What type of data are you plotting?
  - Points
  - Paths
  - Polygons

Selecting a Base Map



leaflet Provider List

- The leaflet packages comes with 100+ provider tiles
- The names of these tiles are stored in a list named providers

```
names(providers)[1:5]
```

```
  [1] "OpenStreetMap"
  [2] "OpenStreetMap.Mapnik"
  [3] "OpenStreetMap.BlackAndWhite"
  [4] "OpenStreetMap.DE"
  [5] "OpenStreetMap.France"
```

Exploring leaflet Provider Tiles

```
names(providers)[str_detect(names(providers), "OpenStreetMap")]

[1] "OpenStreetMap"                "OpenStreetMap.Mapnik"
[3] "OpenStreetMap.BlackAndWhite" "OpenStreetMap.DE"
[5] "OpenStreetMap.France"         "OpenStreetMap.HOT"
```

addProviderTiles()

- Replace addTiles() with addProviderTiles() to change your basemap
- Pass name of provider tile to addProviderTiles()

```
leaflet() %>%
     # addTiles()
     addProviderTiles("OpenStreetMap.BlackAndWhite")
```

INTERACTIVE MAPS WITH LEAFLET IN R

Let's practice!

INTERACTIVE MAPS WITH LEAFLET IN R

Setting the Default Map View

Rich Majerus
Assistant Vice President, Colby College

Geocoding in R

- A common approach is to use the geocode() function in the ggmap package
- Returns the latitude and longitude of an address or a place name

```
library(ggmap)

geocode("350 5th Ave, New York, NY 10118")

Information from URL : http://maps.googleapis.com/maps/api/geocode/...

lon       lat
-73.98575 40.74856
```

Geocoding in R II

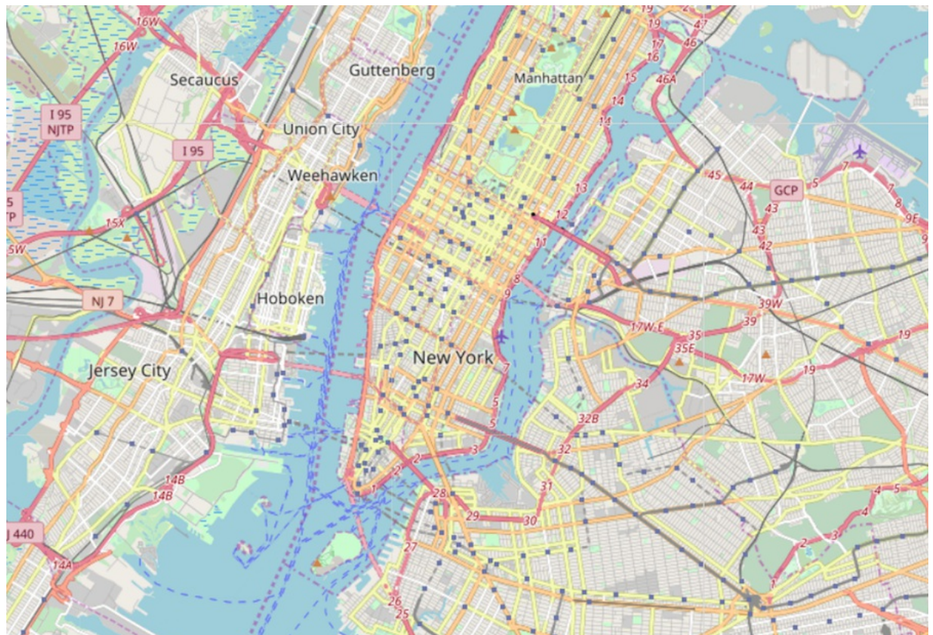
```
    geocode(location,
            output = c("latlon", "latlona", "more", "all"),
            source = c("google", "dsk"))
```

```
geocode("Colby College", output = "more", source = "google")

lon       lat      type          loctype
-69.66264 44.56387 establishment rooftop

address
4000 mayflower hill dr, waterville, me 04901, usa
```

Setting the Default Map View

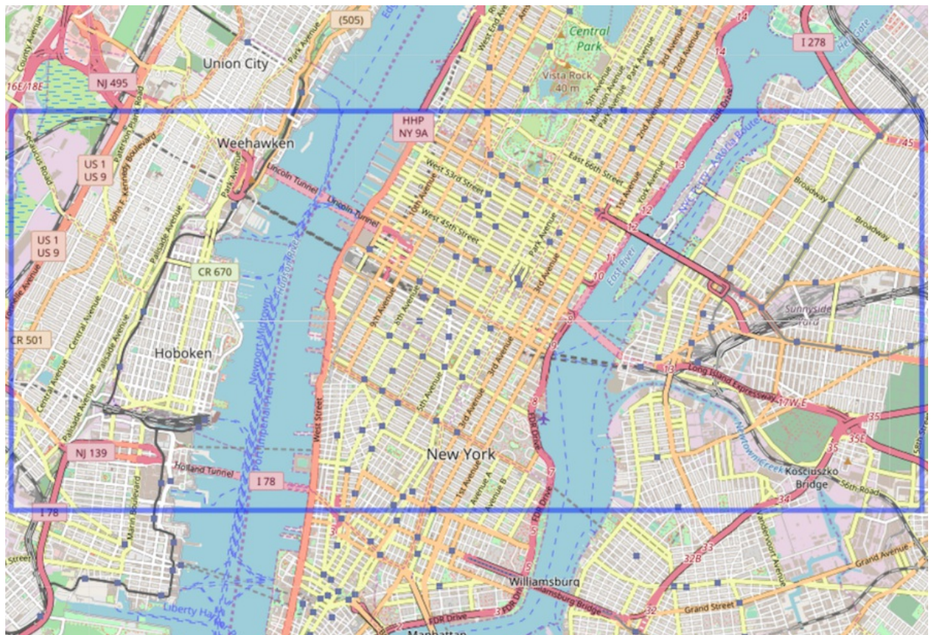
setView()

```
leaflet() %>%
 addTiles() %>%
 setView(lng = -73.98575,
         lat = 40.74856,
         zoom = 13)
```



fitBounds()

```
leaflet() %>%
 addTiles() %>%
 fitBounds(
  lng1 = -73.910, lat1 = 40.773,
  lng2 = -74.060, lat2 = 40.723)
```



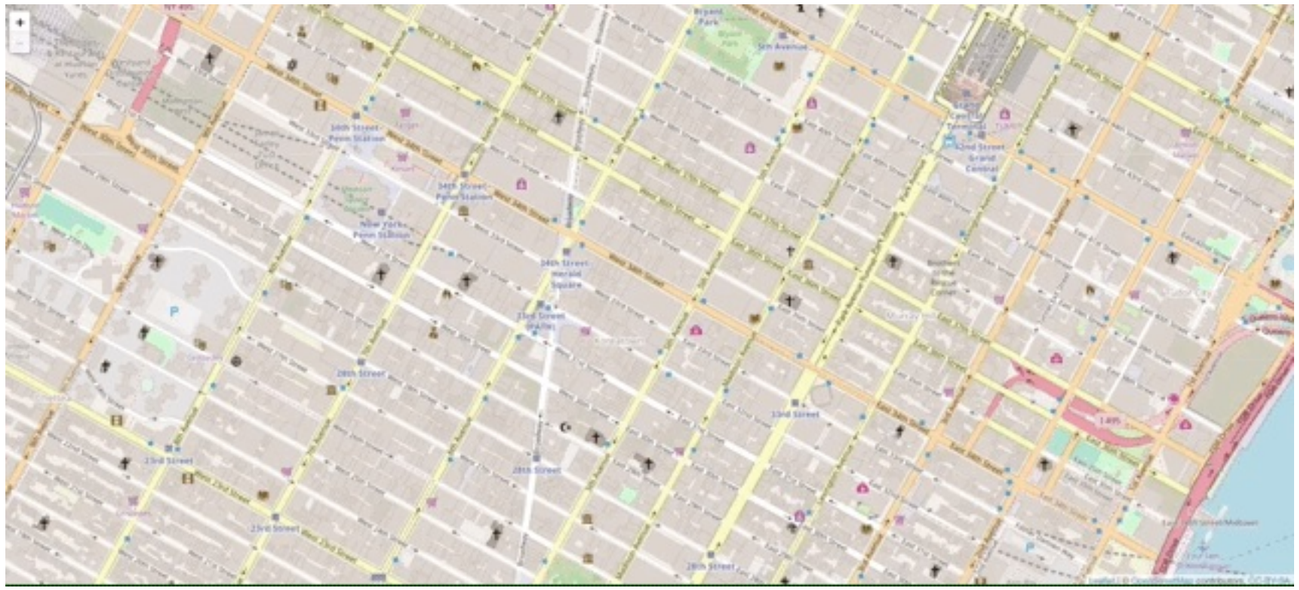
Staying Focused

```
leaflet(options =
            leafletOptions(dragging = FALSE,
                           minZoom = 14,
                           maxZoom = 18))  %>%
    addProviderTiles("CartoDB")  %>%
    setView(lng = -73.98575, lat = 40.74856, zoom = 18)
```

- Leaflet references
  - http://leafletjs.com/reference-1.3.0.html
  - https://rstudio.github.io/leaflet/

Restoring Focus

```
leaflet()  %>%
     addTiles()  %>%
     setView(lng = -73.98575, lat = 40.74856, zoom = 18) %>%
     setMaxBounds(lng1 = -73.98575,
                  lat1 = 40.74856,
                  lng2 = -73.98575,
                  lat2 = 40.74856)
```



INTERACTIVE MAPS WITH LEAFLET IN R

Let's practice!

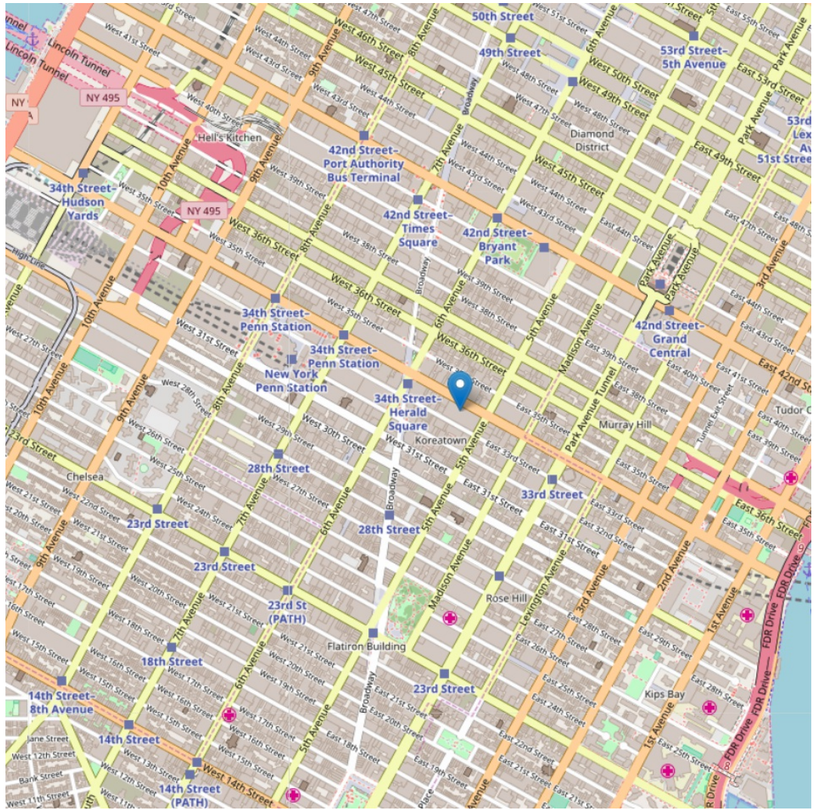
INTERACTIVE MAPS WITH LEAFLET IN R

Plotting DataCamp HQ

Rich Majerus
Assistant Vice President, Colby College

Plotting a Point

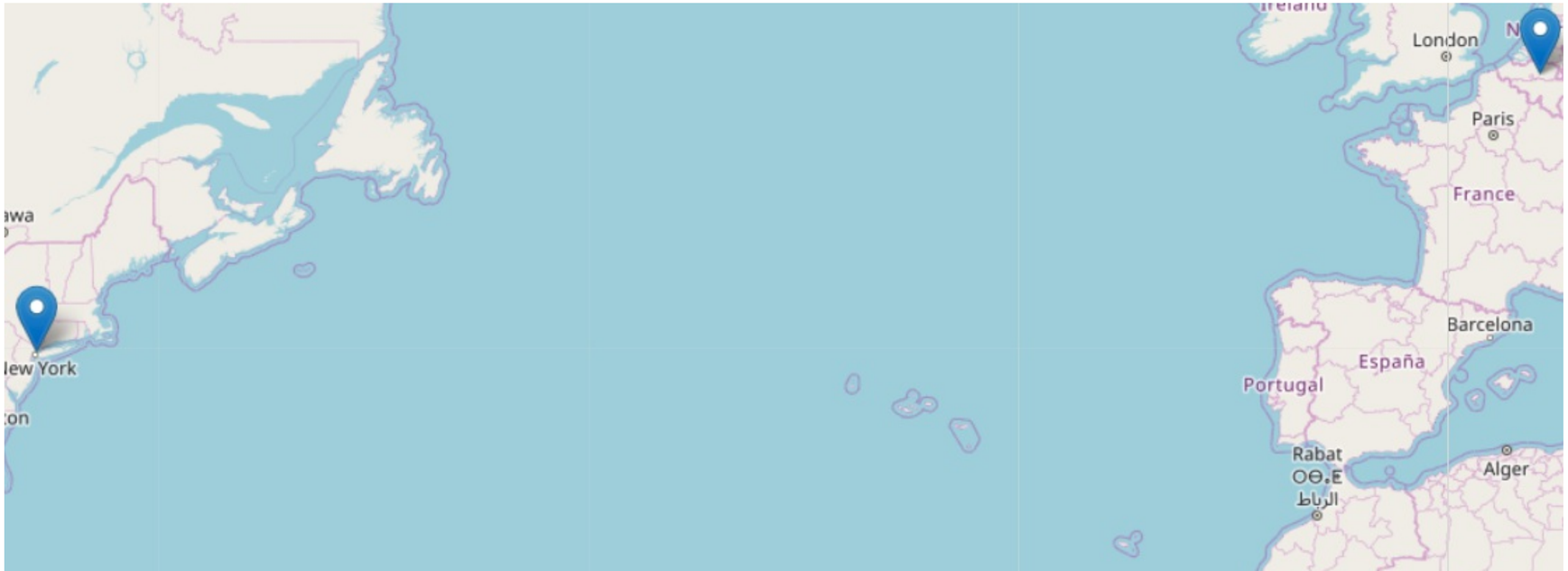
```
# add marker layer to map
leaflet() %>%
     addTiles() %>%
     addMarkers(lng = -73.98575,
                lat = 40.74856)
```



- Supplying Marker Data
  - Numeric data frame columns
  - Numeric vectors
- addMarkers() Defaults
  - Centered on a single point
  - Zoomed to fit all points

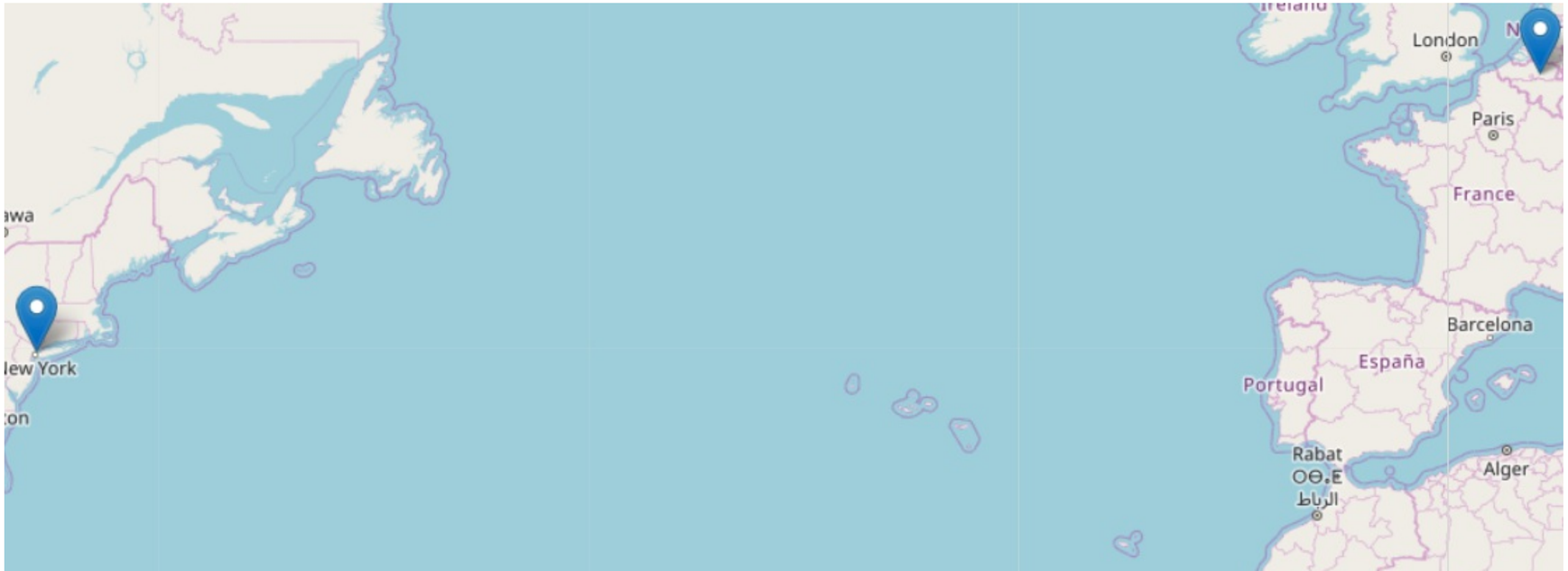
Plotting Multiple Points

```
dc_hq <-
     tibble(
         hq  = c("DataCamp - NYC", "DataCamp - Belgium"),
         lon = c(-73.98575, 4.717863),
         lat = c(40.74856, 50.881363))

leaflet() %>%
     addTiles() %>%
     addMarkers(lng = dc_hq$lon, lat = dc_hq$lat)
```



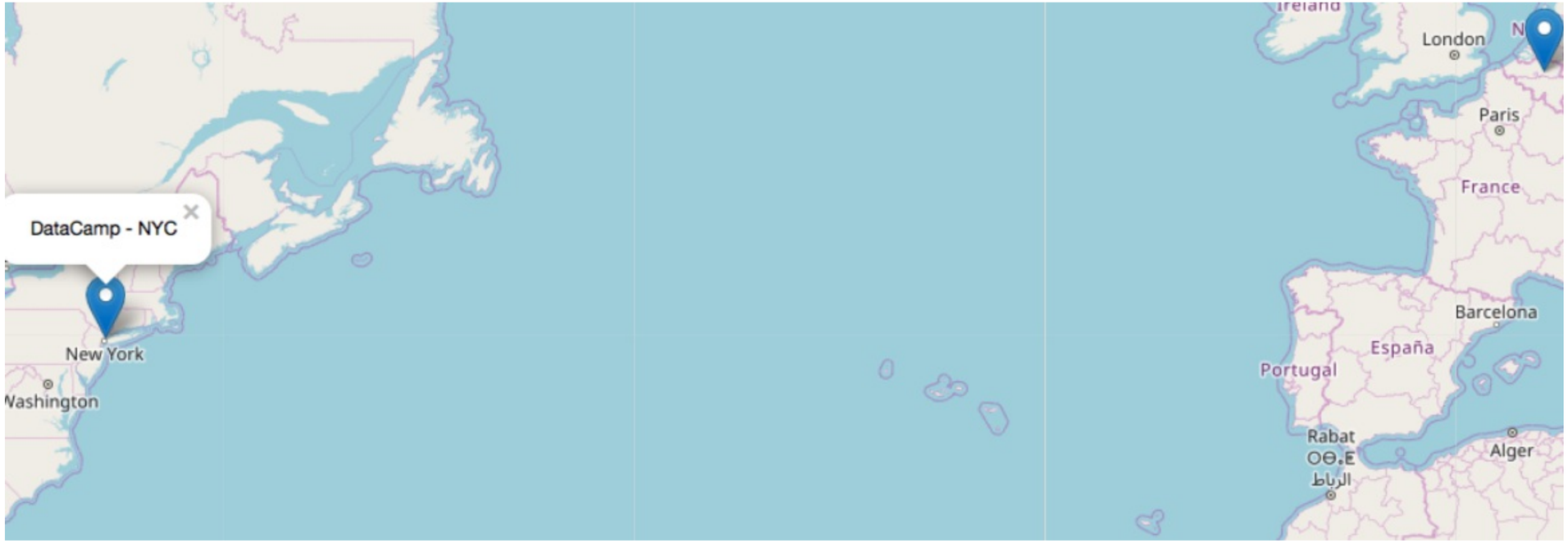
Plotting Multiple Points II

```
# When piping a data frame into the leaflet function
# R will search for columns named lat/latitude and lon/lng/long/longitude

dc_hq  %>%
     leaflet() %>%
     addTiles() %>%
     addMarkers()

Assuming 'lon' and 'lat' are longitude and latitude, respectively
```



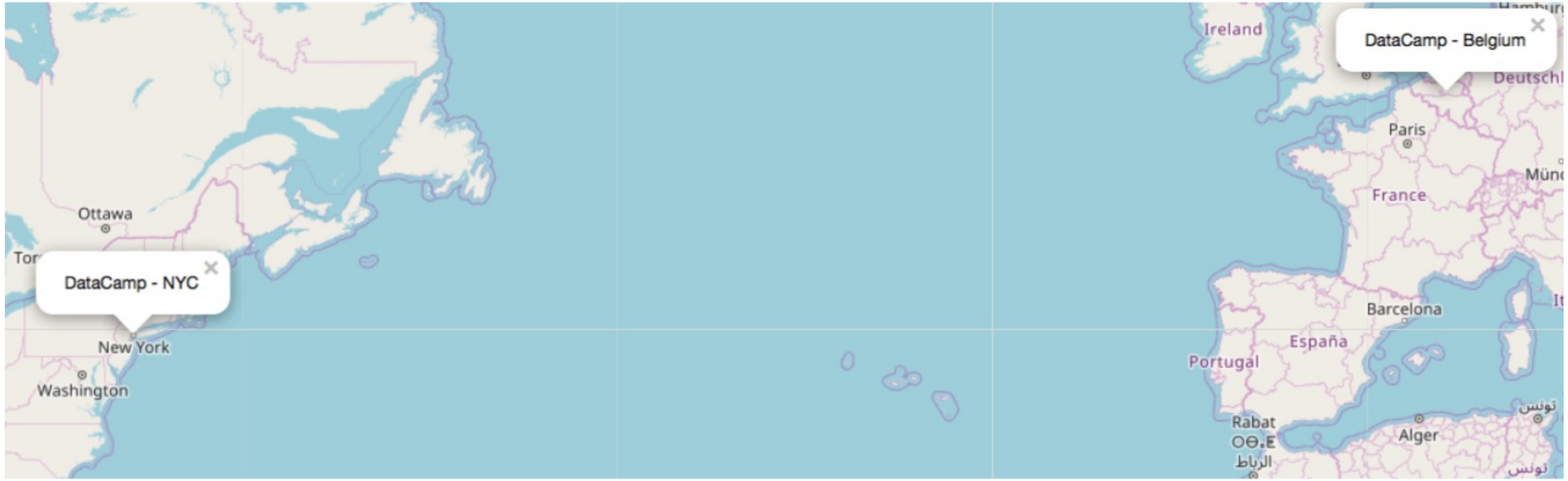
Pop-ups

```
leaflet() %>%
     addTiles() %>%
     addMarkers(lng = dc_hq$lon, lat = dc_hq$lat, popup = dc_hq$hq)
```



Pop-ups II

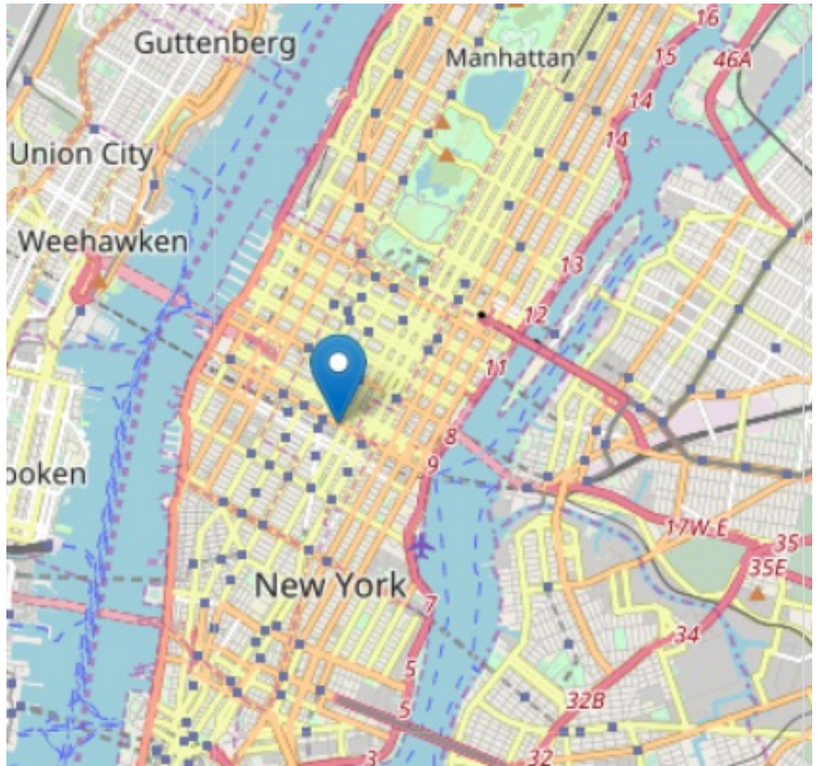
```
leaflet() %>%
     addTiles() %>%
     addPopups(lng = dc_hq$lon, lat = dc_hq$lat, popup = dc_hq$hq)
```



Storing leaflet Maps as Objects

```
m <-
    leaflet() %>%
    addTiles()  %>%
    setView(lng = dc_hq$lon[1],
            lat = dc_hq$lat[1],
            zoom = 12)
```


```
# %>% leaflet objects to functions
# to add or edit layers
m  %>%
    addMarkers(lng = dc_hq$lon,
               lat = dc_hq$lat,
               popup = dc_hq$hq)
```



INTERACTIVE MAPS WITH LEAFLET IN R

Let's practice!