# Make your own GIS service

Summer School on Digital Humanities

Web site: https://bit.ly/dt4h-gis

Augusto Ciuffoletti

13 giugno 2025



- The tutorial consists in the step-by-step creation of a simple app that:
  - Displays a mad
  - Allow the user to add markets to the map
  - Export
- I he tool
   Stockblit
- Stackblit
- Stackblitz
- The link to ea
  - course websit

- The tutorial consists in the step-by-step creation of a simple app that:
  - Displays a map
  - Allow the user to add markers to the map
  - Exposis the markers as a GeoJSON string
- The tool
- The code
   Stackblitz
- The link to ea
  - course website

- The tutorial consists in the step-by-step creation of a simple app that:
  - Displays a map
  - Allow the user to add markers to the map
  - Exports the markers as a GeoJSON string
- Stackblitz https
- The code for e
  Stackblitz broke
- The link to each
  - course website

- The tutorial consists in the step-by-step creation of a simple app that:
  - Displays a map
  - Allow the user to add markers to the map
  - Exports the markers as a GeoJSON string
- The tool we are going to use to practice the leaflet library is Stackblitz (https://stackblitz.com/), an online IDE for JavaScr
- The code of each set of the second of the se
- The link to each project is in mediale of each slide and in the course website trage described of this spoic.

- The tutorial consists in the step-by-step creation of a simple app that:
  - Displays a map
  - Allow the user to add markers to the map
  - Exports the markers as a GeoJSON string
- The tool we are going to use to practice the *leaflet* library is Stackblitz (https://stackblitz.com/), an online IDE for JavaScript
- The code for each slep carries we wed lested, and modified as a Stackblitz project. Lafeth
- The link to each project is in the course website bage ledicated of this tooic.



- The tutorial consists in the step-by-step creation of a simple app that:
  - Displays a map
  - Allow the user to add markers to the map
  - Exports the markers as a GeoJSON string
- The tool we are going to use to practice the *leaflet* library is Stackblitz (https://stackblitz.com/), an online IDE for JavaScript
- The code for each step can be viewed, tested, and modified as a Stackblitz project
- The link to each project is in the title of each slide, and in the course website bage dedicated to this tooic.



- The tutorial consists in the step-by-step creation of a simple app that:
  - Displays a map
  - Allow the user to add markers to the map
  - Exports the markers as a GeoJSON string
- The tool we are going to use to practice the *leaflet* library is Stackblitz (https://stackblitz.com/), an online IDE for JavaScript
- The code for each step can be viewed, tested, and modified as a Stackblitz project
- The link to each project is in the title of each slide, and in the course website page dedicated to this topic.



• Follow the project link for the first step

 In the right frame you see the preview of your service, showing a map

e the URI

a In the left &

• In the left

The select

- Follow the project link for the first step
- In the right frame you see the preview of your service, showing a map
  - the URL on top of the frame is functional: try it.
  - In the left party tipere is the project conten

(ロト 4週 > 4 분 > 4 분 > - 분 - 쒸익()

- Follow the project link for the first step
- In the right frame you see the preview of your service, showing a map
  - the URL on top of the frame is functional: try it...
  - In the left trange there is the project content

- Follow the project link for the first step
- In the right frame you see the preview of your service, showing a map
  - the URL on top of the frame is functional: try it...
- In the left frame there is the project content

- Follow the project link for the first step
- In the right frame you see the preview of your service, showing a map
  - the URL on top of the frame is functional: try it...
- In the left frame there is the project content
  - The README.md describes the step

- Follow the project link for the first step
- In the right frame you see the preview of your service, showing a map
  - the URL on top of the frame is functional: try it...
- In the left frame there is the project content
  - The README.md describes the step
  - The index.html is the HTML code for the page
  - The Wher files en

D + 4 A + 4 E + E + 900

- Follow the project link for the first step
- In the right frame you see the preview of your service, showing a map
  - the URL on top of the frame is functional: try it...
- In the left frame there is the project content
  - The README.md describes the step
  - The index.html is the HTML code for the page
  - The index.js file is the javascript code using the leaflet library
  - The other mestan

- Follow the project link for the first step
- In the right frame you see the preview of your service, showing a map
  - the URL on top of the frame is functional: try it...
- In the left frame there is the project content
  - The README.md describes the step
  - The index.html is the HTML code for the page
  - The index.js file is the javascript code using the leaflet library
  - The other files are not of interest

- Follow the project link for the first step
- In the right frame you see the preview of your service, showing a map
  - the URL on top of the frame is functional: try it...
- In the left frame there is the project content
  - The README.md describes the step
  - The index.html is the HTML code for the page
  - The index.js file is the javascript code using the leaflet library
  - The other files are not of interest
- The selected file is shown in the center frame
  - You can edit the code and see what happens
  - For instance, try to shange the string in line 10 in index.html and notice the preview changes
  - Your edits remain socal. To cave your project you should register on Stackhlitz



- Follow the project link for the first step
- In the right frame you see the preview of your service, showing a map
  - the URL on top of the frame is functional: try it...
- In the left frame there is the project content
  - The README.md describes the step
  - The index.html is the HTML code for the page
  - The index.js file is the javascript code using the leaflet library
  - The other files are not of interest
- The selected file is shown in the center frame
  - You can edit the code and see what happens
  - For instance, try to change the string in line 10 in index.html and notice the preview change
  - Your edits remain scal. To cave your project you should register on Stackblitz



- Follow the project link for the first step
- In the right frame you see the preview of your service, showing a map
  - the URL on top of the frame is functional: try it...
- In the left frame there is the project content
  - The README.md describes the step
  - The index.html is the HTML code for the page
  - The index.js file is the javascript code using the leaflet library
  - The other files are not of interest
- The selected file is shown in the center frame
  - You can edit the code and see what happens
  - For instance, try to change the string in line 10 in index.html and notice the preview change
  - Your edits remain ocal. To save your project you should register on Stackhlitz



- Follow the project link for the first step
- In the right frame you see the preview of your service, showing a map
  - the URL on top of the frame is functional: try it...
- In the left frame there is the project content
  - The README.md describes the step
  - The index.html is the HTML code for the page
  - The index.js file is the javascript code using the leaflet library
  - The other files are not of interest
- The selected file is shown in the center frame
  - You can edit the code and see what happens
  - For instance, try to change the string in line 10 in index.html and notice the preview change
  - Your edits remain local. To save your project you should register on Stackblitz



 The first step in our tutorial consists of using the Leaflet library to display an OpenStreetMap raster



- The first step in our tutorial consists of using the Leaflet library to display an OpenStreetMap raster
- How to:
  - The reference to the library is in the package vock lie.
     In the FTY/L file:
  - The fixe that the fine of the fixe of the fixe of the fixed of the fix
  - The A Hall TVOUE STAFFICATOR SO WE'VE TAFFER TO THE STAFFICATOR SO WE'VE TO THE STAFFICATOR SO THE STAFFICAT

- The first step in our tutorial consists of using the Leaflet library to display an OpenStreetMap raster
- How to:
  - The reference to the library is in the package-lock file

- The first step in our tutorial consists of using the Leaflet library to display an OpenStreetMap raster
- How to:
  - The reference to the library is in the package-lock file
  - In the HTML file:

- The first step in our tutorial consists of using the Leaflet library to display an OpenStreetMap raster
- How to:
  - The reference to the library is in the package-lock file
  - In the HTML file:
    - a head element with the CSS for the Leaflet library

- The first step in our tutorial consists of using the Leaflet library to display an OpenStreetMap raster
- How to:
  - The reference to the library is in the package-lock file
  - In the HTML file:
    - a head element with the CSS for the Leaflet library
    - a div element for the map (its id is mapid)

OP E 4E 4E 4 A A

- The first step in our tutorial consists of using the Leaflet library to display an OpenStreetMap raster
- How to:
  - The reference to the library is in the package-lock file
  - In the HTML file:
    - a head element with the CSS for the Leaflet library
    - a div element for the map (its id is mapid)
  - The index.js file contains the JavaScript code of our App

- The first step in our tutorial consists of using the Leaflet library to display an OpenStreetMap raster
- How to:
  - The reference to the library is in the package-lock file
  - In the HTML file:
    - a head element with the CSS for the Leaflet library
    - a div element for the map (its id is mapid)
  - The index.js file contains the JavaScript code of our App
  - The capital L stands for the Leaflet class
  - So we create a man with two parameters

- The first step in our tutorial consists of using the Leaflet library to display an OpenStreetMap raster
- How to:
  - The reference to the library is in the package-lock file
  - In the HTML file:
    - a head element with the CSS for the Leaflet library
    - a div element for the map (its id is mapid)
  - The index.js file contains the JavaScript code of our App
  - The capital L stands for the Leaflet class
  - So we create a map with two parameters
    - the id of the DOM element hosting the raster (our mapid)
    - A JavaScript officet that describes position of map center and zoom level
  - Next we add a background raster, which is OpenStreetMap



- The first step in our tutorial consists of using the Leaflet library to display an OpenStreetMap raster
- How to:
  - The reference to the library is in the package-lock file
  - In the HTML file:
    - a head element with the CSS for the Leaflet library
    - a div element for the map (its id is mapid)
  - The index.js file contains the JavaScript code of our App
  - The capital L stands for the Leaflet class
  - So we create a map with two parameters
    - the id of the DOM element hosting the raster (our mapid)
    - A Java cript object that describes position of map center and zoom level
  - Next we add a background raster, which is OpenStreetMap



- The first step in our tutorial consists of using the Leaflet library to display an OpenStreetMap raster
- How to:
  - The reference to the library is in the package-lock file
  - In the HTML file:
    - a head element with the CSS for the Leaflet library
    - a div element for the map (its id is mapid)
  - The index.js file contains the JavaScript code of our App
  - The capital L stands for the Leaflet class
  - So we create a map with two parameters
    - the id of the DOM element hosting the raster (our mapid)
    - A JavaScript object that describes position of map center and zoom level
  - Next we add a background raster, which is OpenStreetMap



- The first step in our tutorial consists of using the Leaflet library to display an OpenStreetMap raster
- How to:
  - The reference to the library is in the package-lock file
  - In the HTML file:
    - a head element with the CSS for the Leaflet library
    - a div element for the map (its id is mapid)
  - The index.js file contains the JavaScript code of our App
  - The capital L stands for the Leaflet class
  - So we create a map with two parameters
    - the id of the DOM element hosting the raster (our mapid)
    - A JavaScript object that describes position of map center and zoom level
  - Next we add a background raster, which is OpenStreetMap



#### Step 1: Lab activity

- Browse the web to find the coordinates of a place at your choice as the center of the raster
- Modify/remove the zoom factor

#### IMPORTANT:

- relax: you cannot damage my repo (you'd need my credentials)
- you may Fork (button on top-left corner) a branch in a repo of your own (recommended not strictly needed)
- you can undo unsaved updates with Ctrl-z
- after forking and signing up you can save your work



#### Step 2: show the coordinates (project)

 When the user clicks on the map an alert appears with the coordinates of the click

How to:

## Step 2: show the coordinates (project)

- When the user clicks on the map an alert appears with the coordinates of the click
- How to:
  - The HTML/file is identical, we added management of a click event in the table scient.

40.40.45.45. 5 000

#### Step 2: show the coordinates (project)

- When the user clicks on the map an alert appears with the coordinates of the click
- How to:
  - The HTML file is identical, we added management of a click event in the JavaScript

- When the user clicks on the map an alert appears with the coordinates of the click
- How to:
  - The HTML file is identical, we added management of a click event in the JavaScript
  - We apply the *on* method to the map to catch *click* events



- When the user clicks on the map an alert appears with the coordinates of the click
- How to:
  - The HTML file is identical, we added management of a click event in the JavaScript
  - We apply the on method to the map to catch click events
    - the first parameter is the name of the event we want to capture
    - the second parameter sa callback that takes the event description as
    - the call act the part and the containing data extracted from the event
    - the event
    - we ext

- When the user clicks on the map an alert appears with the coordinates of the click
- How to:
  - The HTML file is identical, we added management of a click event in the JavaScript
  - We apply the on method to the map to catch click events
    - the first parameter is the name of the event we want to capture
    - the second parameter is a callback that takes the event description as a parameter
    - the callback displays an aleg containing data extracted from the event descriptor e
    - the event descriptor is an one
    - we extract the lat and a fields in the latt



- When the user clicks on the map an alert appears with the coordinates of the click
- How to:
  - The HTML file is identical, we added management of a click event in the JavaScript
  - We apply the on method to the map to catch click events
    - the first parameter is the name of the event we want to capture
    - the second parameter is a callback that takes the event description as a parameter
    - the callback displays an alert containing data extracted from the event descriptor e
    - the event descriptor is an object
    - we extract the lat and log fields in the lating field.



- When the user clicks on the map an alert appears with the coordinates of the click
- How to:
  - The HTML file is identical, we added management of a click event in the JavaScript
  - We apply the on method to the map to catch click events
    - the first parameter is the name of the event we want to capture
    - the second parameter is a callback that takes the event description as a parameter
    - the callback displays an alert containing data extracted from the event descriptor e
    - the event descriptor is an object
    - we extract the lat and ing fields in the lating field.



- When the user clicks on the map an alert appears with the coordinates of the click
- How to:
  - The HTML file is identical, we added management of a click event in the JavaScript
  - We apply the on method to the map to catch click events
    - the first parameter is the name of the event we want to capture
    - the second parameter is a callback that takes the event description as a parameter
    - the callback displays an alert containing data extracted from the event descriptor e
    - the event descriptor is an object
    - we extract the lat and lng fields in the lating field.



#### Step 2: Lab activity

• create a named <div> and write within the coordinates. Use

document.getElementById("myDiv").textContent = ...



 Each click on the map adds a marker, and their coordinates are shown on the page



How to:

- Each click on the map adds a marker, and their coordinates are shown on the page
- How to:
  - We and a divitor the coordinates in the htm
  - new marker
  - The c
  - DOM

(ロ) (団) (団) (ほ) (ほ) (ほ) (の)

- Each click on the map adds a marker, and their coordinates are shown on the page
- How to:
  - We add a div for the coordinates in the html
  - In the JavaScript waterdrip the event callback the
  - The cool of Europe DOM

- Each click on the map adds a marker, and their coordinates are shown on the page
- How to:
  - We add a div for the coordinates in the html
  - In the JavaScript we add to the event callback the creation of the new marker

《□▶ 《圖▶ 《意▶ 《意》 - 意 - 釣@♡

- Each click on the map adds a marker, and their coordinates are shown on the page
- How to:
  - We add a div for the coordinates in the html
  - In the JavaScript we add to the event callback the creation of the new marker
    - its position is computed using the lating field in the event descriptor

- Each click on the map adds a marker, and their coordinates are shown on the page
- How to:
  - We add a div for the coordinates in the html
  - In the JavaScript we add to the event callback the creation of the new marker
    - its position is computed using the lating field in the event descriptor
  - The coordinates are appended to the list in a div element of the DOM

#### Step 3: Lab activity

 Reverse the order of coordinates: longitude is shown first in the bottom list

A progressive index is assigned to each new point

 The index is shown in the list and added as a title field in the marker definition

the me

How to

- A progressive index is assigned to each new point
- The index is shown in the list and added as a title field in the marker definition
  - the title field is automatically displayed when the mouse hovers on the market.
- How to:

- A progressive index is assigned to each new point
- The index is shown in the list and added as a title field in the marker definition
  - the title field is automatically displayed when the mouse hovers on the marker





- A progressive index is assigned to each new point
- The index is shown in the list and added as a title field in the marker definition
  - the title field is automatically displayed when the mouse hovers on the marker
- How to:



- A progressive index is assigned to each new point
- The index is shown in the list and added as a title field in the marker definition
  - the title field is automatically displayed when the mouse hovers on the marker
- How to:
  - HTML is the same
  - We add a new elobe
  - The event calls
  - The value of n is also in
  - The mar
  - the marke

- A progressive index is assigned to each new point
- The index is shown in the list and added as a title field in the marker definition
  - the title field is automatically displayed when the mouse hovers on the marker
- How to:
  - HTML is the same
  - We add a new global variable n in the JavaScript
  - The event callback increments the variable each time it is run
  - The value of n laters blad
  - The marker extions

    The marker extinons

    The marke



- A progressive index is assigned to each new point
- The index is shown in the list and added as a title field in the marker definition
  - the title field is automatically displayed when the mouse hovers on the marker
- How to:
  - HTML is the same
  - We add a new global variable n in the JavaScript
  - The event callback increments the variable each time it is run
  - The value of n is displayed on each line in
  - The marker continues takes a second parameter containing the marker options.

- A progressive index is assigned to each new point
- The index is shown in the list and added as a title field in the marker definition
  - the title field is automatically displayed when the mouse hovers on the marker
- How to:
  - HTML is the same
  - We add a new global variable n in the JavaScript
  - The event callback increments the variable each time it is run
  - The value of n is displayed on each line in the list
  - The marker constructor now takes a second parameter containing the marker options.

- A progressive index is assigned to each new point
- The index is shown in the list and added as a title field in the marker definition
  - the title field is automatically displayed when the mouse hovers on the marker
- How to:
  - HTML is the same
  - We add a new global variable n in the JavaScript
  - The event callback increments the variable each time it is run
  - The value of n is displayed on each line in the list
  - The marker constructor now takes a second parameter containing the marker options
    - among which the title option



- A progressive index is assigned to each new point
- The index is shown in the list and added as a title field in the marker definition
  - the title field is automatically displayed when the mouse hovers on the marker
- How to:
  - HTML is the same
  - We add a new global variable n in the JavaScript
  - The event callback increments the variable each time it is run
  - The value of n is displayed on each line in the list
  - The marker constructor now takes a second parameter containing the marker options
    - among which the title option



## Step 4: Lab activity

- Configure the marker as draggable (ignore that the displayed coordinates become inconsistent)
  - hint: in the marker variable definition add a draggable: true property after the title, separated by a ","

- Record the markers in an array to have them accessible
  - in the previous steps the marker was a local variable in the callback
- How to:

- Record the markers in an array to have them accessible
  - in the previous steps the marker was a local variable in the callback



- Record the markers in an array to have them accessible
  - in the previous steps the marker was a local variable in the callback
- How to:
  - Create an array for the markers
  - Pushin arke a in the arra
  - n ind

- Record the markers in an array to have them accessible
  - in the previous steps the marker was a local variable in the callback
- How to:
  - Create an array for the markers
  - Pushinarkers in the array.

- Record the markers in an array to have them accessible
  - in the previous steps the marker was a local variable in the callback
- How to:
  - Create an array for the markers
  - Push markers in the array

- Record the markers in an array to have them accessible
  - in the previous steps the marker was a local variable in the callback
- How to:
  - Create an array for the markers
  - Push markers in the array
  - n index corresponds to array length

- Record the markers in an array to have them accessible
  - in the previous steps the marker was a local variable in the callback
- How to:
  - Create an array for the markers
  - Push markers in the array
  - n index corresponds to array length
    - no need to increment it

## Step 5: Lab activity

- Create a button that fades-out the markers
- Hint
  - · loop through all items in the array with a for loop

```
for (let m in markers) {...}
```

• use the setOpacity(0.5) on each marker

# Step 6: all markers in a layer (project)

Having all markers in a layer is more practical than in an array





# Step 6: all markers in a layer (project)

- Having all markers in a layer is more practical than in an array
- How to:
  - HTML is always the same
  - Replace type array with a layerGroup object added to the man
  - Replace
  - layeri
  - Add\_
  - The
  - See the eff

## Step 6: all markers in a layer (project)

- Having all markers in a layer is more practical than in an array
- How to:
  - HTML is always the same
  - Replace the array with a layerGroup object added to the map
    - Replace the prof operation with an addkayer applie
  - layers oup
  - The Add A The Ad
  - See the effe

- Having all markers in a layer is more practical than in an array
- How to:
  - HTML is always the same
  - Replace the array with a layerGroup object added to the map (markers)
  - Replace the plan operation with an add-aver a
  - Add Xlaver do find property and the management of the control of the
  - The conf

- Having all markers in a layer is more practical than in an array
- How to:
  - HTML is always the same
  - Replace the array with a layerGroup object added to the map (markers)
  - Replace the push operation with an addLayer applied to the layerGroup
  - Add a layer do fit of protein the map to toggle layer visibility
  - The coriner of cation takes avo object

- Having all markers in a layer is more practical than in an array
- How to:
  - HTML is always the same
  - Replace the array with a layerGroup object added to the map (markers)
  - Replace the push operation with an addLayer applied to the layerGroup
  - Add a layer control icon to the map to toggle layer visibility

- Having all markers in a layer is more practical than in an array
- How to:
  - HTML is always the same
  - Replace the array with a layerGroup object added to the map (markers)
  - Replace the push operation with an addLayer applied to the layerGroup
  - Add a layer control icon to the map to toggle layer visibility
  - The control creation takes two object arguments

One of the base layers (radio button just one)

- Having all markers in a layer is more practical than in an array
- How to:
  - HTML is always the same
  - Replace the array with a layerGroup object added to the map (markers)
  - Replace the push operation with an addLayer applied to the layerGroup
  - Add a layer control icon to the map to toggle layer visibility
  - The control creation takes two object arguments
    - One for the base layers (radio button, just one)
  - See the effection the system by then toggraph in the man



- Having all markers in a layer is more practical than in an array
- How to:
  - HTML is always the same
  - Replace the array with a layerGroup object added to the map (markers)
  - Replace the push operation with an addLayer applied to the layerGroup
  - Add a layer control icon to the map to toggle layer visibility
  - The control creation takes two object arguments
    - One for the base layers (radio button, just one)
    - One for the overlay layers (multiple choice)
  - See the effect on the layers builton top right in the map



- Having all markers in a layer is more practical than in an array
- How to:
  - HTML is always the same
  - Replace the array with a layerGroup object added to the map (markers)
  - Replace the push operation with an addLayer applied to the layerGroup
  - Add a layer control icon to the map to toggle layer visibility
  - The control creation takes two object arguments
    - One for the base layers (radio button, just one)
    - One for the overlay layers (multiple choice)
  - See the effect on the layers button top-right in the map



#### Step 6: Lab activity

• move the markers layerGroup in the base layers. Any change?



- It is handy to have a standard string representation of a piece of data (serialization)
  - e.g. to store the data in a file
- The GeoJSOM representation can be easely transfer and into a

  ISON strand popular and a second can be easely transfer as a second can be easel
- 14/
- The toge
- object w
- The stringify
- The string is fin

- It is handy to have a standard string representation of a piece of data (serialization)
  - . e.g. to store the data in a file
- The GeoJSOW representation can be easely transformed into a

  ISON string land viscoversa.
- Wa wan Ella in 1117 days and
- The together me
   object with the General control of the co
- The stringify weth
- The string is fin

- It is handy to have a standard string representation of a piece of data (serialization)
  - . e.g. to store the data in a file
- The GeoJSON representation can be easely transformed into a JSON string, and viceversa
- We want in print in the consplet the ISON string for our markers
- The toge and method converts the markers layer into a JavaScrip object with the seek stock at IAFRICA
- The stringif
- The string is fir



- It is handy to have a standard string representation of a piece of data (serialization)
  - . e.g. to store the data in a file
- The GeoJSON representation can be easely transformed into a JSON string, and viceversa
- We want to print in the console the JSON string for our markers
- The toged son method converts the markers layer into a JavaScrip object with the Geod Son format. A FRICA
- The stringify wethor serial serial estimates object as a String object
- The string is fir



- It is handy to have a standard string representation of a piece of data (serialization)
  - . e.g. to store the data in a file
- The GeoJSON representation can be easely transformed into a JSON string, and viceversa
- We want to print in the console the JSON string for our markers
- The toGeoJSON method converts the markers layer into a JavaScript object with the GeoJSON format
  - · alas, in this way we take the title
- The stringify method serial results object as a String object
- The string is finally recorded.



- It is handy to have a standard string representation of a piece of data (serialization)
  - . e.g. to store the data in a file
- The GeoJSON representation can be easely transformed into a JSON string, and viceversa
- We want to print in the console the JSON string for our markers
- The toGeoJSON method converts the markers layer into a JavaScript object with the GeoJSON format
  - alas, in this way we lose the title field
- The stringify method serial zesone object as a String object
- The string is finally recorded out on the display



- It is handy to have a standard string representation of a piece of data (serialization)
  - . e.g. to store the data in a file
- The GeoJSON representation can be easely transformed into a JSON string, and viceversa
- We want to print in the console the JSON string for our markers
- The toGeoJSON method converts the markers layer into a JavaScript object with the GeoJSON format
  - alas, in this way we lose the title field
- The stringify method serializes the object as a String object
- The string is finally recorded put on the display



- It is handy to have a standard string representation of a piece of data (serialization)
  - . e.g. to store the data in a file
- The GeoJSON representation can be easely transformed into a JSON string, and viceversa
- We want to print in the console the JSON string for our markers
- The toGeoJSON method converts the markers layer into a JavaScript object with the GeoJSON format
  - alas, in this way we lose the title field
- The stringify method serializes the object as a String object
- The string is finally recorded put on the display



#### Step 7: Lab activity

- Copy the generated GeoJSON and feed it to an online viewer, like https://www.geometrymapper.com/
- Is there any way to record the title field in the JSON string?
- Study the GeoJSON format in the console and find a solution