



**HKUST**  
VISLAB

# **COMP 4462**

## **Data Visualization Tutorial**

Leo Yu Ho, Lo  
Wenchao Li

Friday 24 April, 2019  
<https://bit.ly/vis-t08>

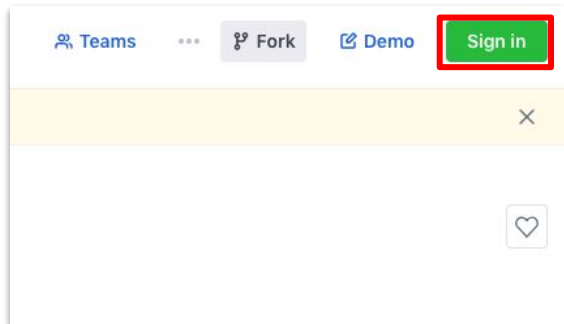
# Visualization with D3.js

- SVG (Scalable Vector Graphics)
  - An extension of HTML for representing scalar graphics in XML syntax
  - Available in all the web browsers
- D3.js
  - The most widely used visualization library
  - The library behind Vega, Vega-Lite and Altair
  - Binding data with SVG DOM, marking data points visually onto screen
  - Imperative syntax, compared to the declarative syntax of Vega-Lite and Altair
- Why D3.js
  - ❌ Exploring a dataset, use Altair with Python or Tableau instead
  - ❌ Embed visualization in web applications, use Vega-Lite instead
  - ✅ Make customized plots, customized interactions or transitions
- Cost
  - Much more coding, much easier to make mistakes
  - Check [Vega-Lite Gallery](#) and [Vega Gallery](#) before committing to D3.js

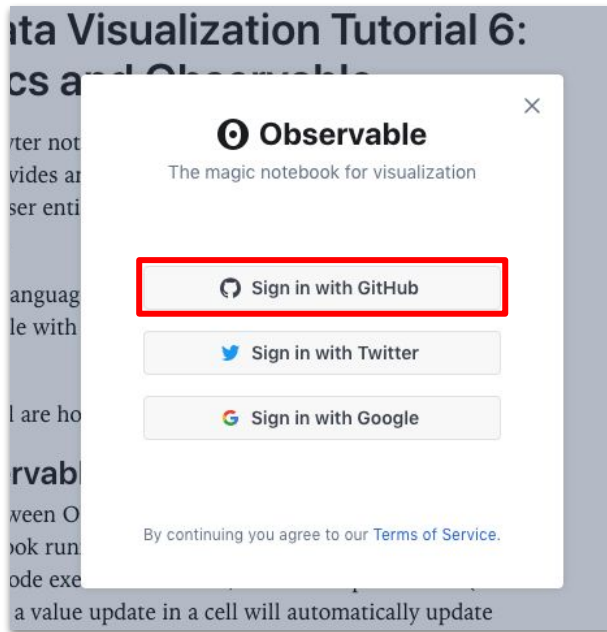
# Sign in Observable

1. Go to the [notebook of this tutorial](#)

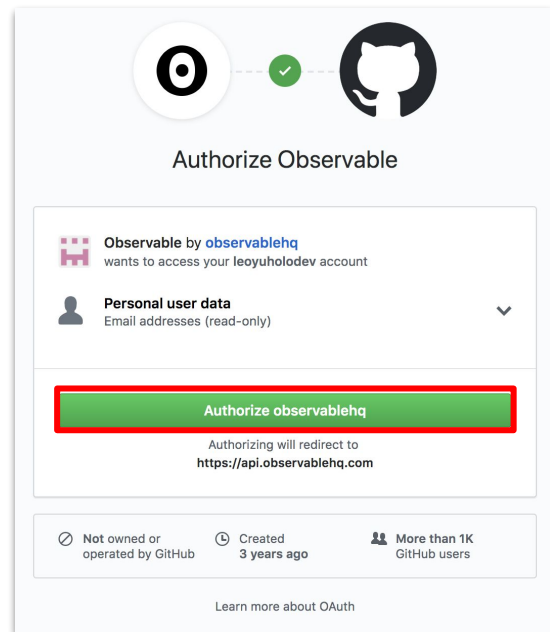
2. Click Sign in



3. Sign in with GitHub (recommended)



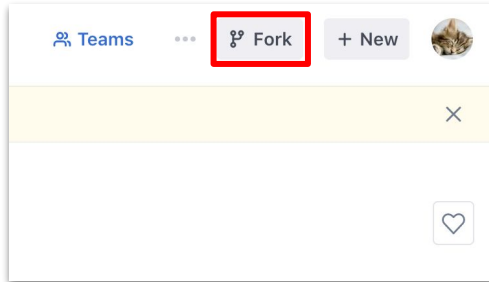
4. Authorize observablehq



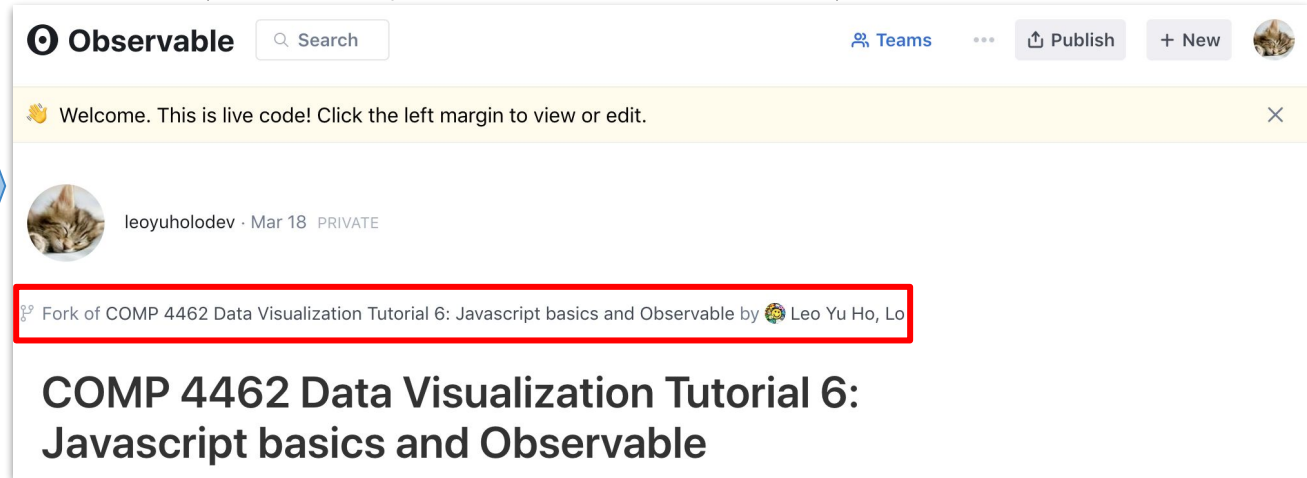
# Fork Observable notebook

1. Go to the [notebook of this tutorial](#)

2. Click Fork



3. Check if you're working on your copy of the notebook (otherwise, your work will not be saved)



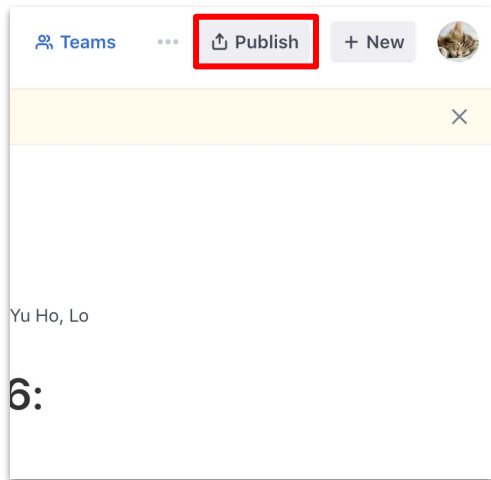
# Visualization with D3.js

- See the [Observable notebook of this tutorial](#)
- SVG
- Scales
  - Linear scale (numeric, color)
  - Time scale
  - Point scale (categorical)
- Coordinate System
  - Axes
  - Cartesian coordinate (X and Y)
  - Polar coordinate (angular and distance)
- Marks and channel
  - Point: scatter plot
  - Line: (multi-)line chart, parallel coordinate, radar chart
- Selection

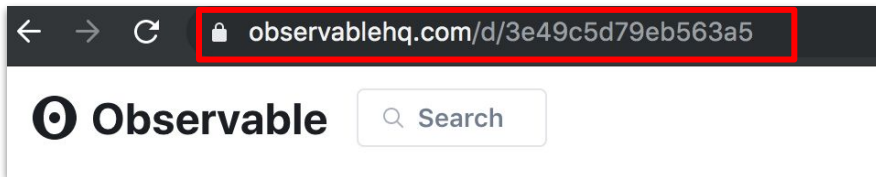
# Publish your Observable notebook

1. In your working copy of the notebook

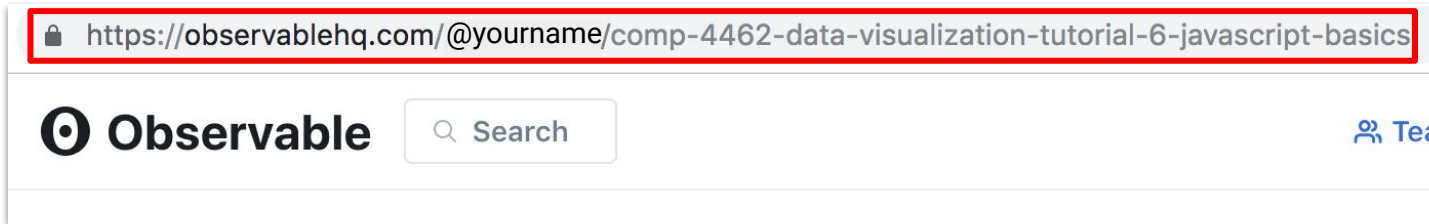
2. Click Publish



3. The URL changes from a hash...



4. To a meaningful URL, this URL is publicly accessible



# Lab exercise

- Tasks

- Sign in [Observable](#)
- Open [this Observable notebook](#) and fork it (otherwise, your work will not be saved)
- Read through the notebook and fill in the “TODO” cells
- If you are going to use D3.js in your project, make sure you understand these concepts:
  - What is SVG? What is “selection” in D3.js?
- You will make 4 visualizations:
  - Scatter plot, multi-line chart, parallel coordinates, and radar chart
- Publish your notebook when finished
- Copy the URL of your Observable notebook and submit to Canvas
  - The URL should be something like:
    - <https://observablehq.com/@yourname/comp-4462-data-visualization-tutorial-8-visualization-wit>
- Help us improve this tutorial by answering [the questionnaire](#)

- Optional

- Like [our Observable notebook](#) ❤️❤️❤️ and star [our GitHub repository](#) ★★ ★ Thank you! ❤️
- Learn about [how selection works](#) and check out [the notebooks created by D3](#)

# More on D3.js

- More on SVG
  - A [list of all the SVG elements](#) and a [list of all the SVG attributes](#)
- More on D3.js
  - A [long list of d3.js examples](#)
  - Even more examples on [Block Builder](#)
  - [Observable notebooks by D3](#)
  - [Color schemes available in D3.js](#)
  - The reference book of these tutorials:
    - [Interactive Data Visualization for the Web](#) by [Scott Murray](#)
    - Code examples are available on [this GitHub repository](#)
- Interactions, transition and maps
  - Next tutorial: Visualization and Interaction with D3.js



# Next tutorial

Visualization and  
Interaction with D3.js

- We will use [Observable](#) and [D3.js](#) again
- And learn about adding interactions to our visualizations!