ELECTRONIC PUBLISHING
AND DIGITAL STORYTELLING
Lesson 7

DATA VISUALIZATION ON THE WEB

OUTLINE

1

STRATEGIES TO VISUALIZE DATA ON THE WEB

Overview

2

HANDS-ON BOKEH

Still Jupyter notebooks

3

HANDS-ON GOOGLE CHARTS

Javascript libraries

4

HANDS-ON D3.JS

Javascript libraries

ELECTRONIC PUBLISHING
AND DIGITAL STORYTELLING
Lesson 7

STRATEGIES TO VISUALIZE DATA ON THE WEB

A FEW WAYS

Some libraries allow to export visualizations as png/jpg/html+js. Others allow to build server-side applications

PYTHON LIBRARIES

Online and desktop tools that allow to manipulate data and export png/jpg

WYSIWYG TOOLS

Many libraries allow to query static/dynamic data and visualize on the fly.

JAVASCRIPT

FOR EXAMPLE

Bokeh, Plotly, pygal, Gleam

PYTHON LIBRARIES

Tableau, Infogram, Google charts/spreadsheet

WYSIWYG TOOLS

Google Charts, D3.js, and many others

JAVASCRIPT

A FEW DRAWBACKS

PYTHON LIBRARIES

Requires good programming skills and know-how on specific libraries for plotting/web

Bokeh, Plotly, pygal, Gleam

WYSIWYG TOOLS

Less technical skills, but more restrictions (on data manipulation, chart selection, pricing)

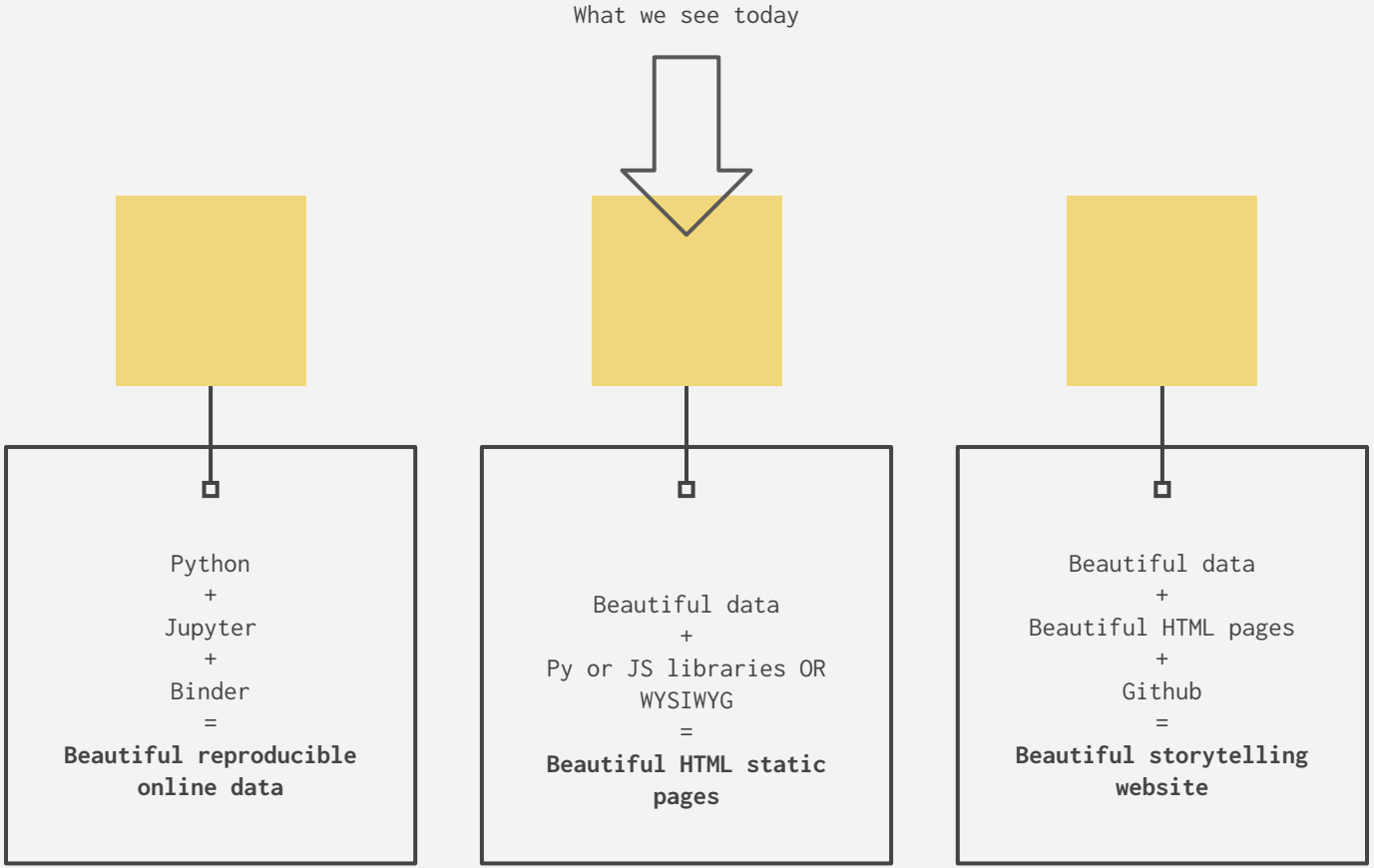
Tableau, Infogram, Google charts/spreadsheet

JAVASCRIPT

Requires know-how of dedicated libraries and acquaintance with how the web works

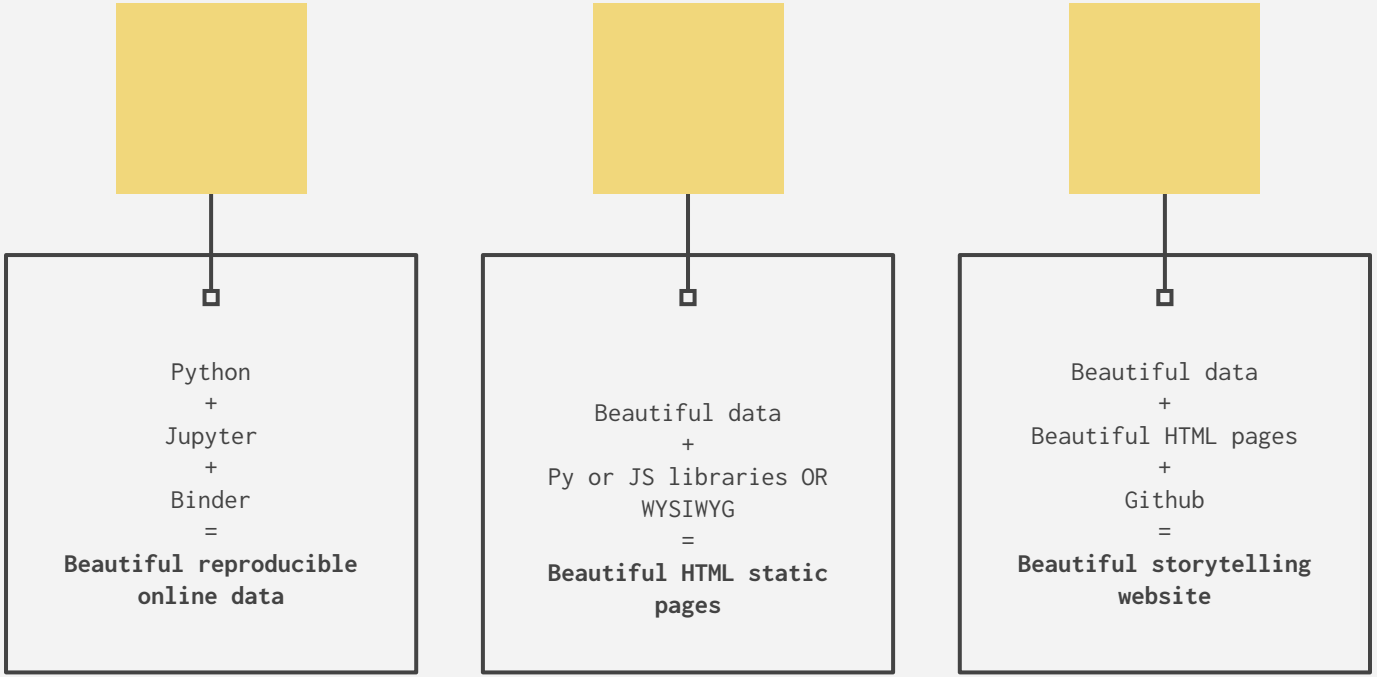
Google Charts, D3.js, and many others

OUR WAY


What we see today
Python
+
Jupyter
+
Binder
=
Beautiful reproducible online data
Beautiful data
+
Py or JS libraries OR WYSIWYG
=
Beautiful HTML static pages
Beautiful data
+
Beautiful HTML pages
+
Github
=
Beautiful storytelling website

OUR WAY

Separate data filtering and analysis from the visualization. **Drawback**: data are static (if data change in the original source you need to recompute everything)



ELECTRONIC PUBLISHING
AND DIGITAL STORYTELLING
Lesson 7

HANDS-ON BOKEH

Reference documentation:
https://docs.bokeh.org/en/latest/

INSTALL THE PYTHON LIBRARIES AND MOVE TO THE TUTORIAL

Bokeh
pip install bokeh

Open the repository of tutorials
and open the file **web_data_viz.ipynb**
in the browser

HANDS-ON GOOGLE CHARTS

Reference documentation:
https://developers.google.com/chart/interactive/docs/quick_start

PREPARE THE ENVIRONMENT

Open the repository of tutorials, and download the files

- **google_charts_tutorial.html**
- **periods_dates_sample.csv**
- **jquery-1.10.1.min.js**
- **jquery.csv-0.71.min.js**
- **loader.js**

in the folder with the tutorials.

PREPARE THE ENVIRONMENT

Disable CORS Cross-origin restrictions

To work on HTML+CSV files in a browser you need to have either Safari browser or to create a local web server (so that you can access csv from the html file in other browsers like Chrome or Firefox)

If you have safari:
open Safari browser
go to the menu **Develop**
select **Disable cross-origin restrictions**
and open the HTML file in the browser.

INSTALL PYTHON LIBRARIES

If you don’t have Safari install the following python library in the shell

pip install simple-http-server

In the shell open the folder where you downloaded the html+csv files (use the command **cd /path/to/folder**)

Run the local server
python -m http.server 8000

Go to the browser (any) and type the address
localhost:8000

Select from the list the HTML file

LOAD DATA IN ONLINE REPOSITORY

Alternatively, upload your data online and use the URL of the CSV rather than the local file URL in javascript

HANDS-ON d3.js

Reference documentation:
https://d3js.org/
Tutorials:
https://www.d3-graph-gallery.com/

PREPARE THE ENVIRONMENT

Open the repository of tutorials, and download the files

- **d3_tutorial.html**

in the folder with the tutorials.

Open it in the browser.

HOMEWORK

No homework!