

Objectives

With these assignments you will learn how to

- learn theoretical aspects about the assessment of visualizations;
- apply data filtering and preprocessing to prepare data for further analysis and visualization;
- apply D3.js for visualization using basic charts.

For your programming tasks, you will use JavaScript for preprocessing and visualization. You are expected to comment and document your code in a focused and clear manner.

Your solutions must be uploaded via Moodle by **December 09, 2021, 9am (UTC+1)** as one ZIP file that contains all answers and source files. The naming convention for this ZIP file is **sheet1_<group_name>.zip**.

Instructions

Implementation As visualization framework we use *D3.js* and its respective language JavaScript; for details we refer to its documentation¹.

The implementations with D3.js should be implemented as a web page based using the framework. Make sure that relative paths are used. Supplementary exercise-specific data should be organized in a sub-folder called “data”. Color schemes, in particular, can be explored and loaded via colorbrewer2.org². Local provisioning for debugging and testing can be done using a local web server such as the *http-server* by *node*³. The server can be installed with the following command:

- `npm install http-server -g`

In order to load e.g. CSV files you need to start a local server by

- `http-server -a 127.0.0.1 -o`

in your working directory containing both the sources and the data.

Pair Programming On these assignments, you are encouraged (not required) to work with a partner provided you practice pair programming. Pair programming “is a practice in which two programmers work side-by-side at one computer, continuously collaborating on the same design, algorithm, code, or test.” One partner is driving (designing and typing the code) while the other is navigating (reviewing the work, identifying bugs, and asking questions). The two partners switch roles every 30–40 minutes and, on demand, brainstorm.

Violation of Rules A violation of rules results in grading the affected assignments with 0 points.

- Writing code with a partner without following the pair programming instructions listed above (e.g., if one partner does not participate in the process) is a serious violation of the course collaboration policy.
- Plagiarism represents a serious violation of the course policy.

¹D3 documentation can be found at <https://github.com/d3/d3/blob/master/API.md>

²<https://colorbrewer2.org>

³<https://nodejs.org/en>

Exercise 2.1: Interactive Scatter Plot (8 Points)

On the first exercise sheet, the data set *UnRegionsGdp.csv* has been visualized using a scatter plot. In this exercise we will extend a baseline implementation provided by the file *ScatterUnRegionsGdp.html* by various interaction techniques.

- As a third visual variable of a two-dimensional scatter plot, we can use the radius of the circles. Apply this idea to visualize the size of population in the visualization. (2 Points)
- Interaction enables the user to capture more information from a single visualization. Extend the implementation from part a) to enable details-on-demand for single data points, which show the GDP and the HDI. (2 Points)
- Further extend the results from b) to filter all points that belong to the same region provided by the legend. When a region is selected, all data points should be highlighted in an appropriate way. (2 Points)
- Extend the version of a) by adding zoom and brush functionalities. (2 Points)

Exercise 2.2: Two Layouts for ETF Data (8 Points)

The data set *Turnover.csv* contains the turnover in million Euro of 10 large Exchange Traded Funds (ETF) traded at Xetra⁴ for the months July, August, September and October. The goal of this exercise is to apply two layouts to visualize the distribution.

- Create a pie chart to capture the division of the traded ETFs for October. Make sure that the visualization applies an appropriate color scheme, shows labels, and numbers in an effective way. (4 Points)
- A stacked bar chart can be used to capture the division at various points in time. Create a stacked bar chart for four months that display the division. Make sure that the visualization applies an appropriate color scheme and shows labels and numbers. (4 Points)

Exercise 2.3: Visualization of the Covid-19 Dataset (8 Points)

The “COVID19 Cases Worldwide” dataset *owid-covid-data.csv*⁵ contains the latest publicly available data on COVID-19, including a daily update of the situation, epidemiological curve, and global geographic distribution (EU/EEA and UK, worldwide).

In this exercise, your task is to create a visualization inspired by the visualization of the French first names shown in the lecture⁶. No geographic visualization (e.g., map) is required, that is, the visualization should view the Covid-19 situation and evolution in the different countries in a more analytical way.

The minimum criteria are listed below:

- At least one interaction technique
- A reasonable use of a color scheme
- At least two points in time

Your submission is evaluated based on its implementation and its theoretical foundation.

Exercise 2.4: Evaluation of Visualization Examples (2 Points)

In the zip file of this exercise sheet you will find four screenshots of visualization examples from different applications and domains. Describe at least one negative aspect of each visualization and explain why you consider this aspect to be negative.

⁴Deutsche Börse Xetra

⁵<https://github.com/owid/covid-19-data/tree/master/public/data/>

⁶70 years of names in France, dataaddict.fr (2016)