Exercise 2: Data Preprocessing and d3.js Basics   **(20 points)**

Due: 29.04.2023 10AM

Important Notes:
- Please find yourself in the same group of two students like the last exercise. Only 1 member of the group must submit the exercise in ILIAS.
- Please add both student names on the html file so it is visible when opened. Please also add your text answers to the html file so it is visible when opened.
- Submission: Zipped linechart folder including all files (index.html, index.js, index.css, data.js, d3.js)

Task 0: Local Server Setup   **(0 points)**

Setting up a local server is good for testing your web projects directly on your computer and lets you see changes instantly.

Quick Setup:
First, make sure Python 3 is installed on your computer and executable from your terminal.
Extract the folder from the zip File. Open your terminal in the folder or navigate to it.
Type **python -m http.server 8000**.
Open your web browser and go to **http://localhost:8000** to see your site live. You can update your files and refresh the page in the browser to see your changes right away.

Within the folder *histogram* there are 4 files:
- **index.html**
  The main entry point of the visualization. When your implementation is finished, opening this file with a browser should show a *histogram*. Currently, your browser should show a heading but no visualization.
- **index.js**
  The main JavaScript entry point which implements a histogram.
- **index.css**
  Implements CSS Rules for specific elements.
- **data.js**
  Initializes a variable called *data* and reflects the dataset we want to visualize. It represents unemployment rate of U.S. counties.

Hint: Start editing the files via your IDE of choice (e.g. VS Code). Make use of the console output of your browser to check for possible JavaScript errors.

Task 1: Data Preprocessing (10 points)

Task 1.1: Cleaning
It seems like some employment rates do not make sense and were incorrectly entered to the file - data cleaning is needed. View the data file and chose a applicable method to clean the data as discussed in the lecture.
- Which exact data cleaning method would you chose? Justify you answer in 1 to 2 sentences regarding the given data set.
- Implement your chosen data cleaning approach in the *preprocessData()* method as further indicated in the index.js.
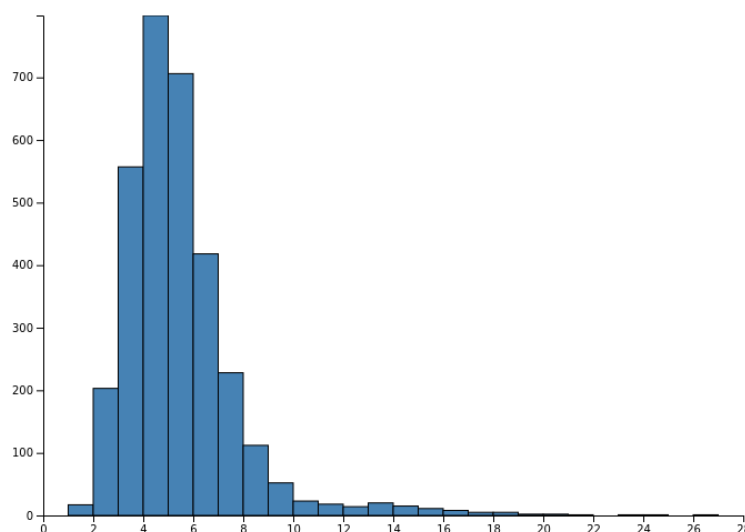
Task 1.2: Binning
To create our histogram in the next task, we want to partition our data into equal-width bins.
- How many data bins would you apply? Justify you answer in one sentence with respect to the given data set.
- Implement data binning at the given position as described in the index.js file.

Task 2: Create a Histogram (10 points)

Task 2.1: Cleaning
Goal of this exercise is to implement a histogram with D3. Attached to this exercise you will find a folder called *histogram* which contains an unfinished implementation of a histogram. Your task is to finish the implementation such that opening the *index.html* shows a histogram as depicted in figure above. This example shows the unemployment rate of U.S. counties as of August. Follow the steps described as comments within the dedicated file. Each comment starting with *TASK* indicates a position you have to add code.

Task 2.2
- How would the histogram look like without data cleaning? Describe the difference in one answer.