

Exercise 10: Temporal Visualization

(20 points)

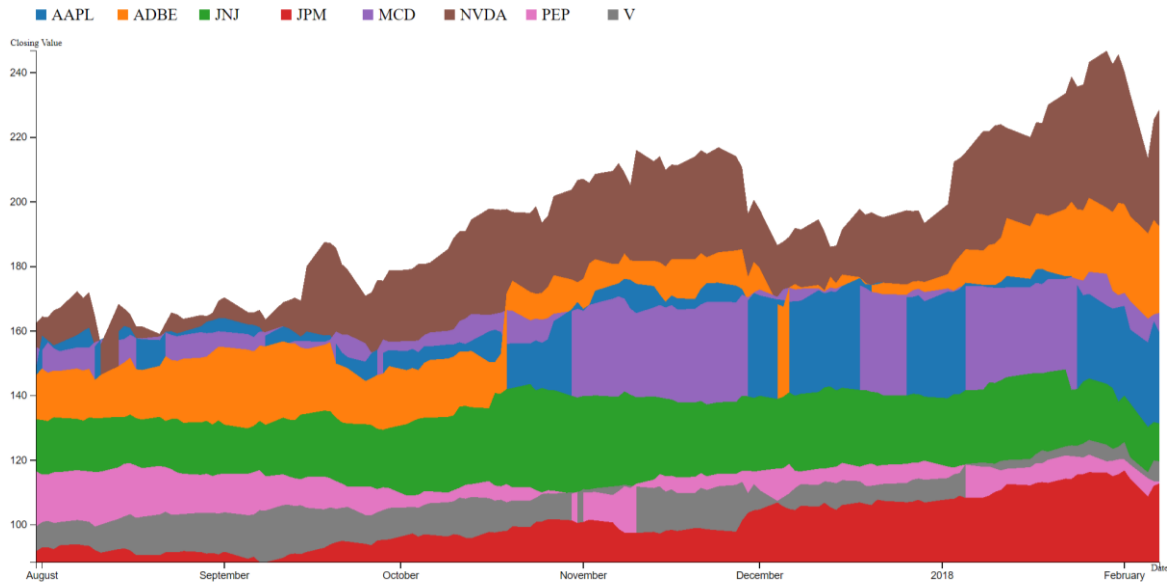
Due: 24.06.2024 10AM

Contributor 1:

Contributor 2:

Task 1: Implementation

(11 points)



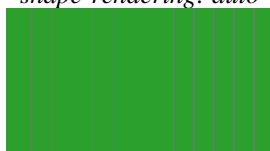
For this exercise, your task is to create a braided graph of the appended *stons.js* dataset as shown in the figure above. You may use your own previously written code as orientation.

Mind the following hints as orientation:

When building the visualization, it makes sense to group the data by its days. For each day, create a *g* element. Draw a respective *path* element between the current and the next timestamp for each stock within each created *g*. Think of plotting the higher value first to make the smaller elements visible.

Additional Hint: To avoid vertical rendering lines between the paths, set the css attribute *shape-rendering*. (<https://developer.mozilla.org/en-US/docs/Web/SVG/Attribute/shape-rendering>)

shape-rendering: auto



shape-rendering: crispEdges



Task 2: Theory

(9 points)

Imagine you are employed at a global bank. Your task is to **design visualizations** to display **multiple stock prices over a six-month period** like the data in Task 1. (Precisely but briefly) identify **three different scenarios** within the banking context that would require **distinct visualization designs**, as discussed in the lecture. One of these visualizations should be a braided graph (as displayed in Task 1). Justify your decisions.

For each scenario, provide a brief explanation following this structure:

1. **Domain Situation:**
2. **Visualization Design Description:**
3. **Justification:**

Submission: Zipped folder including all files of the programming exercise (index.html, index.js, index.css, data.js) and a PDF/text file with the answers to the theoretical questions. Make sure the visualization is directly displayed when opening index.html.

Please find yourself in Groups of **2 Students** and clearly state both names in your submission. Only 1 member of the group must submit the exercise in ILIAS.