Criteria

- Does it appears has expected on first load?
- Does interactions work as expected?
- Are there transitions?
- Does the overall thing respects CRAP design principles and is appealing?
- Is the code structured in meaningful components?
- Is the code readable?

You can ignore :

- Responsive design: just make it work on a normal computer resolution.
- Cross browser compatibility: I will run your code in Chrome.

I will run your code :

- Unzip the archive
- Run a server with `python3 -m http.server` and it should work from here.

Attach the data in your archive, if you have done some conversion from CSV to JSON using python, attach the converted data.

Deadline: Sunday 2nd April, 23h59 CET.

Suggestion

1. Define the key interfaces / data representation
2. Delegate **meaningful** work unit to specific member.

e.g.:

- One does the form
- One does the visualization charts
- One does the event handling

Projects

Project 1: Scale changing line chart

Source data: https://www.kaggle.com/datasets/danielfesalbon/covid-19-global-reports-early-march-2022

Any country.

1. Project a line chart number of case over time.
2. Add two text inputs that accept dates like `2019-12-31`.
3. When both of these text input have a valid value, it changes the considered dates inside the chart.
4. Animate the date change on the chart.
5. Clicking once on the chart selects a first date and mark it with a bar, a second click will finish selecting a period. The period should be visible before a click.
6. Left click resets the dates.

Project 2: Projection

2.1 Source data: https://www.kaggle.com/datasets/iffee6/labour-characteristics-in-pakistan

2.2 Source data: https://www.kaggle.com/datasets/kamilpytlak/personal-key-indicators-of-heart-disease

1. By default display the total number of elements in a bar on a chart.
2. Display on top the different dimensions with associated checkboxes.
3. When a first checkbox is selected the `total` bar dissappears and instead the sum accross each value in the dimension is displayed.
4. Other checkboxes will had new bars containing the sum accross each value of the dimension.
5. When at least two dimensions are selected, hovering over a bar selects this value for this dimension and the other dimensions bar should only reflect the data matching the selected dimension.

Illustration: https://excalidraw.com/#json=03A*85*1pHqxi0*cvlJX6,Axpo7*dTsPbVvH4gLyH1wg

Project 3: Many lines

3.1 Source data: https://www.kaggle.com/datasets/danielfesalbon/covid-19-global-reports-early-march-2022

3.2 Source data (countries are loss type): https://www.kaggle.com/datasets/piterfm/2022-ukraine-russian-war

1. Start with an empy line chart.

2. Create a search box, typing in the search box should display a list of matching countries (or loss types).
3. Clicking an item marks it as selected.
4. Selected items are displayed in another area, clicking one unselect it.
5. Selected items should be displayed as different lines.
6. Add a total line at the top (that additions all the countries if at least two are selected)
7. Add a select enabling to switch between cumulated value and daily values

Project 4: Data explorer

Source data: https://www.kaggle.com/datasets/kamilpytlak/personal-key-indicators-of-heart-disease

1. Create a checkbox or a select for every nominal attribute
2. Create a pie chart to represent the value of the target attribute / class (heart disease)
3. Make an histogram of the continuous variables
4. Every interaction with the checkboxes/select should modify the selection
5. Animate the transition on change