

Question 1: Reasons for Pattern Choices

The question entailed the problem of having a certain set amount of data represented in different non-discrete views. This means that the types of views that need to be generated are not a specific set, but a whole assortment of possibilities resulting from a combination of provided parameters. That is why we chose the **strategy** pattern to build this application around.

The application utilizes two main types of view manipulation:

- Selection – The application requires isolation of certain data from others without modification to the original data.
- Sorting – The application requires reordering of data without modification to original data.

Therefore we came up with a scheme that has a Pipe Architecture composed of selection and sorting objects.



Whenever one of the parameters is changed within the GUI, the corresponding slot is filled with the required object, and the pipe is executed. The output is a filtered output of the data.