

Fullstack Home Assignment

Thank you for taking the time to complete this assignment. The purpose of this task is to evaluate your proficiency in both Front-end and Back-end skills, with a specific focus on React. and to give us an idea of how you solve a problem of this kind.

It is essential to thoroughly read the entire document before initiating the design and implementation process. Additionally, please be explicit in stating any assumptions you make.

Our platform is designed to serve and visualize data in various ways. We put emphasis on speedy and efficient queries along with an intuitive user interface.

In this assignment, you are asked to create a web application that visualizes data using two different components: A time-series graph and a table.

Approach this project as you would for a large-scale implementation. Pay meticulous attention to the architectural layers and ensure the code is scalable and of high quality. Feel free to incorporate comments, particularly on edge cases that you might not handle but believe deserve attention.

Use the FS_home_assignment_data as mock data. The data consists of several performance metrics, broken down by categories, over a 6 month time frame. Use this dataset to build and demonstrate the functionality of your web application, making sure to implement the solution in TypeScript for both Front-end and Back-end components.

User Interface

Graph:

- Display aggregated (sum) data as a time-series trend graph. The y-Axis can be any of the metrics present in the csv file (units_sold, revenue, or product_views).
- Recommendation: use [highcharts](#) package
- **Bonus:** Implement checkable chart legends that allow the user to compare up to 2 different metrics at a time

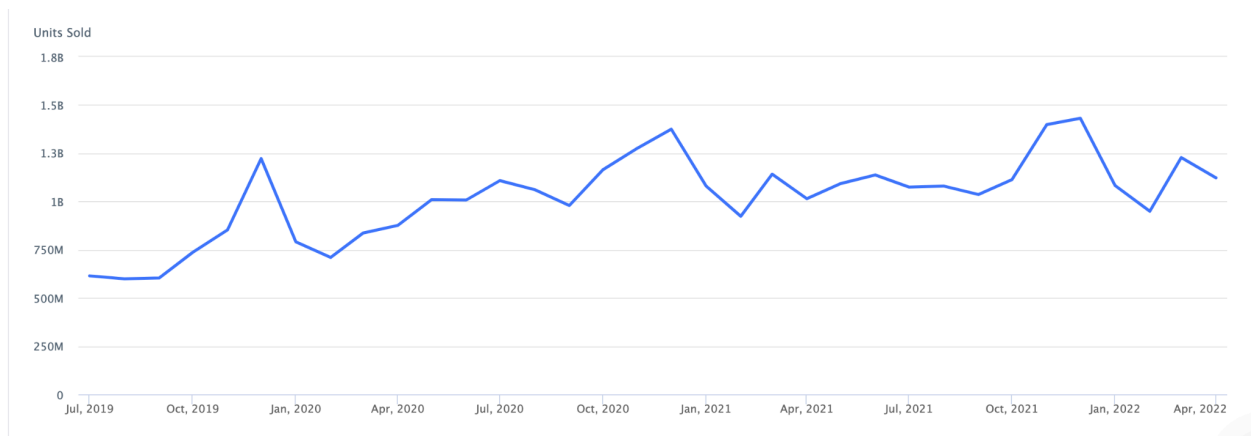


Table:

- Display a table with aggregated (sum) metrics grouped by categories (see basic sample structure below).
- Add a CVR (conversion) column which is calculated using: $cvr = \text{units_sold} / \text{product_views}$
- **Bonus:** Add a change ratio column per category throughout the time frame for a selected metric. For example: if Category X had 80 *units_sold* on 01/01/2021 and 140 *units_sold* on 01/01/2022, the change rate over the time period would be 75%

Category	Product Views	Revenue	Units Sold	CVR
Category A	100	1000	5	5%

Server

- Consider creating a data access layer, specifically an in-memory SQL server such as SQLite.

Bonus

How would you handle a similar web application as above, but at a larger scale (such as 100 million rows)?

- How would you design a query to fetch and serve this data?
- What technologies could be beneficial to store and serve the data?
- What solutions could you implement on the client-side to display a table of this size?

FS_home_assignment_data

category_name	date	product_views	revenue	units_sold
Washers & Dryers	2021-11	341832	13903564	23564
Washers & Dryers	2021-12	309767	7807240	17956
Washers & Dryers	2022-01	401221	10681328	23125
Washers & Dryers	2022-02	337520	9514407	19002
Washers & Dryers	2022-03	401450	12714770	28033
Washers & Dryers	2022-04	391542	12068450	23454
Sports Sunglasses	2021-11	174867	1214005	14312
Sports Sunglasses	2021-12	189659	1272633	16627
Sports Sunglasses	2022-01	203772	752359	10732
Sports Sunglasses	2022-02	217790	1426838	11811
Sports Sunglasses	2022-03	309594	2144667	19960
Sports Sunglasses	2022-04	363165	1509186	19067
Ground Coffee	2021-11	4088720	24046585	1406066
Ground Coffee	2021-12	4932806	24594699	1470766
Ground Coffee	2022-01	4596392	17997054	1164572
Ground Coffee	2022-02	4375323	20559326	1200930
Ground Coffee	2022-03	5290895	24302580	1448151
Ground Coffee	2022-04	4000361	22388047	1280880
Bathroom Sets	2021-11	222882	1035483	12318
Bathroom Sets	2021-12	220332	920361	9117
Bathroom Sets	2022-01	237065	674358	7584

Bathroom Sets	2022-02	266626	622110	7338
Bathroom Sets	2022-03	285245	955452	10761
Bathroom Sets	2022-04	263852	821482	10176