

An online jeans retailer has asked us to build a system to help their purchasing team decide which items to stock based on previous orders. For example, when The Hipster Jeans Company informs the purchasing team of the products it will be selling in spring 2017, the purchasing team has to decide how many of each style, colour and size it should stock and in which countries.

The server will provide an API that serves up the count of previous sales for the last 12 months, grouped by order date, delivery country, manufacturer, gender, colour, size and style.

An example of the data returned from this endpoint is:

OrderDate	DeliveryCountry	Manufacturer	Gender	Size	Colour	Style	Count
01/01/2016	Germany	The Hipster Jeans Company	F	16	Dark Blue	Relaxed	3
01/01/2016	United Kingdom	Denzil Jeans	M	32/32	Light Blue	Skinny	7
02/01/2016	France	The Hipster Jeans Company	M	28/30	Red	Skinny	6
02/01/2016	Austria	Wrangled Jeans	F	12	Yellow	Boot Cut	1

The endpoint will return tens of thousands of results.

We would like you to design and build a UI that allows the purchasing team to explore the summary of orders, helping them to decide which products to buy. The purchasing team would like to know things like:

- The top selling manufacturers by gender and country
  - The top selling sizes by country
  - The top selling months globally and by country
- 
- Please implement the solution as a web-based UI
  - You may use any UI libraries and frameworks
  - The frontend code you submit should be of production quality
  - You may choose any data format and protocol for the client/server API
  - You are not required to implement the server side – it is perfectly acceptable to use static example data or run a simple stub server that serves data from a file
  - It is acceptable to use a simpler protocol than you would use in for this problem in production if it helps you complete the exercise in a shorter period of time
  - Please include a README with the following information:
    - A brief description of the functionality your solution provides to the users
    - How to open your solution in a browser and use it with a sample dataset
    - How to run any tests you have included
    - Reasons/justification for each library or framework you decide to use, if any
    - Reasons for your choice of data format and protocol for the API. If you chose to use a simpler protocol than you would in production, please also explain the protocol you would use for this problem in production
    - Assumptions you made, if any, due to unclear requirements

Please zip your entire solution and email it to your agent, who will pass it on to us