

***Answer For Ex-02:***

***How do sub-resource routes like /journalists/:id/articles help in designing a clear and organized API?***

*Explain the benefits of using nested routes for resource relationships in REST APIs.*

Sub-resource routes like /journalists/:id/articles make it super clear that you're asking for articles by a specific journalist. Helps organize the API so it feels more logical and grouped.

***What challenges did you face when managing multiple filter states (journalist and category) in React?***

*Discuss how you handled state updates and conditional rendering when multiple inputs affect the same output.*

Managing multiple filters got messy—had to track more state, avoid re-fetching too much, and make sure filters don't clash or reset each other by mistake.

***What would be the advantages and disadvantages of handling the filtering entirely on the frontend versus using API-based filtering?***

*Compare client-side filtering (in-memory) vs. server-side filtering (via query or nested routes).*

I used separate state for each filter, then updated the article list only when filters changed. Used simple checks to avoid rendering stuff that didn't match.

***If you needed to allow filtering by both journalist and category at the same time on the backend, how would you modify the API structure?***

*Think about adding query parameters or designing a new combined route.*

Frontend filtering is faster and no server calls, but bad if you have a ton of data. Backend filtering is cleaner for large datasets but takes more setup and makes the frontend depend on the API more.

For backend filtering by both, I'd use query params like  
/articles?journalist=2&category=1—easier to combine data and  
scale it later.

***How did this exercise help you understand the interaction between React state, form controls, and RESTful API data?***

*Reflect on how state changes triggered data fetching and influenced the UI rendering logic.*

This made me get how React state ties into form inputs, and how even small changes can totally affect what shows up. You really see how the UI reacts to API data and state together.