A: Search-Focused Assignment

Time: 2–3 hours

Assignment: Inventory Search API + UI

Context (Candidate-facing)

Zeerostock allows buyers to search surplus inventory across multiple suppliers.
Build a **simple inventory search feature**.

Requirements

Backend

Create an API:

`GET /search`

Supports query params:

- `q` → product name (partial match)
- `category`
- `minPrice`
- `maxPrice`

Rules:

- Case-insensitive search
- Multiple filters can be combined
- If no filters → return all results

Data source:

- Static JSON file **OR**
- In-memory array (10–15 records)

Frontend

- Search input box
- Category dropdown
- Price range inputs
- Display results in a list/table
- Show “No results found” state

Edge Cases to Handle

- Empty search query
- Invalid price range
- No matches

README (Short)

- Search logic explanation
- One performance improvement you’d make for large datasets

B: Database-Focused Assignment

Time: 2–3 hours

Assignment: Inventory Database + APIs

Context (Candidate-facing)

Suppliers list surplus stock which buyers can view.

Requirements

Database

Create **2 tables / collections**:

Suppliers

- id
- name
- city

Inventory

- id
- supplier_id
- product_name
- quantity
- price

Relationship:

- One supplier → many inventory items

Backend APIs

`POST /supplier`
`POST /inventory`
`GET /inventory`

Rules:

- Inventory must belong to a valid supplier
- Quantity ≥ 0
- Price > 0

One Required Query

Return:

> All inventory grouped by supplier, sorted by total inventory value (`quantity × price`)

README

- Database schema explanation
- Why you chose SQL / NoSQL
- One indexing or optimization suggestion