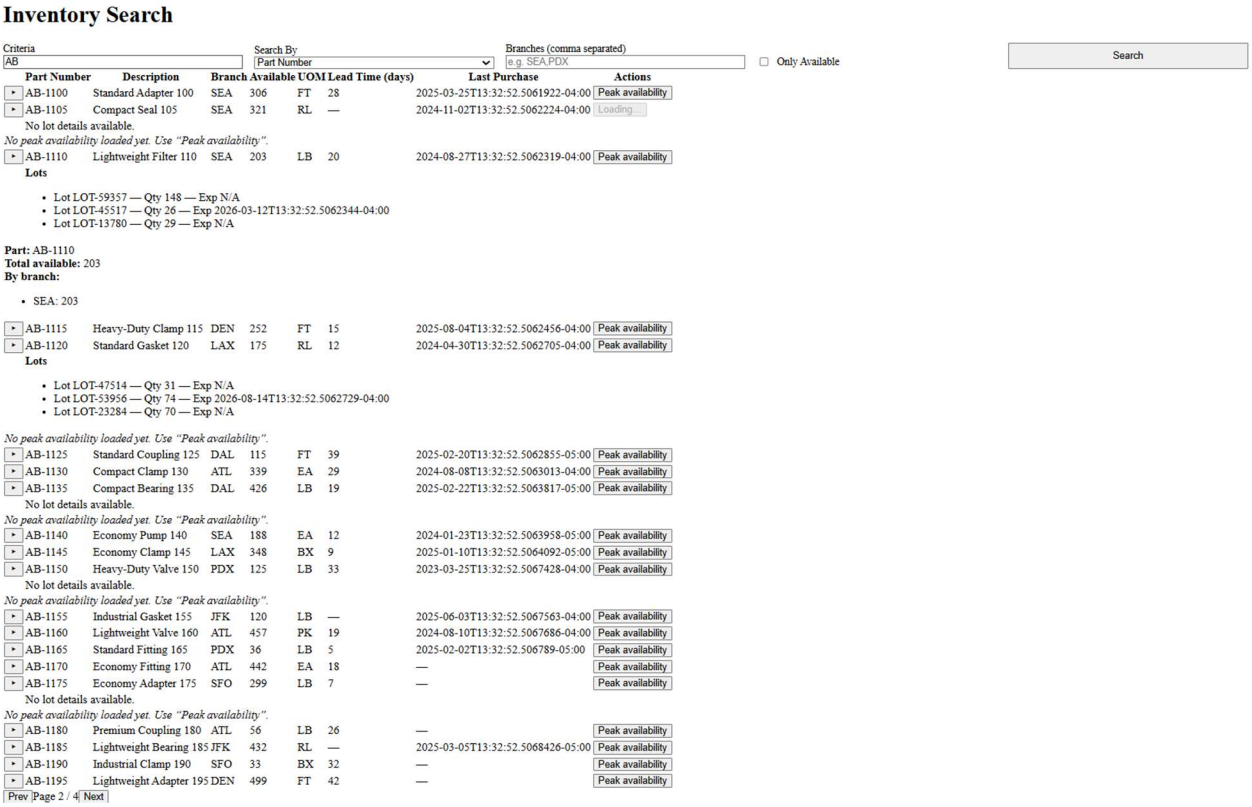
Inventory Search – Code Test

Goal :

Generate an Angular App utilizing a C# .NET backend to create a cleaner looking version of this.



Inventory Search

Criteria: AB | Search By: Part Number | Branches (comma separated): e.g. SEA,PDX | Only Available | Search

Part Number	Description	Branch	Available	UOM	Lead Time (days)	Last Purchase	Actions
AB-1100	Standard Adapter 100	SEA	306	FT	28	2025-03-25T13:32:52.5061922-04:00	Peak availability
AB-1105	Compact Seal 105	SEA	321	RL	—	2024-11-02T13:32:52.5062224-04:00	Loading…
No lot details available.							
No peak availability loaded yet. Use "Peak availability".							
AB-1110	Lightweight Filter 110	SEA	203	LB	20	2024-08-27T13:32:52.5062319-04:00	Peak availability
Lots							
• Lot LOT-59357 — Qty 148 — Exp N/A							
• Lot LOT-45517 — Qty 26 — Exp 2026-03-12T13:32:52.5062344-04:00							
• Lot LOT-13780 — Qty 29 — Exp N/A							
Part: AB-1110							
Total available: 203							
By branch:							
• SEA: 203							
AB-1115	Heavy-Duty Clamp 115	DEN	252	FT	15	2025-08-04T13:32:52.5062456-04:00	Peak availability
AB-1120	Standard Gasket 120	LAX	175	RL	12	2024-04-30T13:32:52.5062705-04:00	Peak availability
Lots							
• Lot LOT-47514 — Qty 31 — Exp N/A							
• Lot LOT-53956 — Qty 74 — Exp 2026-08-14T13:32:52.5062729-04:00							
• Lot LOT-23284 — Qty 70 — Exp N/A							
No peak availability loaded yet. Use "Peak availability".							
AB-1125	Standard Coupling 125	DAL	115	FT	39	2025-02-20T13:32:52.5062855-05:00	Peak availability
AB-1130	Compact Clamp 130	ATL	339	EA	29	2024-08-08T13:32:52.5063013-04:00	Peak availability
AB-1135	Compact Bearing 135	DAL	426	LB	19	2025-02-22T13:32:52.5063817-05:00	Peak availability
No lot details available.							
No peak availability loaded yet. Use "Peak availability".							
AB-1140	Economy Pump 140	SEA	188	EA	12	2024-01-23T13:32:52.5063958-05:00	Peak availability
AB-1145	Economy Clamp 145	LAX	348	BX	9	2025-01-10T13:32:52.5064092-05:00	Peak availability
AB-1150	Heavy-Duty Valve 150	PDX	125	LB	33	2023-03-25T13:32:52.5067428-04:00	Peak availability
No lot details available.							
No peak availability loaded yet. Use "Peak availability".							
AB-1155	Industrial Gasket 155	JFK	120	LB	—	2025-06-03T13:32:52.5067563-04:00	Peak availability
AB-1160	Lightweight Valve 160	ATL	457	PK	19	2024-08-10T13:32:52.5067686-04:00	Peak availability
AB-1165	Standard Fitting 165	PDX	36	LB	5	2025-02-02T13:32:52.506789-05:00	Peak availability
AB-1170	Economy Fitting 170	ATL	442	EA	18	—	Peak availability
AB-1175	Economy Adapter 175	SFO	299	LB	7	—	Peak availability
No lot details available.							
No peak availability loaded yet. Use "Peak availability".							
AB-1180	Premium Coupling 180	ATL	56	LB	26	—	Peak availability
AB-1185	Lightweight Bearing 185	JFK	432	RL	—	2025-03-05T13:32:52.5068426-05:00	Peak availability
AB-1190	Industrial Clamp 190	SFO	33	BX	32	—	Peak availability
AB-1195	Lightweight Adapter 195	DEN	499	FT	42	—	Peak availability

Prev | Page 2 / 4 | Next

Code Test Brief

Basic outlines are given for both to help as a starting point

Looking to get a well designed solution.

Looking to see what your design choices are for the UI with regards to look and feel. The example given above is an ugly UI looking for something much cleaner with a professional look to it.

You are allowed to use any tools to help you complete this task expectation is to be completely open about what you used and why and how you used it.

Requirements

- Use Angular 18 or higher required
    - Use HttpClient for API calls
    - Strong typing via TypeScript interfaces
    - Reactive forms
    - We use Tailwind in house.
- Basic Structure to the code is supplied in the repository with lots of hint comments in the code
- Behavior:
    - Show a loading indicator while searching
    - Cancel in-flight requests if a new search is fired
    - Display errors appropriately
    - Cache the most recent 5 unique searches for 60 seconds (avoid duplicate API calls within TTL)
- Acceptance criteria:
    - Functional form with validation, loading state, cancel-on-new-search
    - Results with sorting, pagination, and expandable rows
    - Caching prevents duplicate API calls within 60 seconds for last 5 distinct queries
- Deliverables:
    - Please upload deliverables to a Git repository and supply link to that repo.
    - Source code for the feature (Angular TypeScript, HTML, SCSS) so that it can be built then run against the mock inventory server
    - C# files required to build the Server with Service layer unit tests for the C# project
    - Quick start instructions (README)

What is provided

Inventory-mock-api: This is the server supplied to allow you to test your angular code. Note this code along with the json returned should be sufficient allow you to complete the C# (Note there may or may not be some missing fields to the supplied C# models)

Inventory-Search: This contains a starting point for the angular code

InventoryServer This contains a starting point for the C# .NET project. ***The repository layer is an area where you can get creative, if you so desire, as to how you want to generate the data***. Note that in the Inventory-mock-api code the data generation is randomized based on a few fixed constants, you can follow that mechanism if you want or you can do something completely different that better demonstrates your knowledge of C#.

Conclusion:

Please be creative what I want to see is well structured easily readable, understandable code.

You are free to modify the API if you deem it appropriate. If doing so please give reasons as to why this is a better solution.

There is no limit on how you choose to develop the solution, I would like a reasoning as to the why and how for using any tools. As software developers our day to day tasks often require the use of these tools this code test should be treated no differently.