Munro Library Challenge

Provided in this archive is a CSV file containing information about munros and munro tops within Scotland. The goal of your solution is to create a simple API which other software can use to sort and filter the munro data. Your solution should use C#, ASP.NET and any relevant libraries you believe would be useful.

The API should provide the following functionality:
- Filtering of search by hill category (i.e. Munro, Munro Top or either). If this information is not provided by the user it should default to either. This should use the "post 1997" column and if it is blank the hill should be always excluded from results.
- The ability to sort the results by height in meters and alphabetically by name. For both options it should be possibly to specify if this should be done in ascending or descending order.
- The ability to limit the total number of results returned, e.g. only show the top 10
- The ability to specify a minimum height in meters
- The ability to specify a maximum height in meters
- Queries may include any combination of the above features and none are mandatory.
- Suitable error responses for invalid queries (e.g. when the max height is less than the minimum height)

Optionally you may choose to include the following feature if you can think of a good approach but it is not required to complete the solution. Correctness and structure of the rest of your solution is more important than adding this extra objective:
- The ability to combine sort criteria in order of preference. For example: sort by height descending and then alphabetically by name as a secondary criteria (for when the height is the same)

The query results should be returned as a list of items using JSON. Each item should contain the name, height in meters, hill category and grid reference (e.g. NN773308). Other fields should not be included.

[NOTES ON NEXT PAGE]

Notes:

- To limit the time required to build your solution there is no need to use a database and it is acceptable for the parsed Munro data to be held in memory.
- To load the CSV data into your system you can either provide an endpoint for the user to upload a file or load the CSV from a local file directly when the API starts.
- There is no need to add authentication or rate limiting to endpoints developed.
- Please include any testing code you write in your submitted solution.
- Whilst developing your solution please commit your work into a git repository as you go. This is not to see how much time you take or at what times you worked on the solution but is so that we can evaluate how you broke down and approached the problem.