Data Engineering Technical Test

It is important for us to get a sense of how you approach problems, and how you produce deployable code. To that end, we have a practical stage in the hiring process, which is to ask you to develop and share with us a simple data structure, search query and API to ingest data and expose the search query. Below are the instructions:

Dataset Overview

- The source data for this exercise is a set of json files, each representing the content and metadata of **Parts** of a piece of **Legislation** . Each file also contains (and therefore repeats) the metadata of the ***Legislation*** it belongs to.
- A ***Legislation*** entity represents a real world piece of legislation. As an example see the FCA Handbook (https://www.handbook.fca.org.uk/handbook/).
- A book consists of ***Parts*** which represent the sections, chapters, paragraphs, and clauses of the legislation. As such ***Parts*** are organised in a hierarchy, since a chapter may have sub chapters, sub chapters may have clauses etc.
- Legislation is versioned as the ***LegislationVersion*** *entity*. All versions of legislation have the same *LegislationSourceId*, but each version of the ***Legislation*** gets a new *LegislationVersionOrdinal*. These two properties make up the composite key of a ***LegislationVersion***. There is also a *LegislationVersionId* which is an internal identifier for the ***Legislationversion***.
- ***Parts*** are also versioned, following the same identification structure as *Legislation (i.e. a SourceId* and *VersionOrdinal)*. Note that ***Part*** versioning is semi-independent of ***Legislation*** versioning - for example a ***Part*** may not change on the first two versions of ***Legislation***, but change on the third version, but it cannot change version between ***Legislation*** versions.

Instructions

1. Create an API to import the provided json files into an optimal MS SQL Server schema. The json files are in the legislation.zip file.
2. Downstream for the application using these data must be clean (HTML removed) and decoded format.
3. Create a stored procedure that can search return a list of ***Legislation*** that meet a given set of search criteria. At a minimum this must allow for text search across ***Legislation*** and ***Part*** title and content.
4. Optionally, extend the API to import the provided data into a NoSQL database and provide the ability to search as per 3 above.

What to provide

1. A SQL script that creates the database and schema
2. Data Import and Search API
3. If you undertake the task using a NoSQL database, provide 1 and 2 above for NoSQL also.
4. Provide access to a code repository with a ReadMe to allow us to review the code, build and run the API.

You can use .NET or Python for the API - but do focus on simple, robust code that would work at scale.