

## Technical task

Create simple Web API application which provides information about different assets available on market. The application should read information about assets from file.

Assets are defined by **name** and **ID**. Each asset is additionally described by predefined list of properties which can be true or false:

1. is fix income
2. is convertible
3. is swap
4. is cash
5. is future

It should be assumed that in database there are already existing assets (only properties and timestamps can change). The database schema should be proposed by candidate based on what should be done in the task.

Preferred technology stack:

- .NET Core 3.1, C#
- Web API
- SQL Server
- Entity Framework Core 3.1 code first approach

Application should expose few endpoints:

1. **Request that triggers file processing.**

This action doesn't have any arguments, it simply indicates that CSV file should be taken from disk (any location) and processed (send to DB). Example file structure:

AssetId,Properties,Value,Timestamp

1,"is cash","true"," 2020-07-01T16:32:32"

3,"is convertible","false"," 2020-08-02T16:32:32"

8,...

Only records for which timestamp is greater than from last update should be updated. If we didn't read any information from file about asset yet, default value for property is false and default timestamp equals to DateTime.Min.

It is possible that in the file we will have duplicate values for property, the same asset ID for two or more records. In this case a record with a newer timestamp should be sent to DB.

If asset from file cannot be found in DB it has to be logged (anyhow can be even console).

### OPTIONAL

Information from the file should be stored in the DB in batches. For example if the file contains 9900 lines and the batch size (configurable) is set to 1000 then 10 batches should be created. The batches can be sent to the DB in parallel.

## 2. **Get asset IDs for property set to specific value**

Example request

```
{
  "property": "is cash",
  "value": "true"
}
```

This should return list of all asset IDs for which "is cash" property is set to true.

## 3. **Set property for asset**

Example request

```
{
  "AssetId": 1,
  "Property": "is cash",
  "Value": "true",
  "Timestamp": "2020-07-01T16:32:32"
}
```

This is additional way to set asset property. Timestamp should be checked before update the same way as in case of file so the asset should be updated only if timestamp in the request is greater than the one from in the DB.

**OPTIONAL:** Create Dockerfile for web application exposing the endpoints

Additional notes:

- We would like to discuss the solution during interview if it is possible (screen sharing)
- If candidate didn't work with .NET Core and it will take too much time to get familiar with it of course .NET Framework can also be used
- Solution can be shared however it is preferred (archive, GitHub etc.)