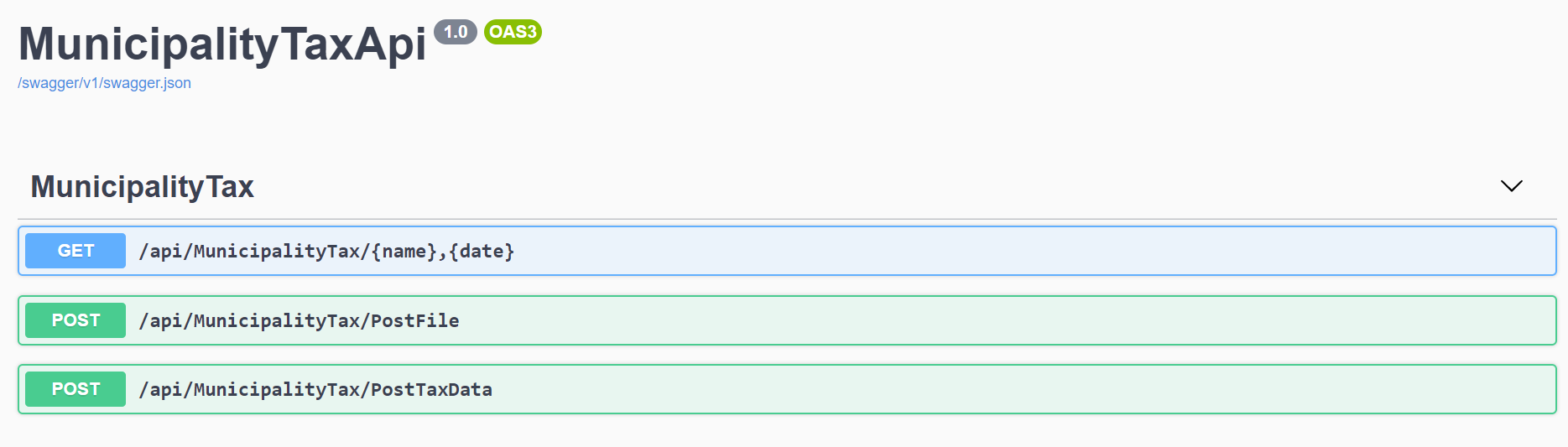
**Municipality tax Application**

It is a small application which manages taxes applied in different municipalities.

Application has no User Interface as it works as API exposing the endpoints to any other consumer application to use.

* It has its own database(MunicipalityTaxStore) where municipality taxes are stored in table MunicipalitytaxDetail.
* POST API also takes input as excel file to import the municipalities data from file.
* POST API have ability to insert new records for municipality taxes.
* GET API returns the specific municipality tax by entering municipality name and date.
* Swagger support is provided to test the API easily.



**Example:** Municipality Copenhagen has its taxes scheduled like this :

- yearly tax = 0.2 (for period 2016.01.01-2016.12.31),

- monthly tax = 0.4 (for period 2016.05.01-2016.05.31),

- it has no weekly taxes scheduled,

- and it has two daily taxes scheduled = 0.1 (at days 2016.01.01 and 2016.12.25).

The result according to provided example would be:

|  |  |  |
| --- | --- | --- |
| Municipality (Input) | Date (Input) | Result |
| Copenhagen | 2016.01.01 | 0.1 |
| Copenhagen | 2016.05.02 | 0.4 |
| Copenhagen | 2016.07.10 | 0.2 |
| Copenhagen | 2016.03.16 | 0.2 |

**Database scripts to run :-**

CREATE DATABASE MunicipalityTaxStore

GO

USE MunicipalityTaxStore

GO

CREATE TABLE MunicipalityTaxDetail

(

Id BIGINT NOT NULL,

Name NVARCHAR(50) NOT NULL,

MunicipalityId BIGINT NOT NULL,

Frequency NVARCHAR(50) NULL,

StartDate DATE NULL,

EndDate DATE NULL,

Tax FLOAT NULL

PRIMARY KEY (Id)

)

GO

INSERT INTO MunicipalityTaxDetail

VALUES

(1,'Copenhagen',0001,'yearly','1/1/2016','12/31/2016',0.2),

(2,'Copenhagen',0001,'monthly','5/1/2016','5/31/2016',0.4),

(3,'Copenhagen',0001,'daily','1/1/2016','1/1/2016',0.1),

(4,'Copenhagen',0001,'daily','12/31/2016','12/31/2016',0.1)

**Note:**

**In case there is any issue related to packages then install the below packages from**

**Tools –> NuGet Package Manager –> Package Manager Console**

Install-Package Microsoft.EntityFrameworkCore.Tools

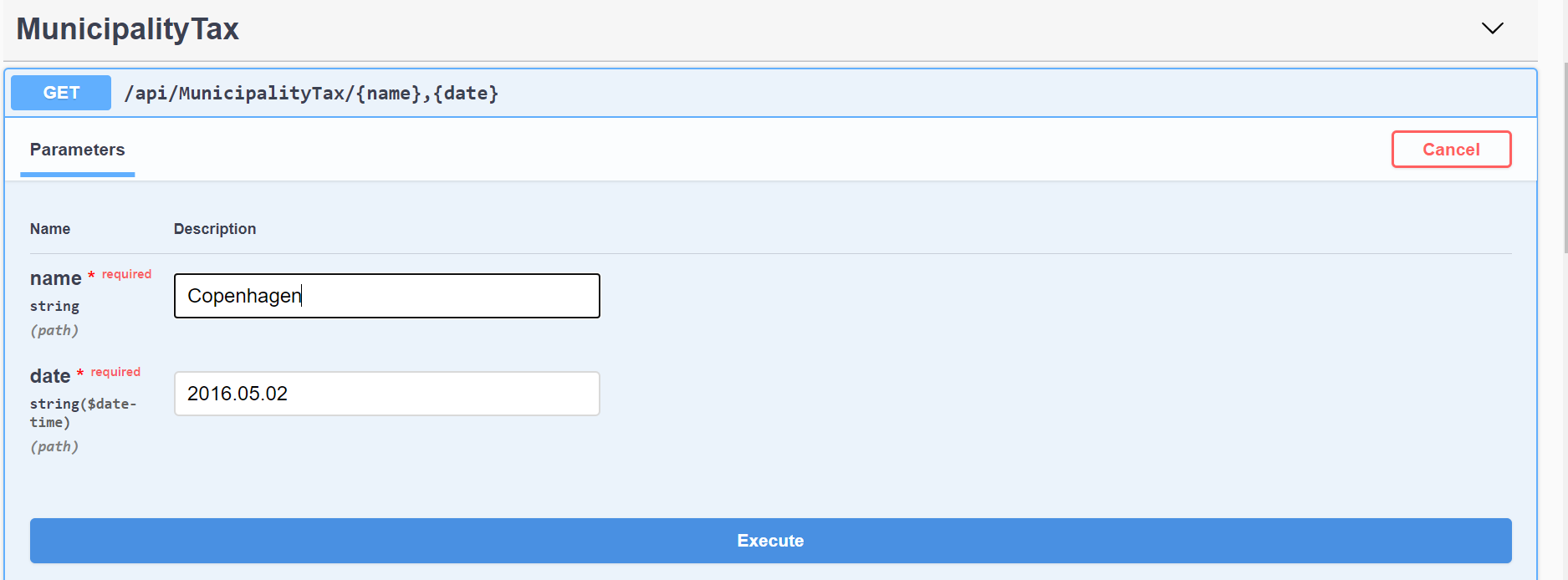
Install-Package Microsoft.EntityFrameworkCore.SqlServer

*Install-Package Swashbuckle.AspNetCore*

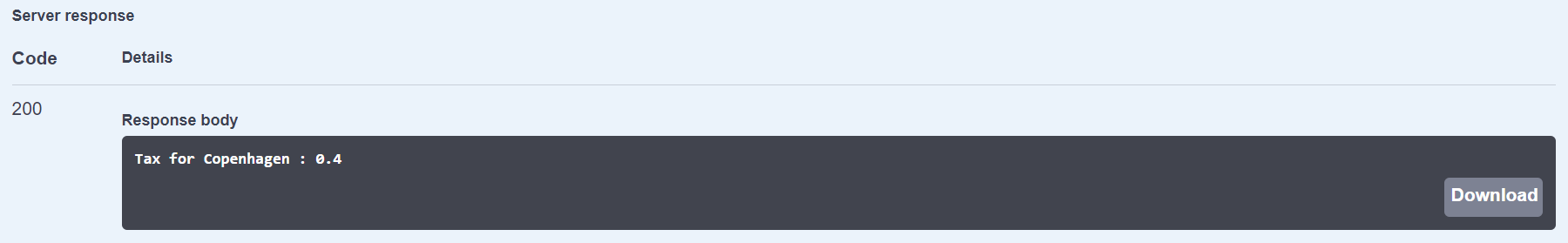
Install-Package Swashbuckle.AspNetCore.Newtonsoft

**API EndPoints :-**

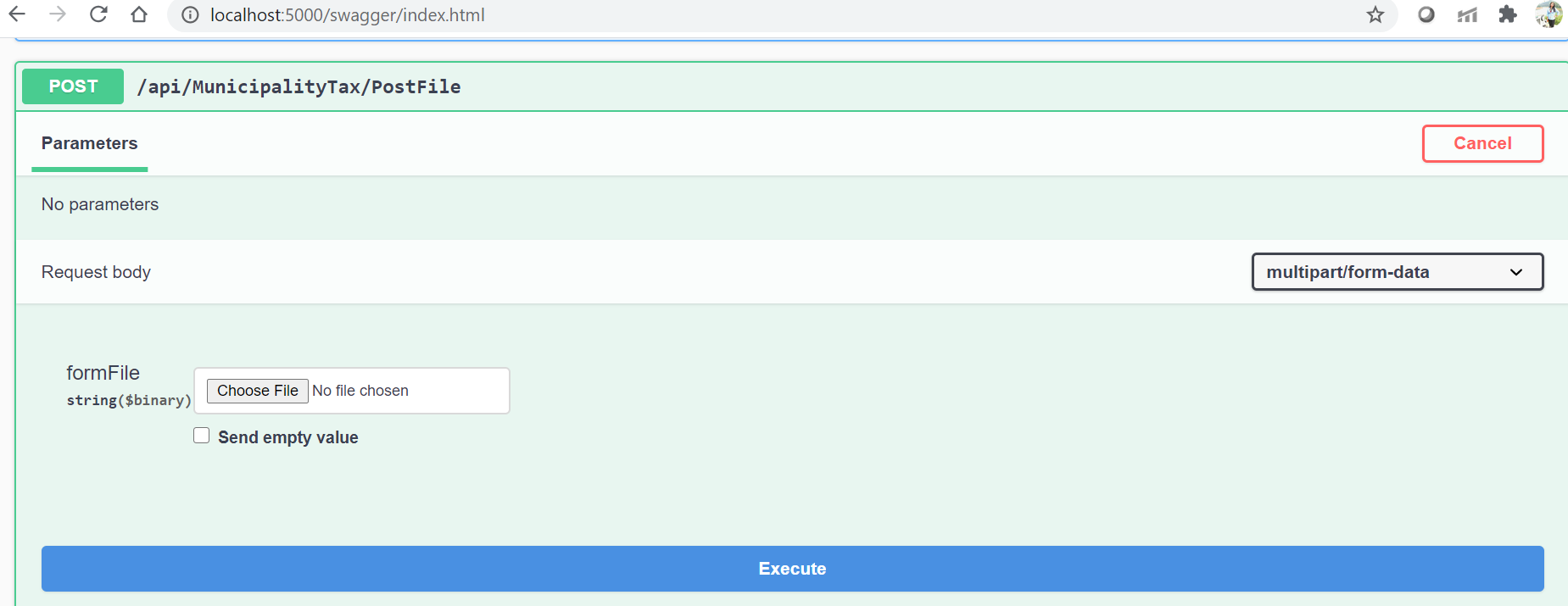
1. Get API



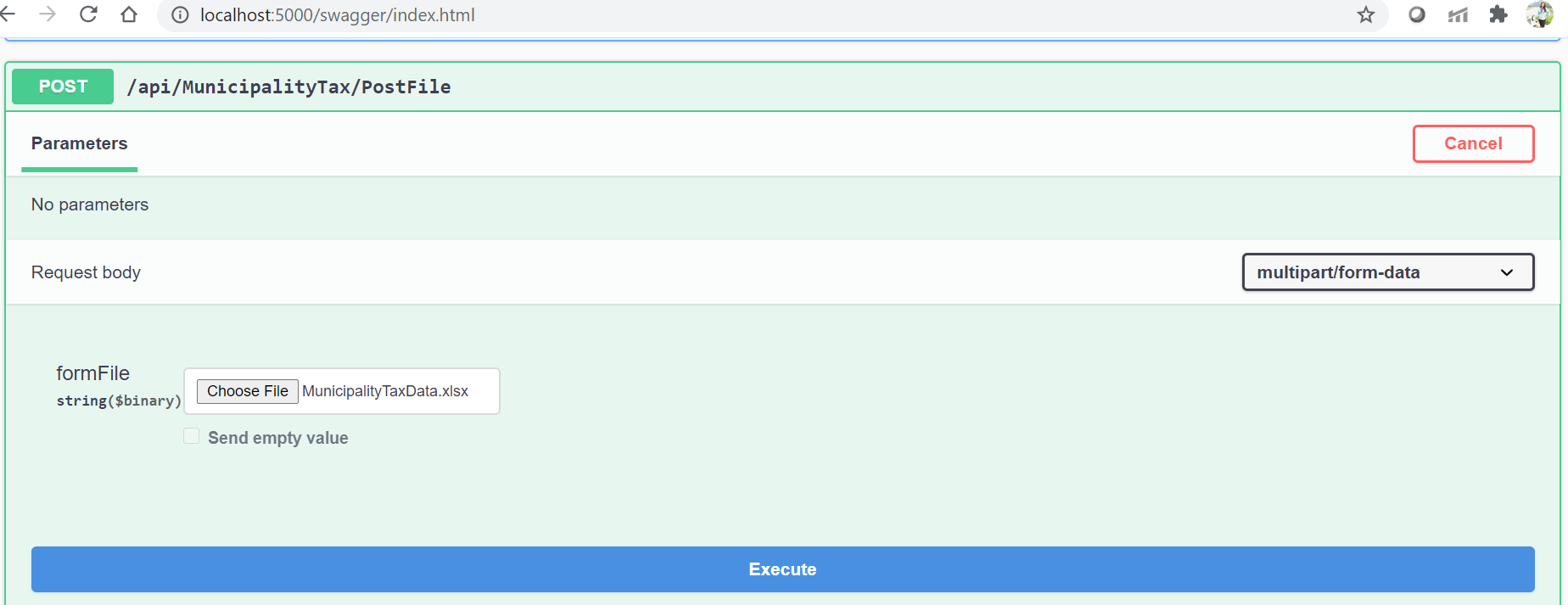
Result Window –



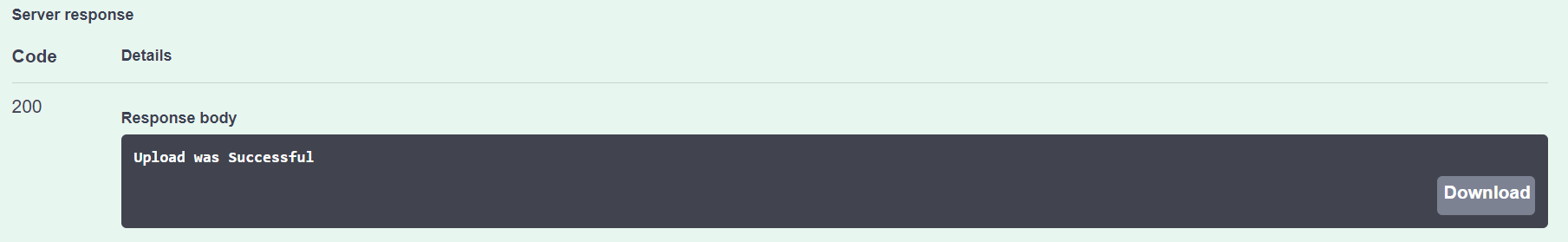
1. POST file API



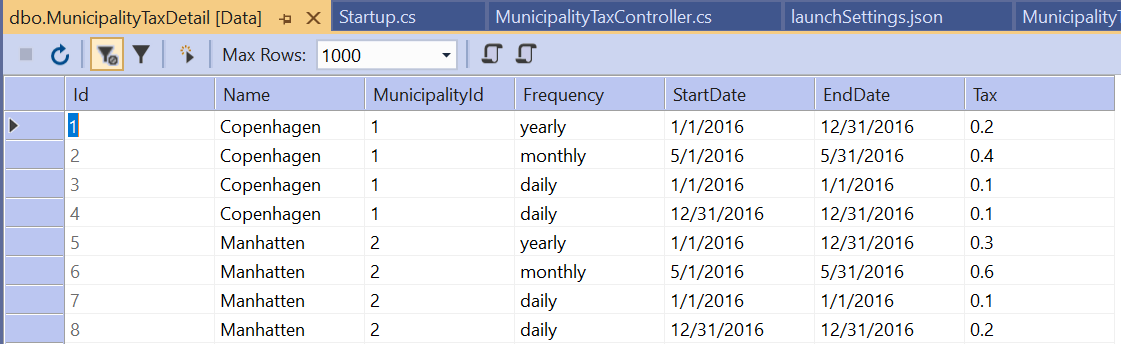
Select an excel file to upload –



Result window –



In Database –



1. Post a tax data

Sample Input :

{

"id": 0,

"name": "Manhatten",

"municipalityId": 0002,

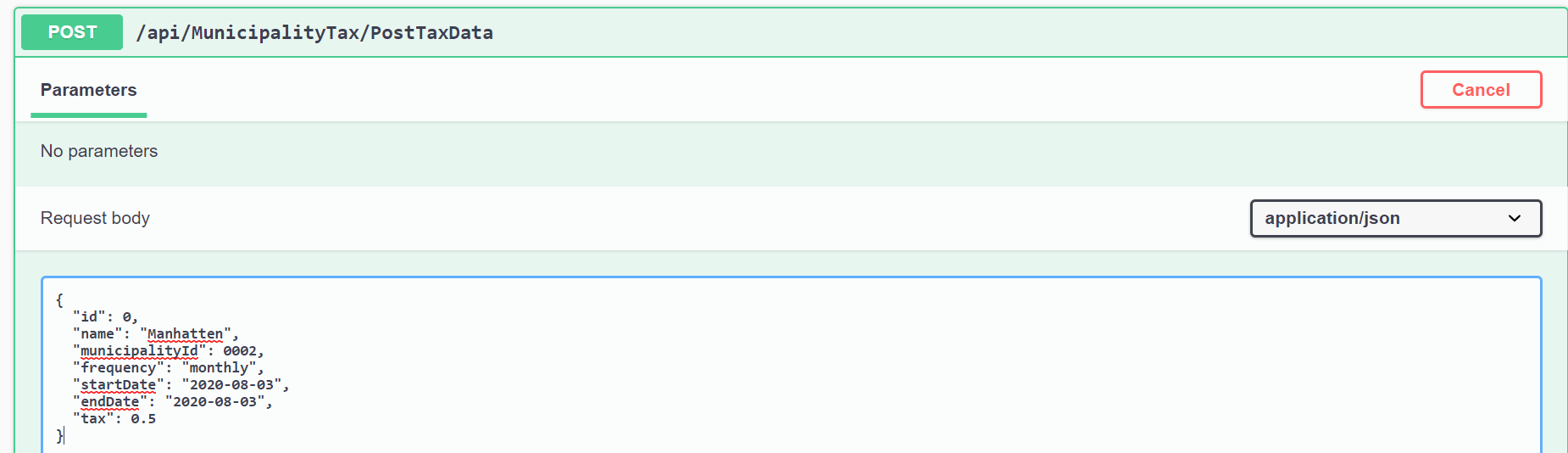
"frequency": "monthly",

"startDate": "2020-08-03",

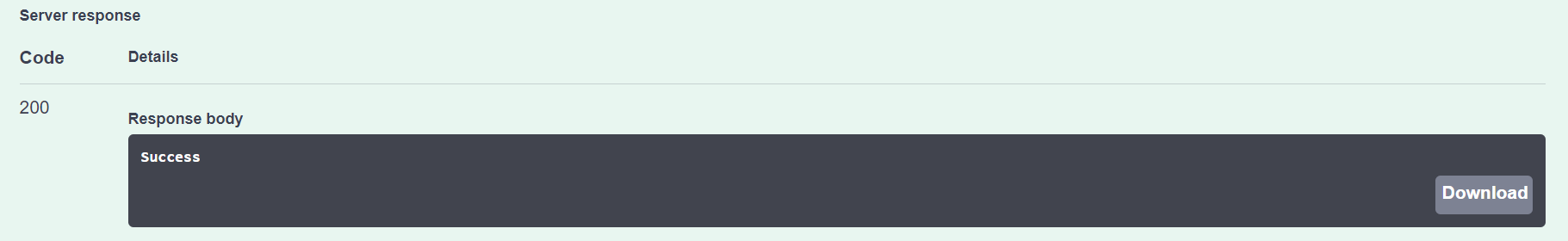
"endDate": "2020-08-03",

"tax": 0.5

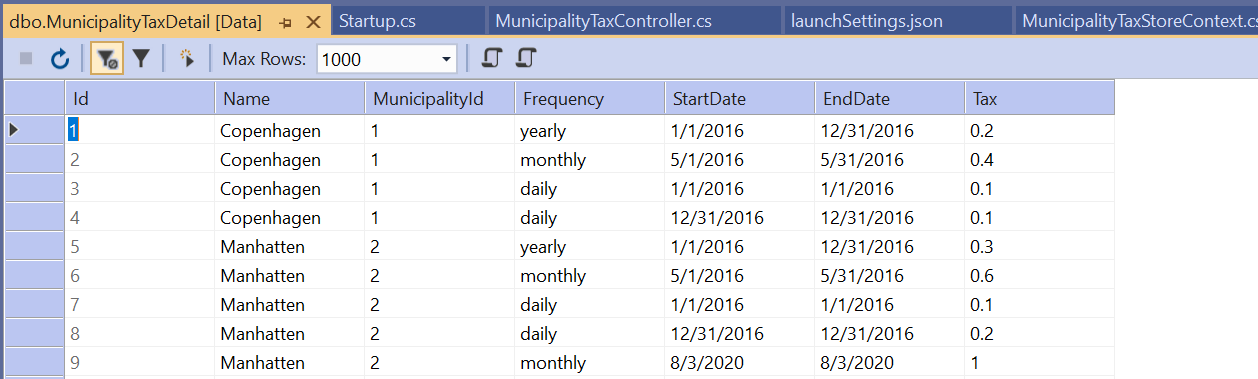
}



Result Window -



In Database –



**In Future Tasks –**

* A consumer application can be built to consume the above API end points (producer application).
* Scheduled data can also be saved into different tables assigning a job id to be picked by a job.
* Application is deployed as a self-hosted windows service