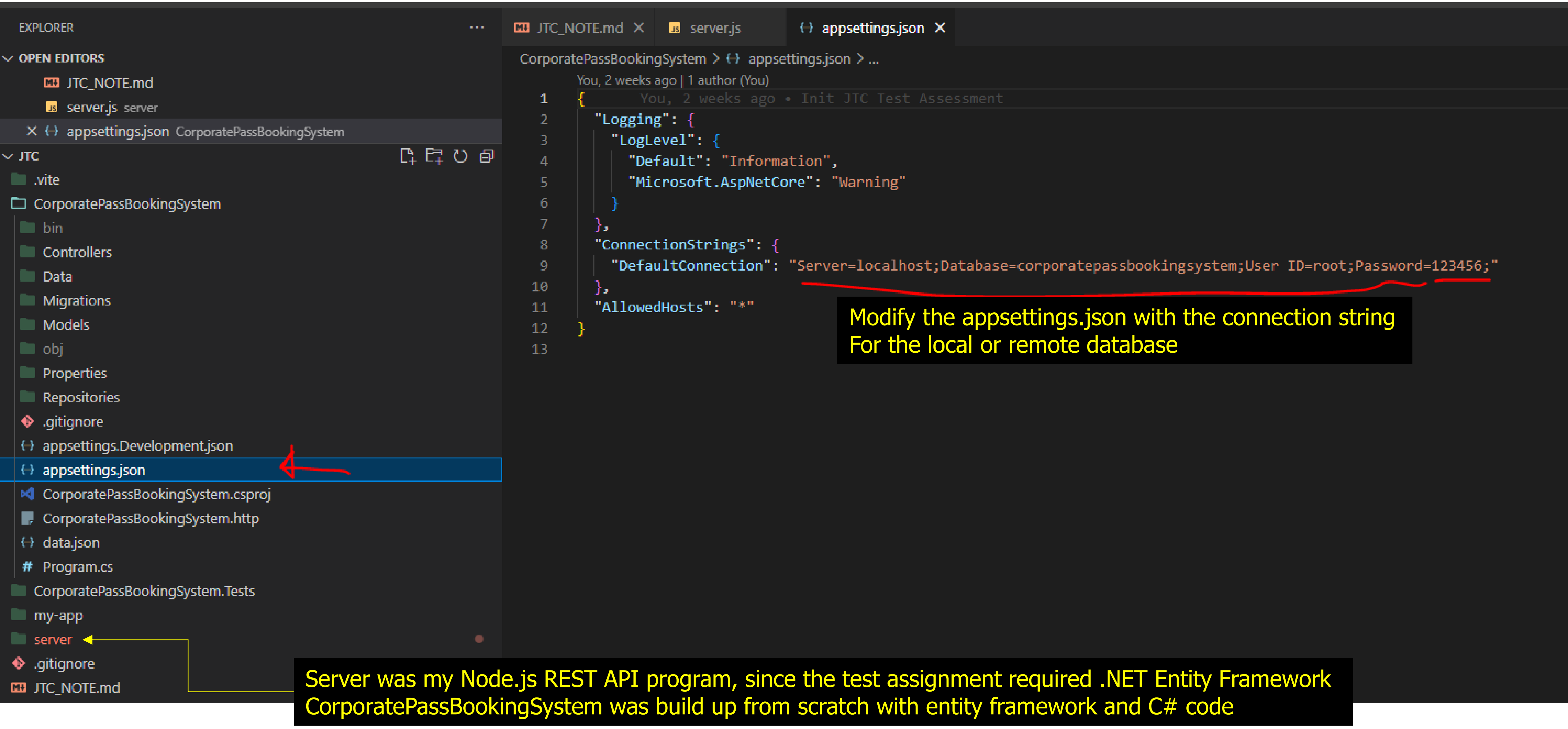
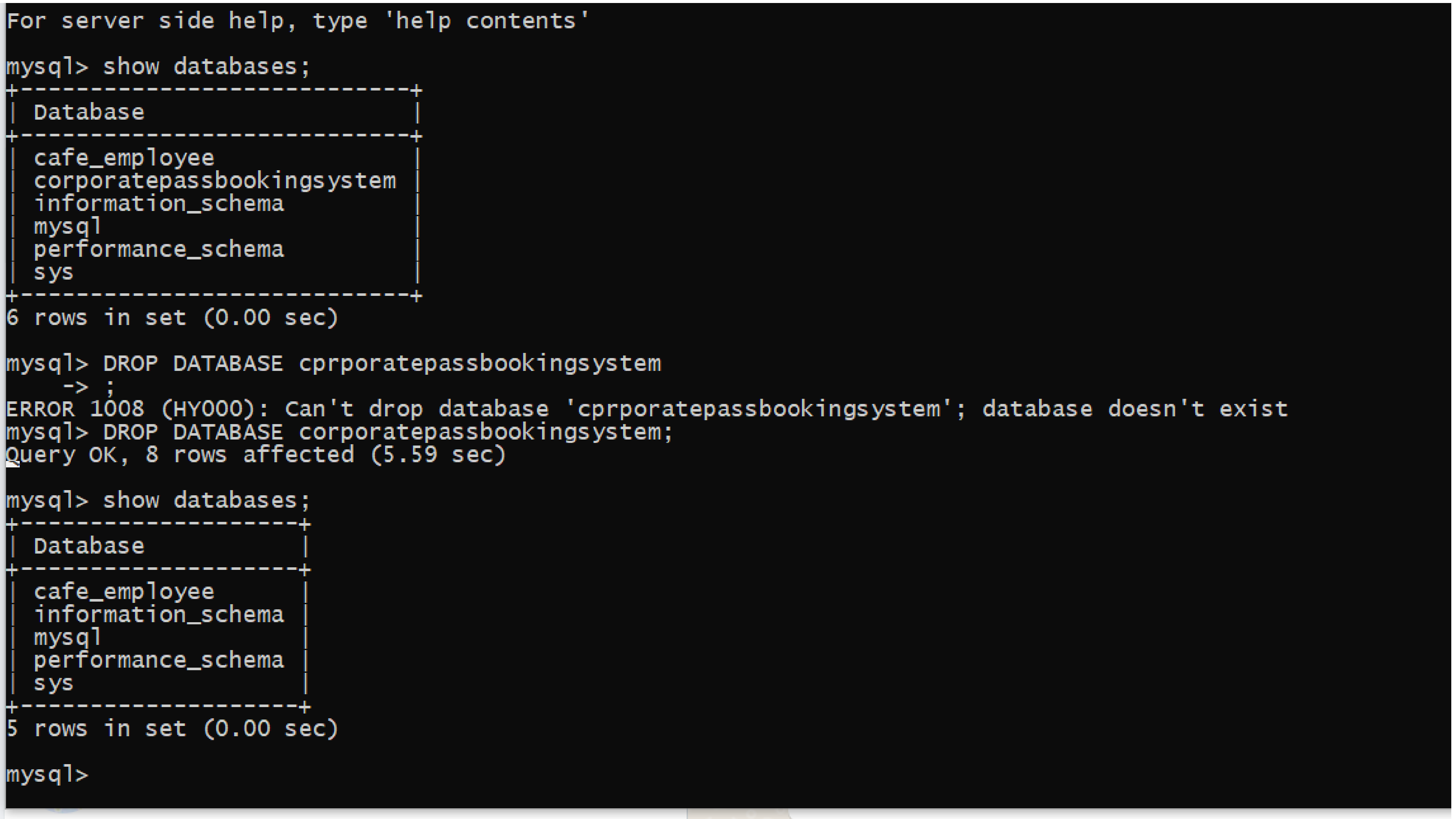


EXPLORER
OPEN EDITORS
JTC_NOTE.md
server.js server
appsettings.json CorporatePassBookingSystem
JTC
.vite
CorporatePassBookingSystem
bin
Controllers
Data
Migrations
Models
obj
Properties
Repositories
.gitignore
appsettings.Development.json
appsettings.json
CorporatePassBookingSystem.csproj
CorporatePassBookingSystem.http
data.json
Program.cs
CorporatePassBookingSystem.Tests
my-app
server
.gitignore
JTC_NOTE.md
JTC_NOTE.md
server.js
appsettings.json
CorporatePassBookingSystem > appsettings.json > ...
You, 2 weeks ago | 1 author (You)

```
{       You, 2 weeks ago • Init JTC Test Assessment
  "Logging": {
    "LogLevel": {
      "Default": "Information",
      "Microsoft.AspNetCore": "Warning"
    }
  },
  "ConnectionStrings": {
    "DefaultConnection": "Server=localhost;Database=corporatepassbookingsystem;User ID=root;Password=123456;"
  },
  "AllowedHosts": "*"
}
```

Modify the appsettings.json with the connection string
For the local or remote database

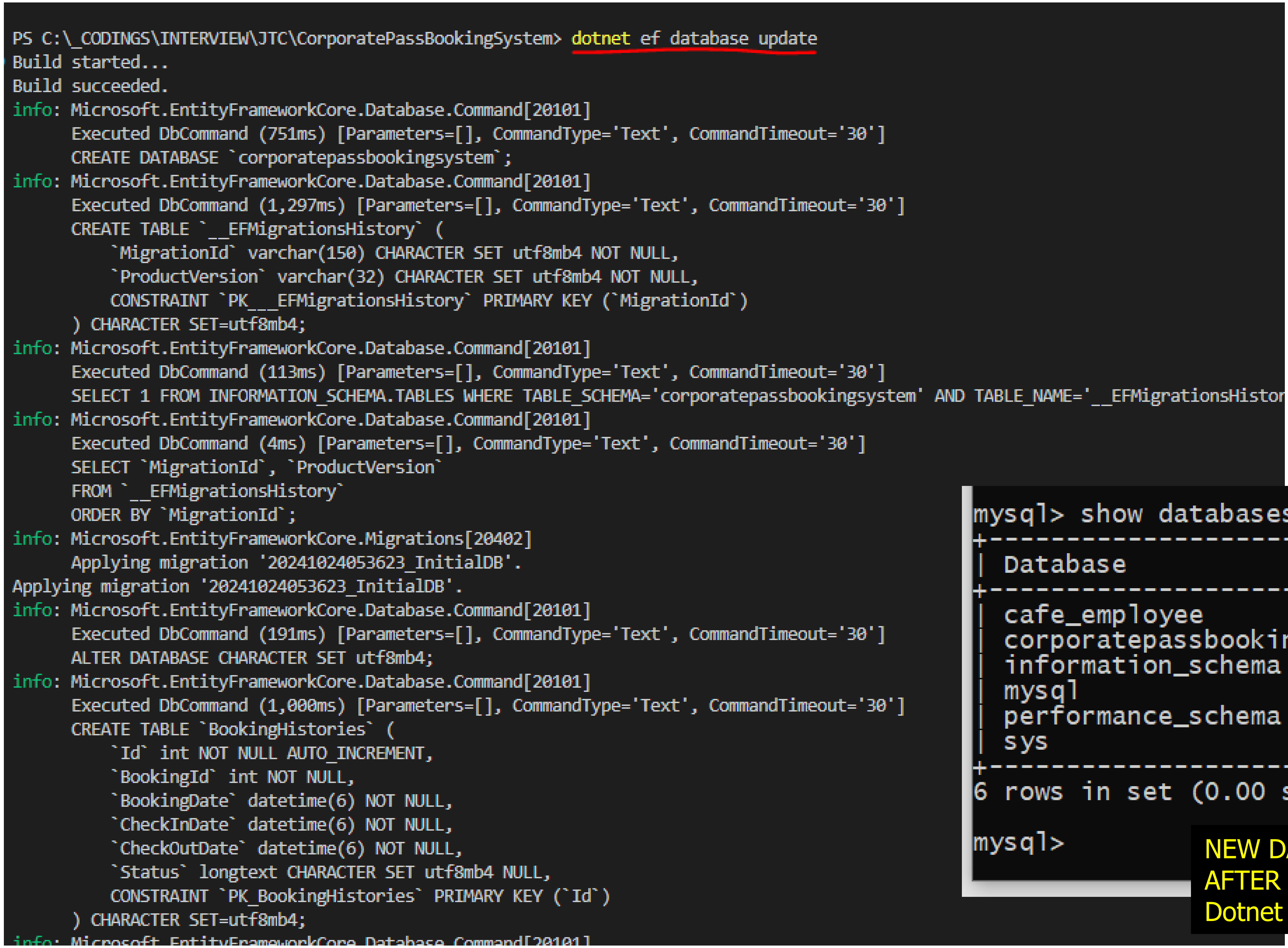
Server was my Node.js REST API program, since the test assignment required .NET Entity Framework
CorporatePassBookingSystem was build up from scratch with entity framework and C# code


```
For server side help, type 'help contents'

mysql> show databases;
+---------------------------+
| Database                  |
+---------------------------+
| cafe_employee             |
| corporatepassbookingsystem |
| information_schema        |
| mysql                     |
| performance_schema        |
| sys                       |
+---------------------------+
6 rows in set (0.00 sec)

mysql> DROP DATABASE cprporatepassbookingsystem
    -> ;
ERROR 1008 (HY000): Can't drop database 'cprporatepassbookingsystem'; database doesn't exist
mysql> DROP DATABASE corporatepassbookingsystem;
Query OK, 8 rows affected (5.59 sec)

mysql> show databases;
+--------------------+
| Database           |
+--------------------+
| cafe_employee      |
| information_schema |
| mysql              |
| performance_schema |
| sys                |
+--------------------+
5 rows in set (0.00 sec)

mysql>
```

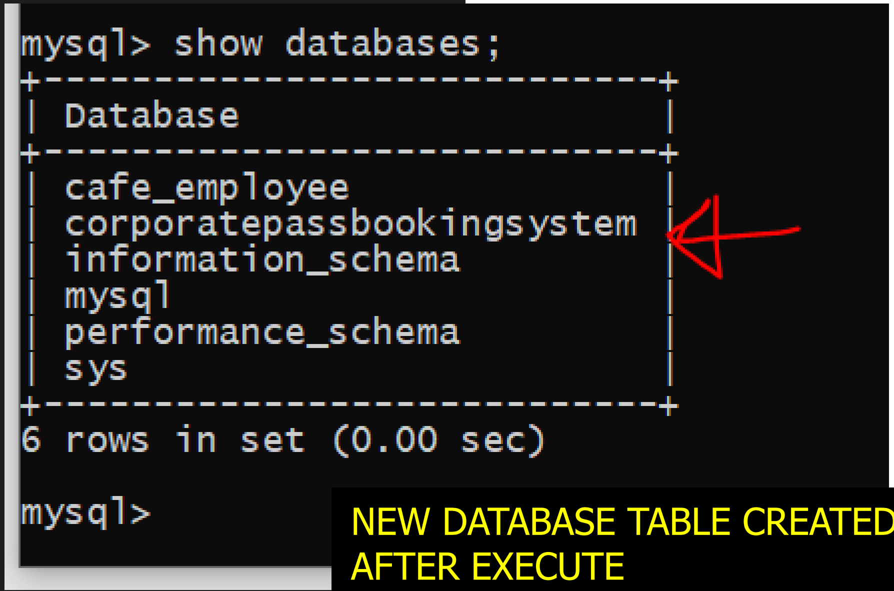
- logging to direcrtory CorporatePassBookingSystem

SETUP OF DATABASE AND TABLES

- In Terminal run : dotnet ef migrations InitializeDB
- In Terminal run : dotnet ef database update



```
PS C:\_CODINGS\INTERVIEW\JTC\CorporatePassBookingSystem> dotnet ef database update
Build started...
Build succeeded.
info: Microsoft.EntityFrameworkCore.Database.Command[20101]
      Executed DbCommand (751ms) [Parameters=[], CommandType='Text', CommandTimeout='30']
      CREATE DATABASE `corporatepassbookingsystem`;
info: Microsoft.EntityFrameworkCore.Database.Command[20101]
      Executed DbCommand (1,297ms) [Parameters=[], CommandType='Text', CommandTimeout='30']
      CREATE TABLE `__EFMigrationsHistory` (
          `MigrationId` varchar(150) CHARACTER SET utf8mb4 NOT NULL,
          `ProductVersion` varchar(32) CHARACTER SET utf8mb4 NOT NULL,
          CONSTRAINT `PK___EFMigrationsHistory` PRIMARY KEY (`MigrationId`)
      ) CHARACTER SET=utf8mb4;
info: Microsoft.EntityFrameworkCore.Database.Command[20101]
      Executed DbCommand (113ms) [Parameters=[], CommandType='Text', CommandTimeout='30']
      SELECT 1 FROM INFORMATION_SCHEMA.TABLES WHERE TABLE_SCHEMA='corporatepassbookingsystem' AND TABLE_NAME='__EFMigrationsHistor
info: Microsoft.EntityFrameworkCore.Database.Command[20101]
      Executed DbCommand (4ms) [Parameters=[], CommandType='Text', CommandTimeout='30']
      SELECT `MigrationId`, `ProductVersion`
      FROM `__EFMigrationsHistory`
      ORDER BY `MigrationId`;
info: Microsoft.EntityFrameworkCore.Migrations[20402]
      Applying migration '20241024053623_InitialDB'.
Applying migration '20241024053623_InitialDB'.
info: Microsoft.EntityFrameworkCore.Database.Command[20101]
      Executed DbCommand (191ms) [Parameters=[], CommandType='Text', CommandTimeout='30']
      ALTER DATABASE CHARACTER SET utf8mb4;
info: Microsoft.EntityFrameworkCore.Database.Command[20101]
      Executed DbCommand (1,000ms) [Parameters=[], CommandType='Text', CommandTimeout='30']
      CREATE TABLE `BookingHistories` (
          `Id` int NOT NULL AUTO_INCREMENT,
          `BookingId` int NOT NULL,
          `BookingDate` datetime(6) NOT NULL,
          `CheckInDate` datetime(6) NOT NULL,
          `CheckOutDate` datetime(6) NOT NULL,
          `Status` longtext CHARACTER SET utf8mb4 NULL,
          CONSTRAINT `PK_BookingHistories` PRIMARY KEY (`Id`)
      ) CHARACTER SET=utf8mb4;
info: Microsoft.EntityFrameworkCore.Database.Command[20101]
```



```
mysql> show databases;
+----------------------------+
| Database                   |
+----------------------------+
| cafe_employee              |
| corporatepassbookingsystem |
| information_schema         |
| mysql                      |
| performance_schema         |
| sys                        |
+----------------------------+
6 rows in set (0.00 sec)

mysql>
```

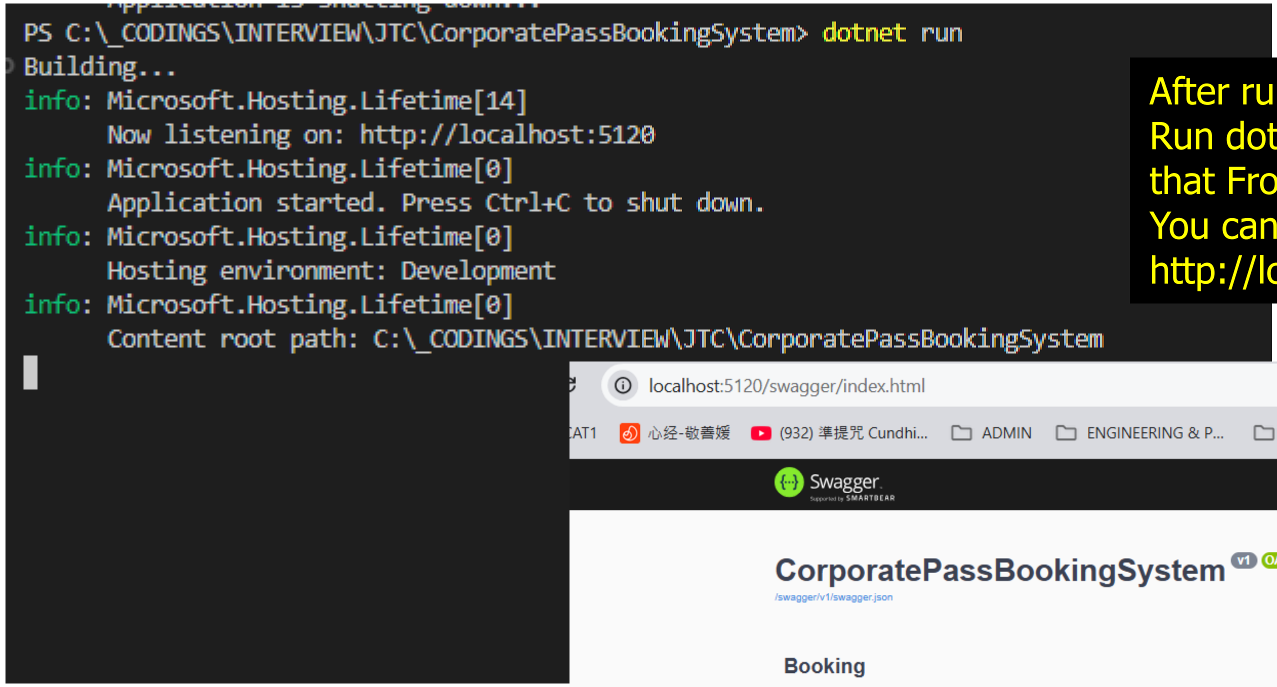
NEW DATABASE TABLE CREATED
AFTER EXECUTE
Dotnet ef database update

SETUP OF REST API (WEBAPI)

- In Terminal run : dotnet build -> To builld the solution in \bin
- In Terminal run : dotnet run -> This will fired up the REST API Server
- In Browser type : http://localhost:5120/swagger/index.html -> This will run swagger to verify and check on the EndPoints

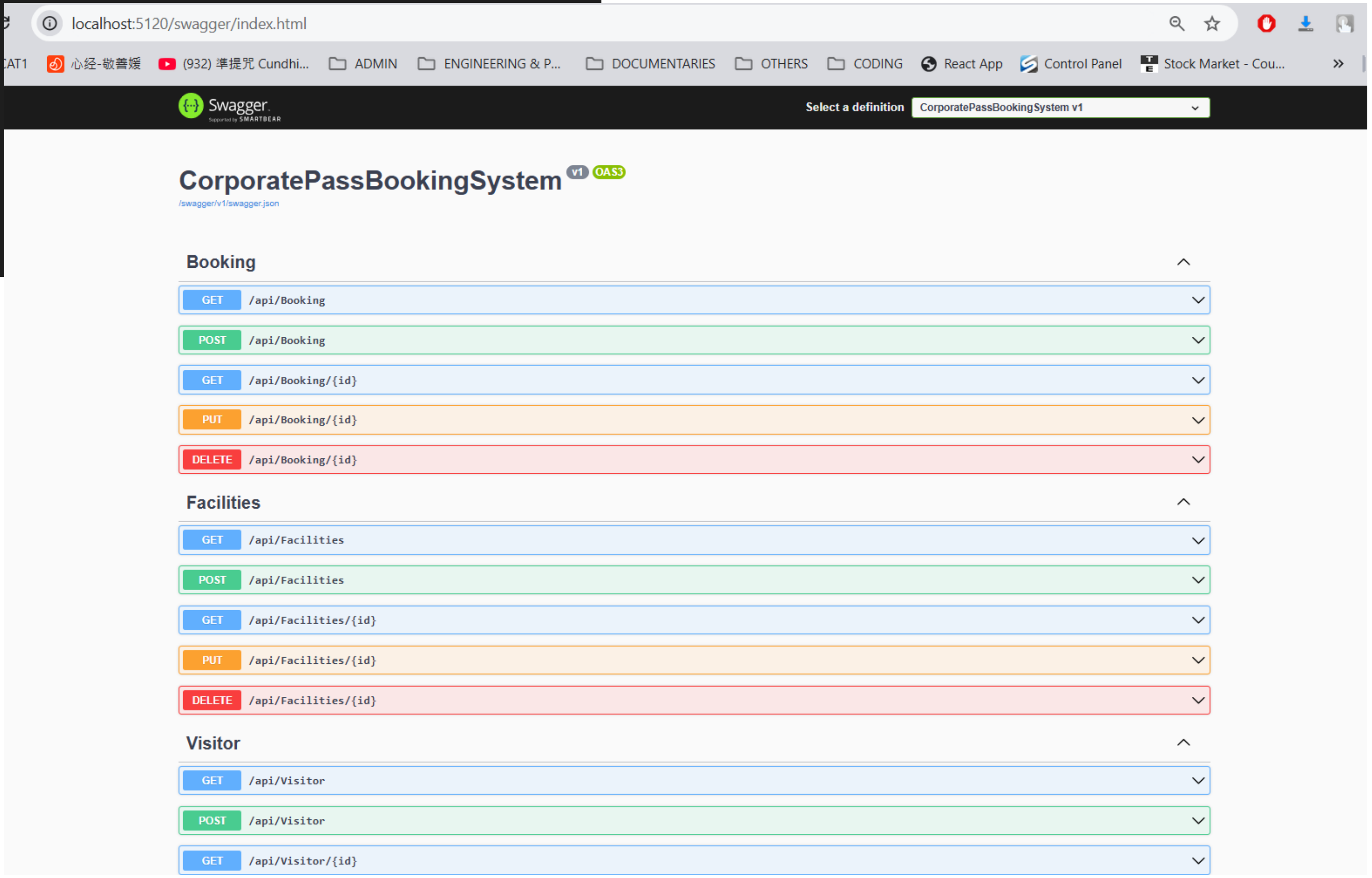
o GET /api/facilities – List all facilities
o GET /api/facility/{id} – Get a single facility
o GET /api/bookings – Get all booking list
o GET /api/bookings/{VisitorId} – Get booking by visitor
o GET /api/booking/{id} – Get a single booking

o POST /api/booking – Create a booking
o PUT /api/booking – Update a booking
o GET /api/visitors – Get all visitors
o GET /api/visitor/{id} – Get a single visitor
o POST /api/visitor – Create a visitor
o PUT /api/visitor – Update a visitor



```
PS C:\_CODINGS\INTERVIEW\JTC\CorporatePassBookingSystem> dotnet run
Building...
info: Microsoft.Hosting.Lifetime[14]
      Now listening on: http://localhost:5120
info: Microsoft.Hosting.Lifetime[0]
      Application started. Press Ctrl+C to shut down.
info: Microsoft.Hosting.Lifetime[0]
      Hosting environment: Development
info: Microsoft.Hosting.Lifetime[0]
      Content root path: C:\_CODINGS\INTERVIEW\JTC\CorporatePassBookingSystem
```

After running dotnet build with \bin and \obj will be created
Run dotnet run to fire up the REST API that become the Endpoint that FrontEnd App can access.
You can also in browser start swagger to check and test the API
http://localhost:5120/swagger/index.html



CorporatePassBookingSystem v1 OAS3
/swagger/v1/swagger.json

Booking

```
GET     /api/Booking
POST    /api/Booking
GET     /api/Booking/{id}
PUT     /api/Booking/{id}
DELETE  /api/Booking/{id}
```

Facilities

```
GET     /api/Facilities
POST    /api/Facilities
GET     /api/Facilities/{id}
PUT     /api/Facilities/{id}
DELETE  /api/Facilities/{id}
```

Visitor

```
GET     /api/Visitor
POST    /api/Visitor
GET     /api/Visitor/{id}
```