**Phase-End Project**

**EMS Full Stack App-Requirement**

**Aim:**

To create a Full Stack application EMS (Employee Management System) to maintain their employee profile details.

**Github link:**

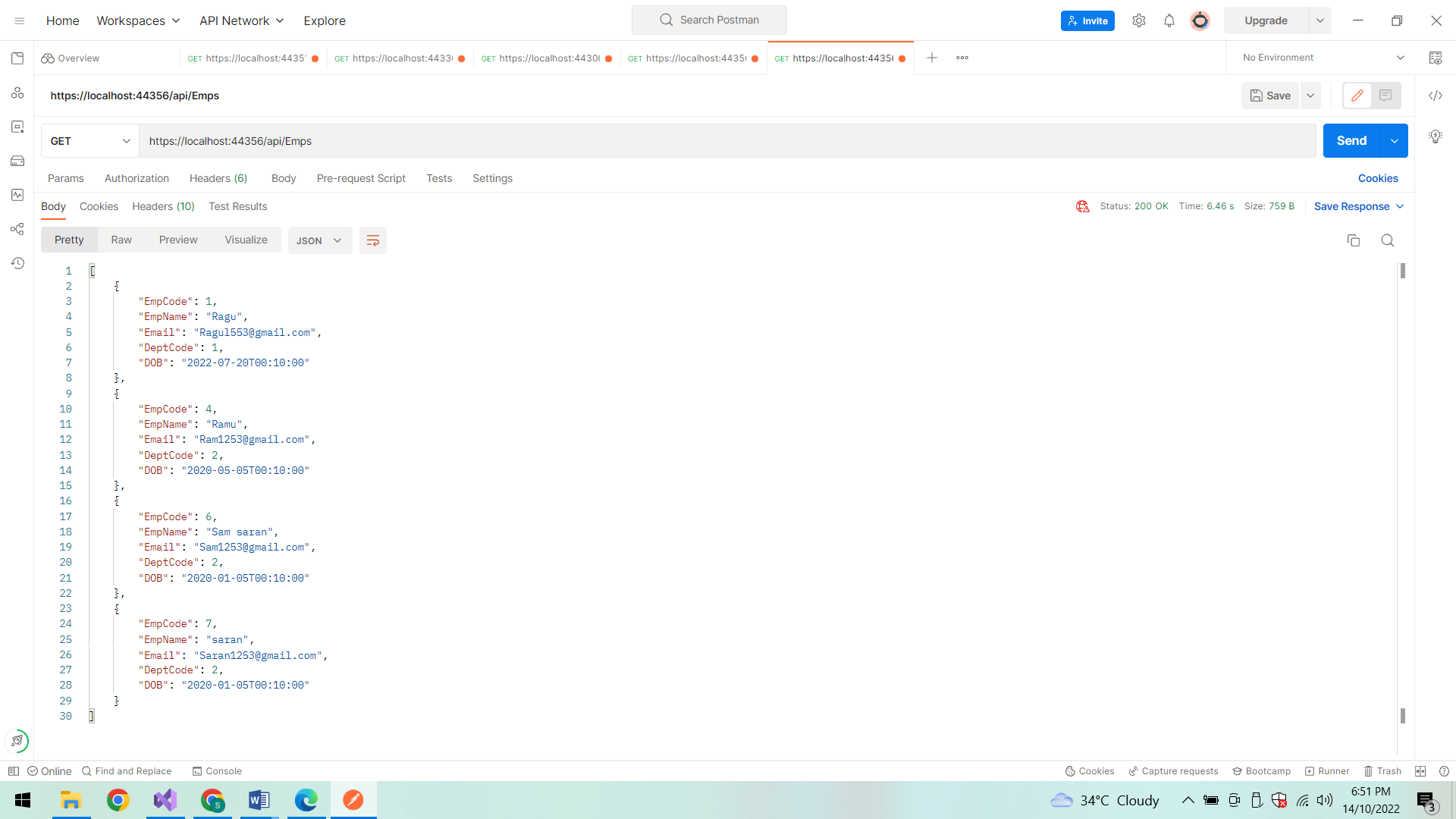
[**https://github.com/Sukirthalakshmanan1/Project\_API**](https://github.com/Sukirthalakshmanan1/Project_API)

**Procedure:**

* Create an ASP.NET Web application.Select WEB API application.
* Create a Library(DAL)Data Access Layer and Create a class EmpProfile,DeptMaster in Code first Approach and set necessary properties.
* Download Entity Framework
* Create a Business Logic Layer(BLL) and set the properties.
* Create EmpOperations and perform the insert,update,delete,List,search operations in DAL Library.
* In DeptMaster class use DropCreateDatabaseIfModelChanges<ContextClass> and add some default data into the DeptMaster table by overriding the Seed method.
* Add dependencies for the Project.
* Add controller named Emps and Add methods GET,POST,PUT,Delete,search methods and access the operation with the help of EmpOperations class.
* Add model class Emp.cs in Model Folder and write the necessary properties like EmpCode,EmpName,DeptCode,DOB,Email.
* Download Postman Application and create an account.
* Run the project and copy the URL and paste it in Postman.
* Test the Method GET(<https://localhost:44300/api/Emps>) to get All Employee details
* POST Method(/api/Emps) –To Insert Employee details in the database.
* PUT(/api/Emps/id)-To update employee details using EmpCode.
* Delete(/api/Emps/id) –To delete Employee details based on EmpCode.
* To (find) search the Employee details with id(<https://localhost:44300/FindById/2>)

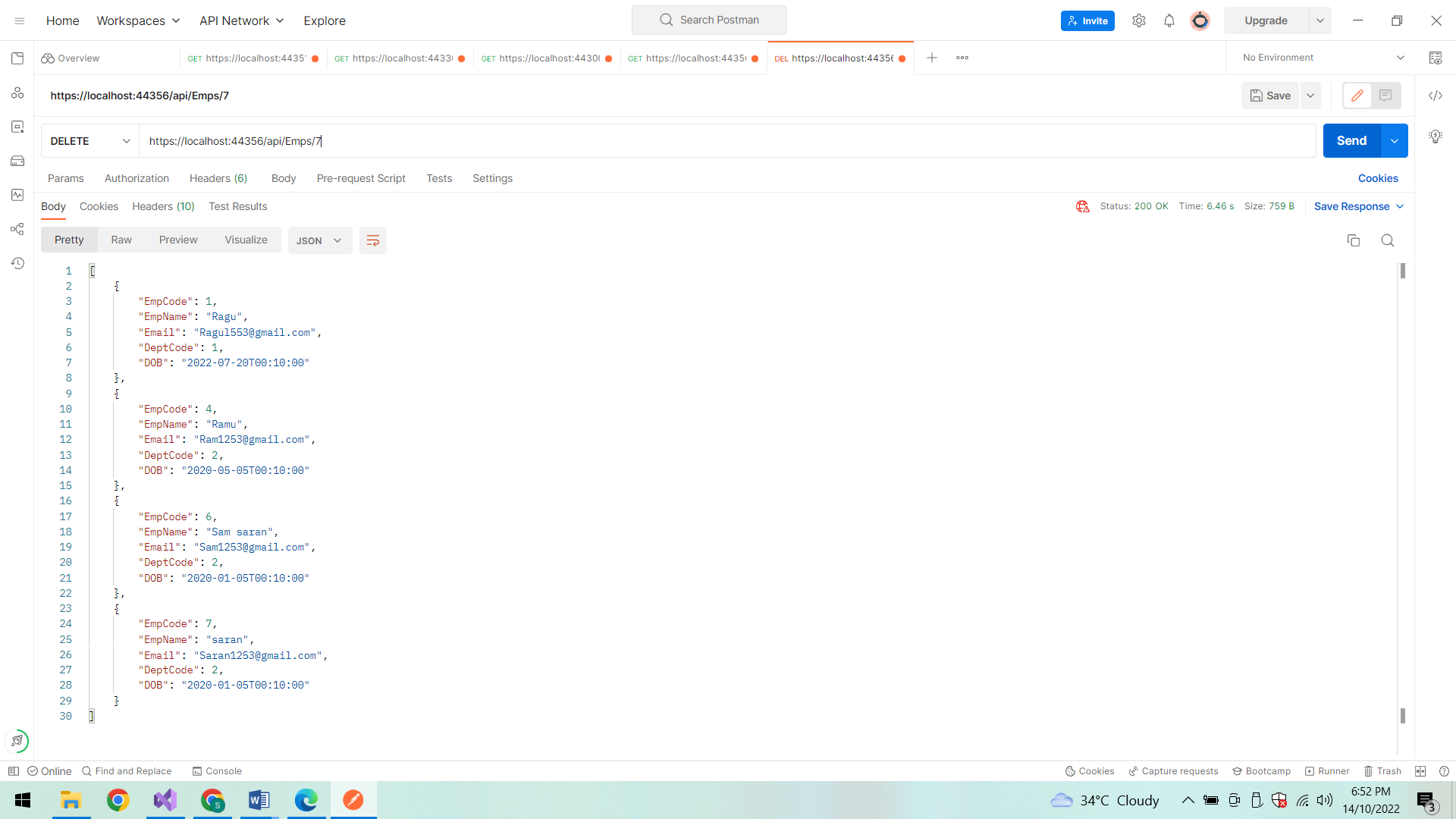
**Output:**

**GET:(Displays all Employee details)**

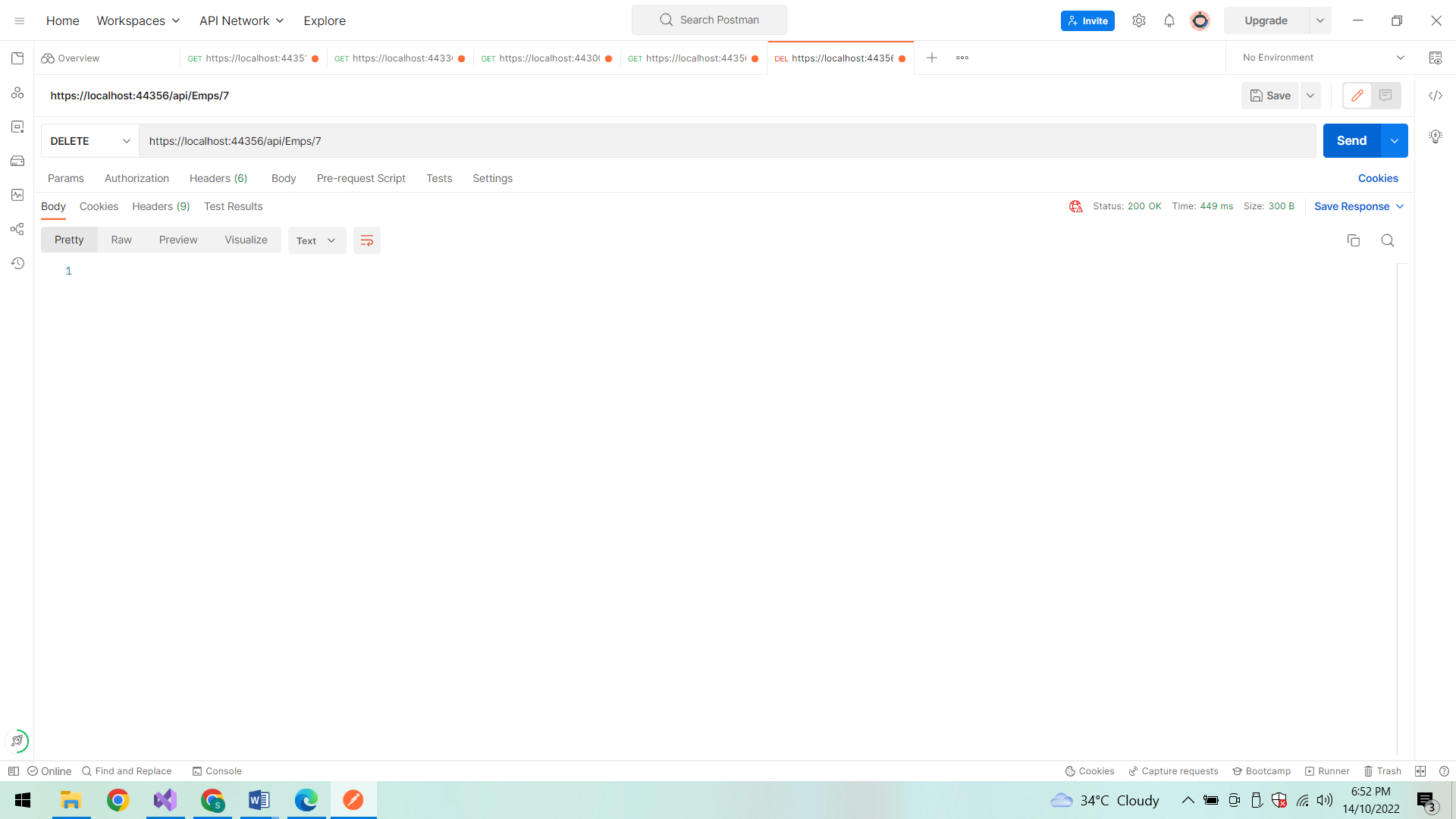
****

**DELETE Method:**

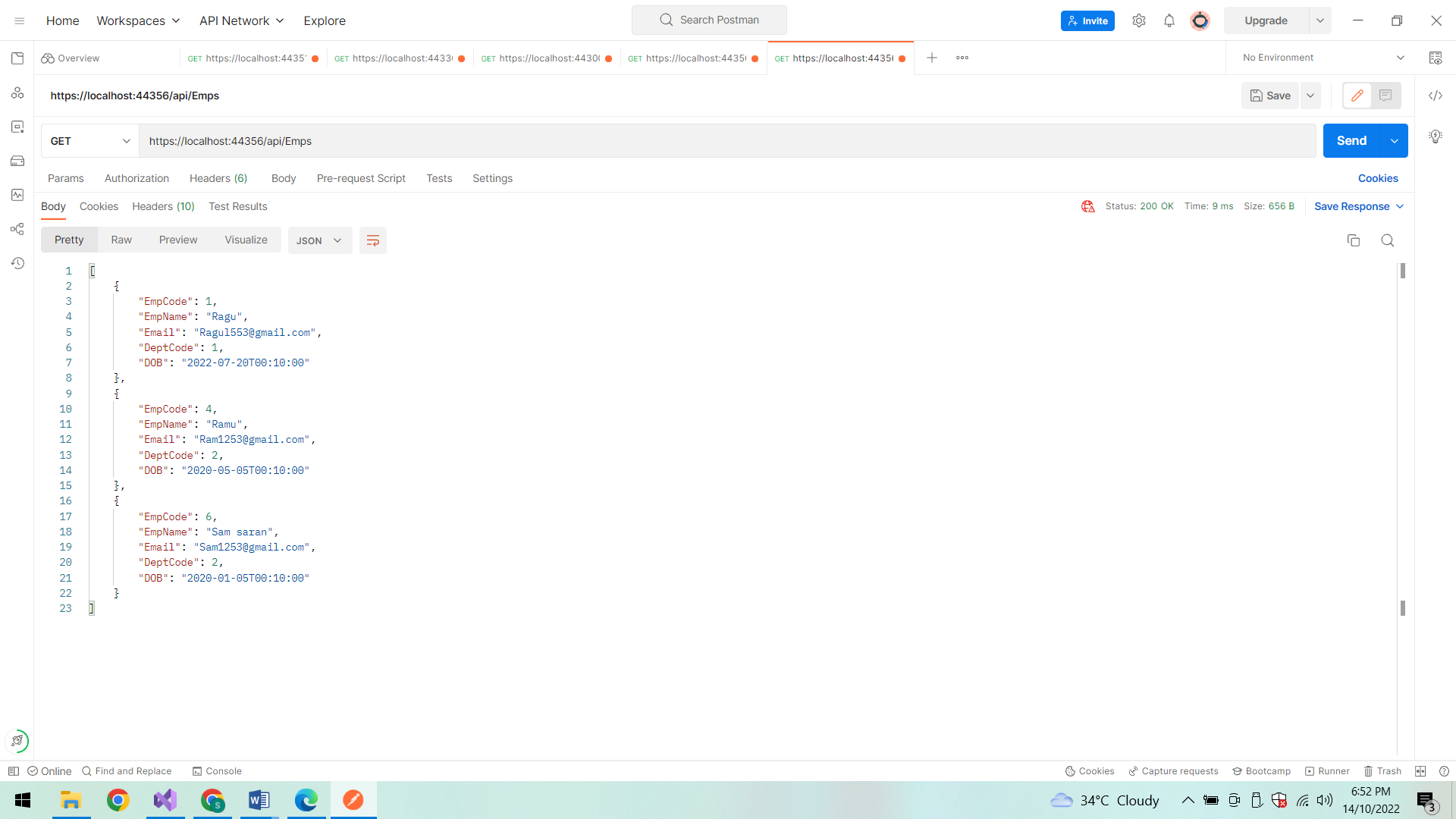
Delete based on Employee Id:

****

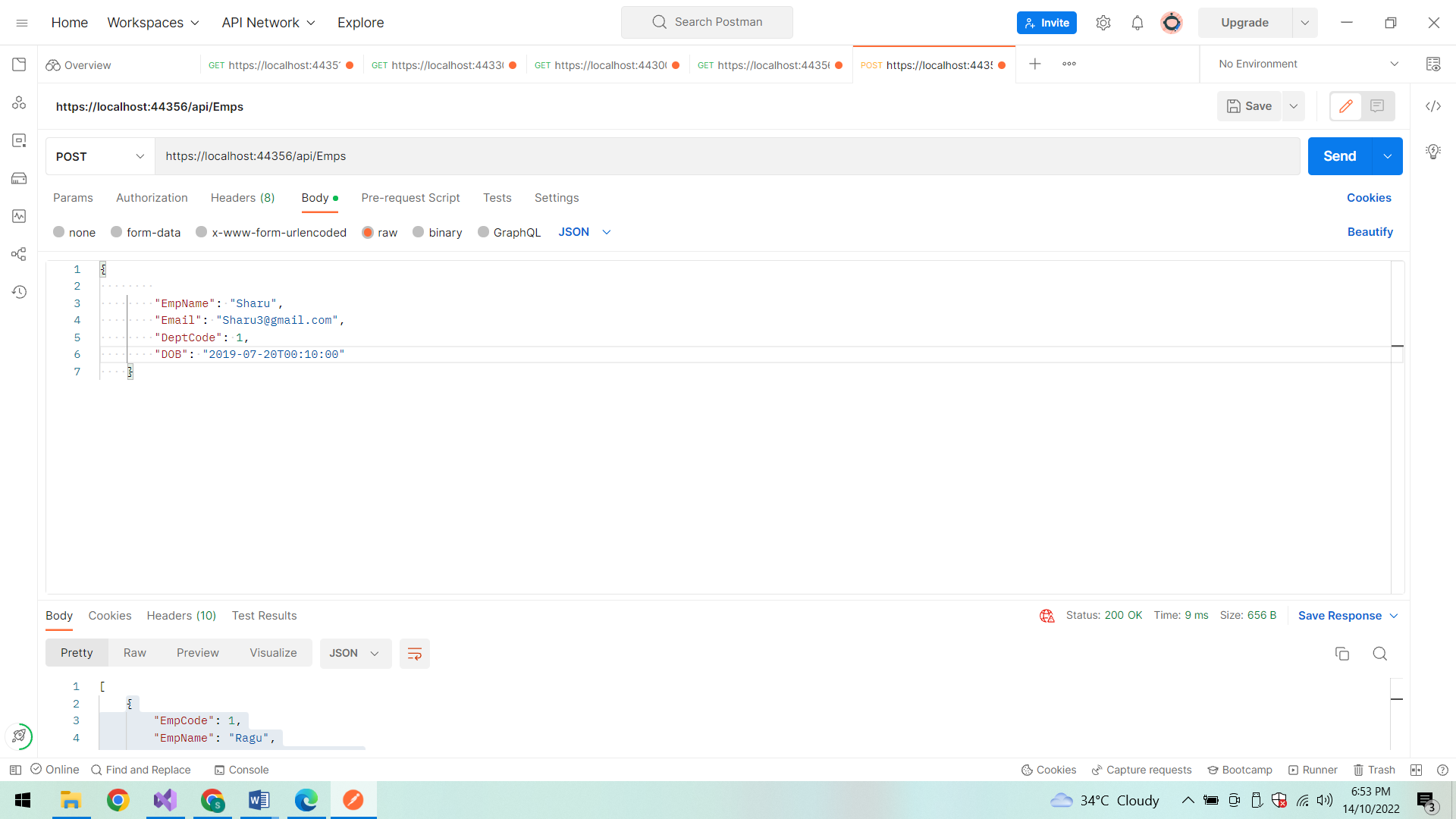
**Status code After deletion:**

****

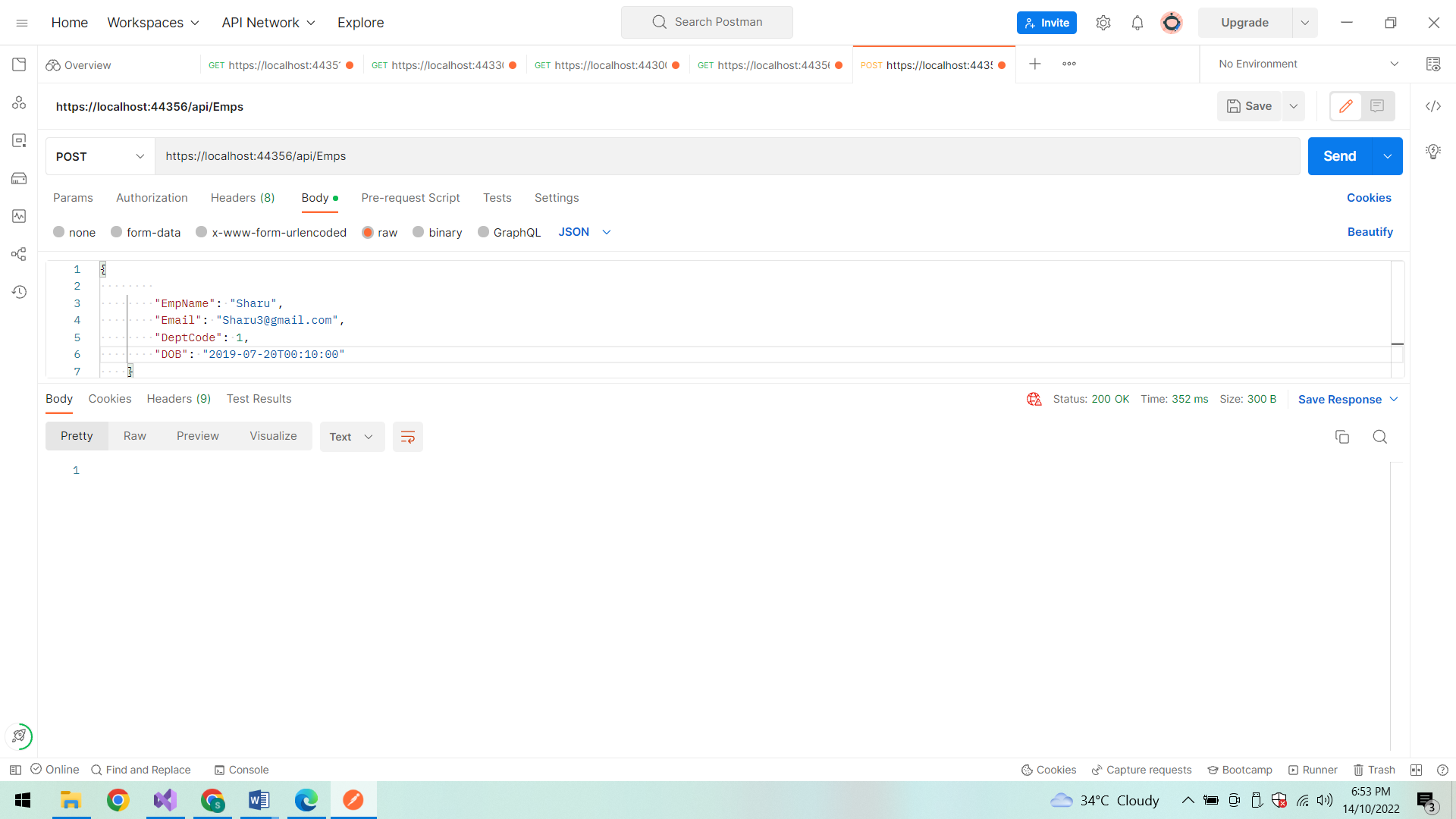
**After deletion:**

****

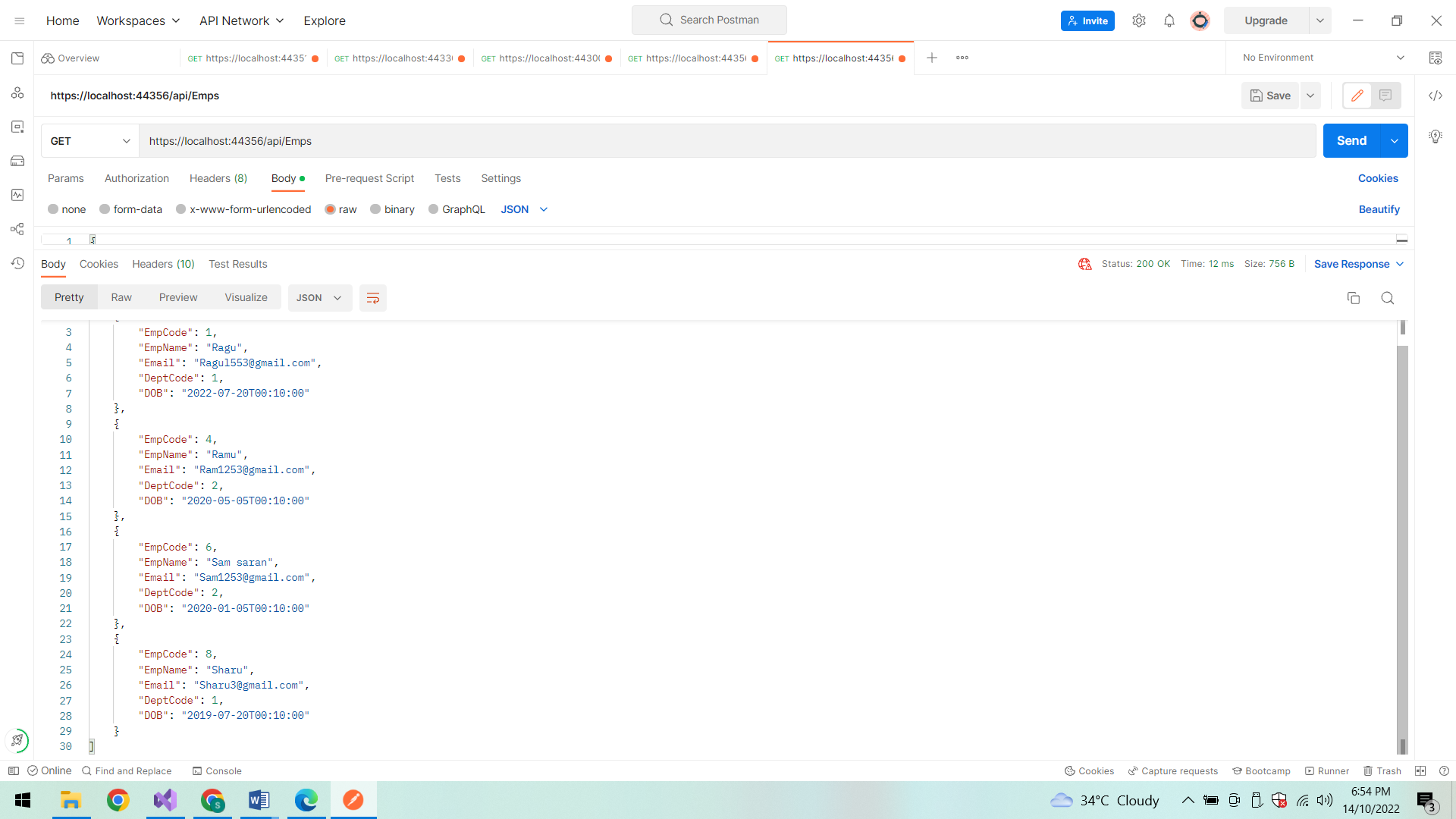
**POST Method:(To insert data in the database)**

****

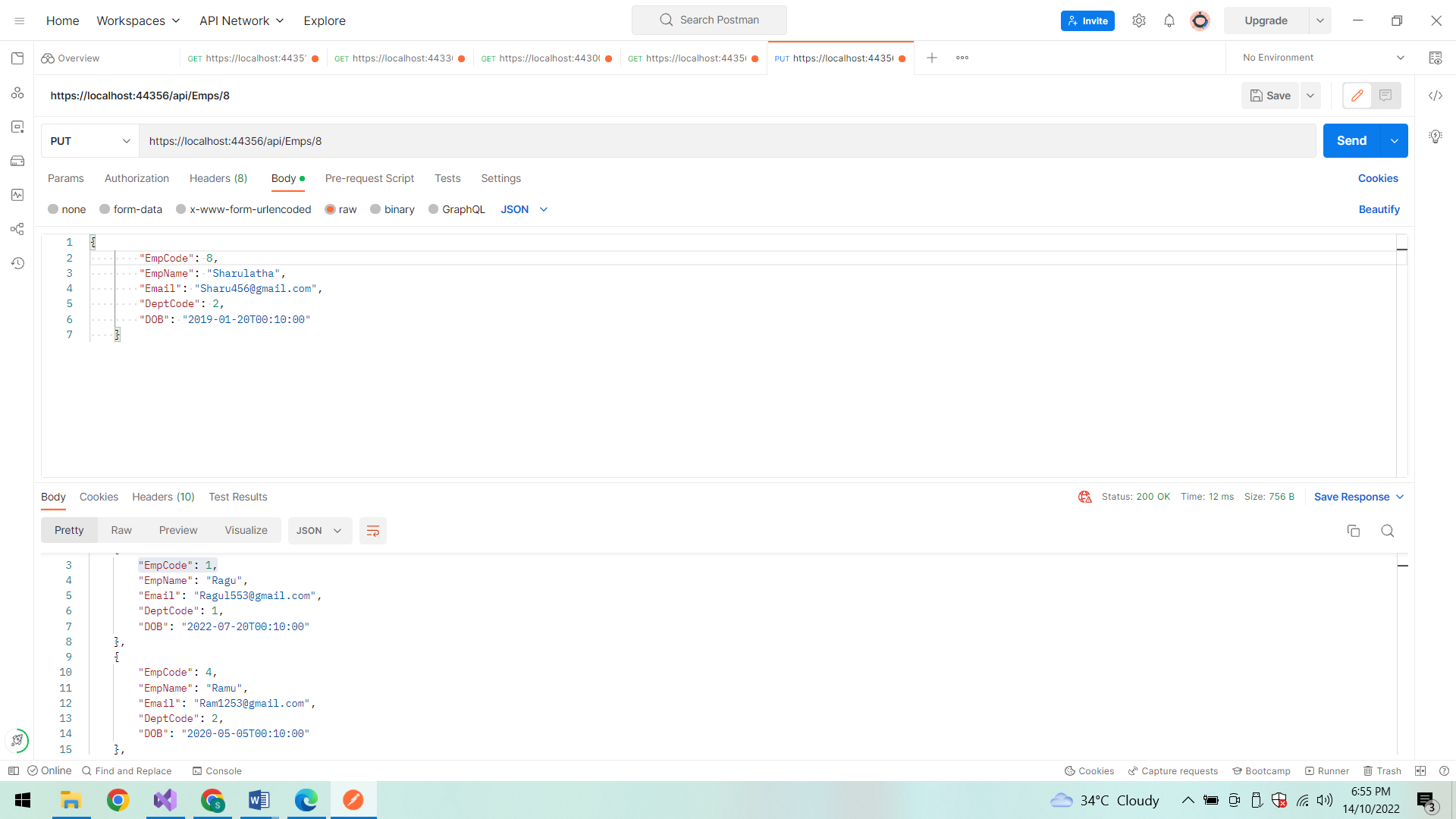
**Status code:**

****

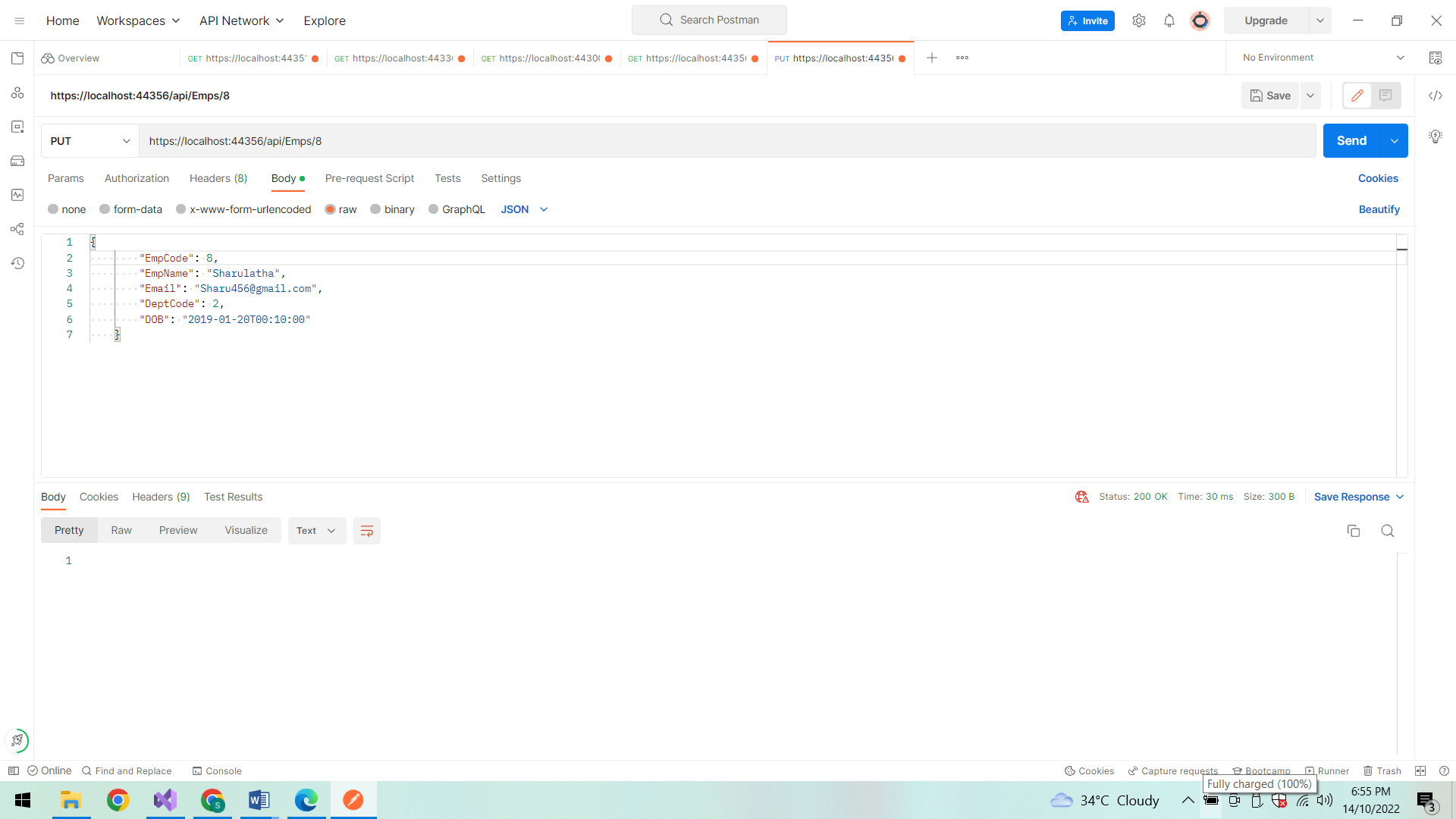
**After Insertion the Employee details are displayed:**

****

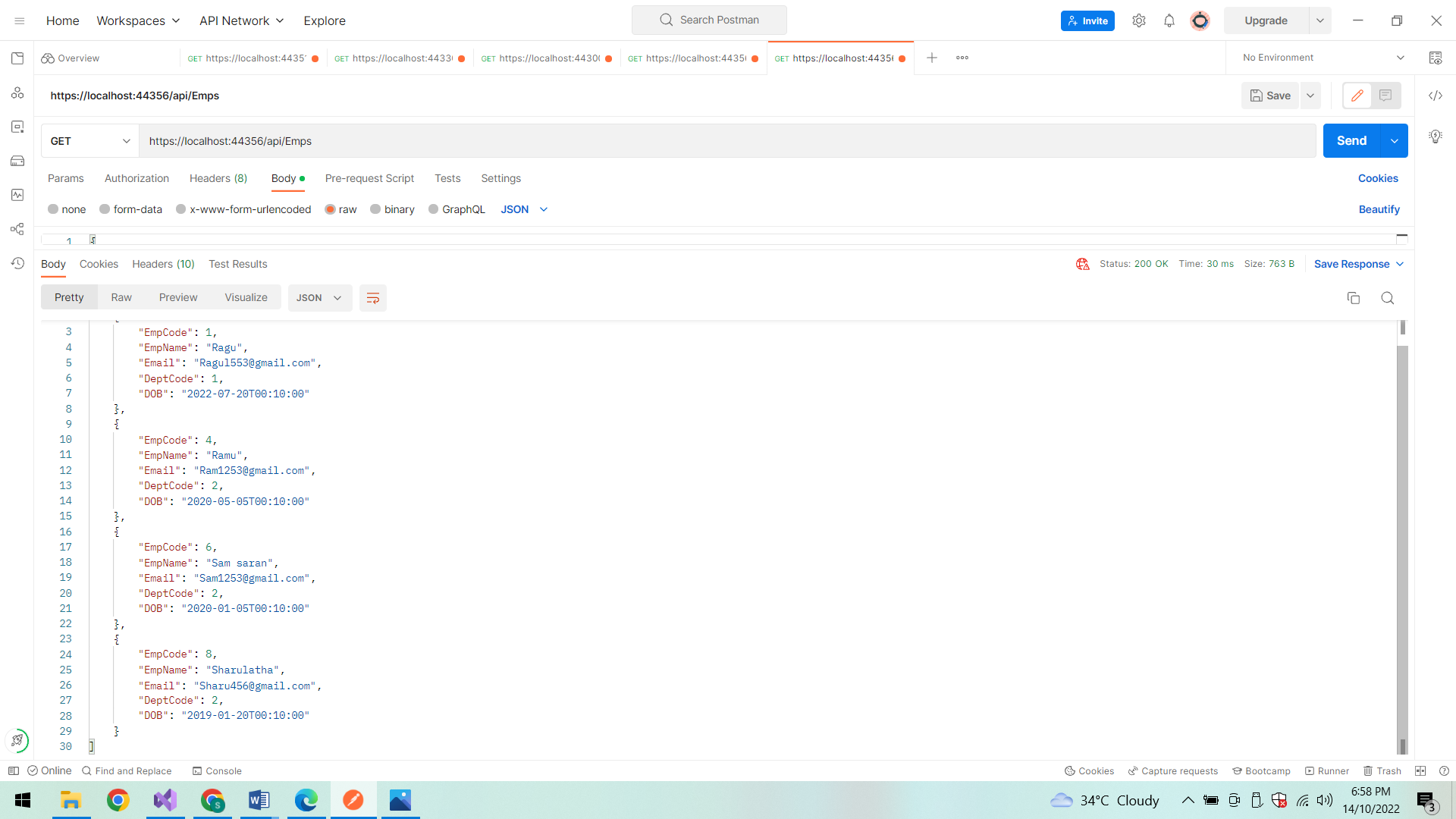
**PUT method:**

****

**Status code:**

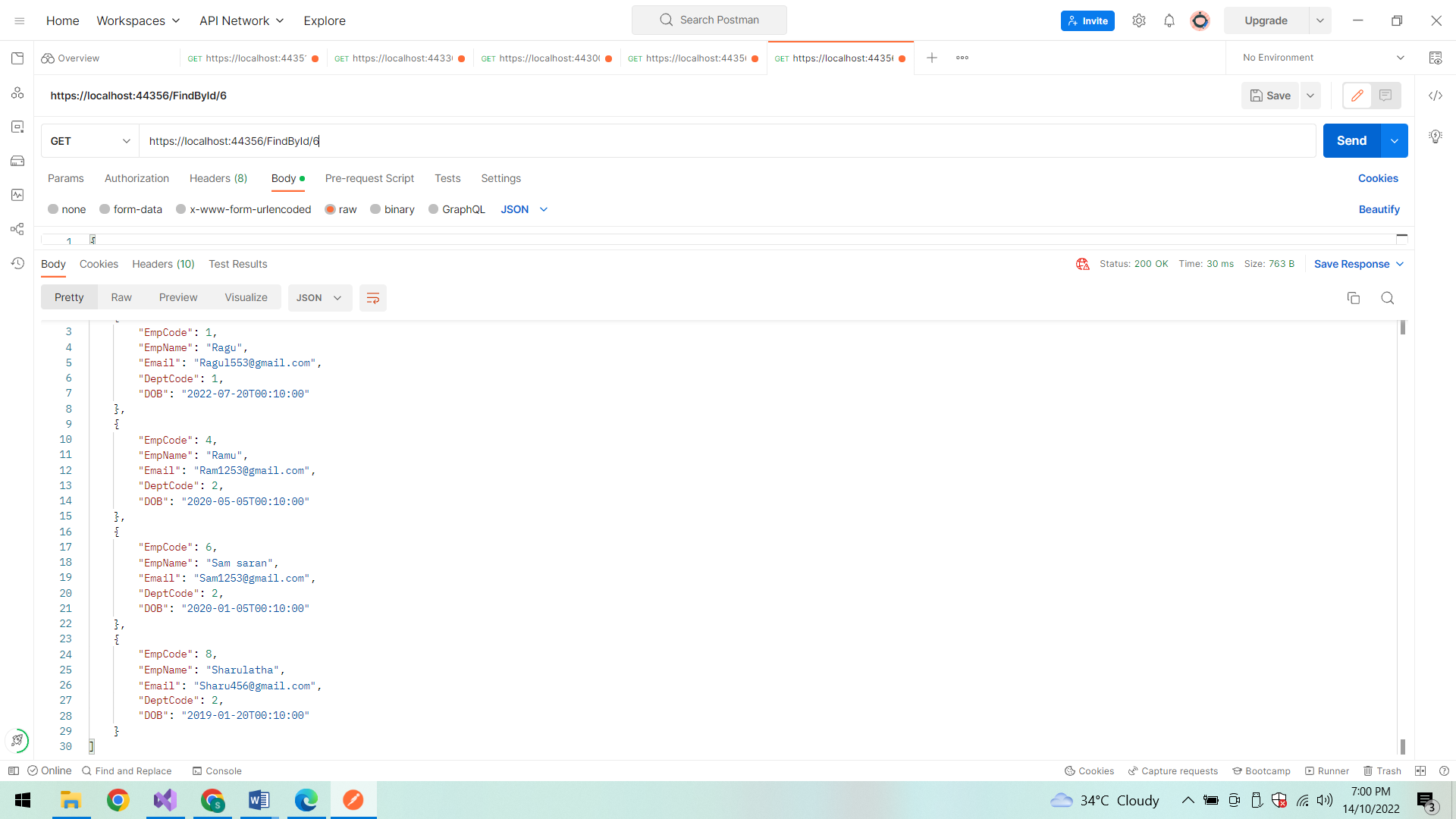
****

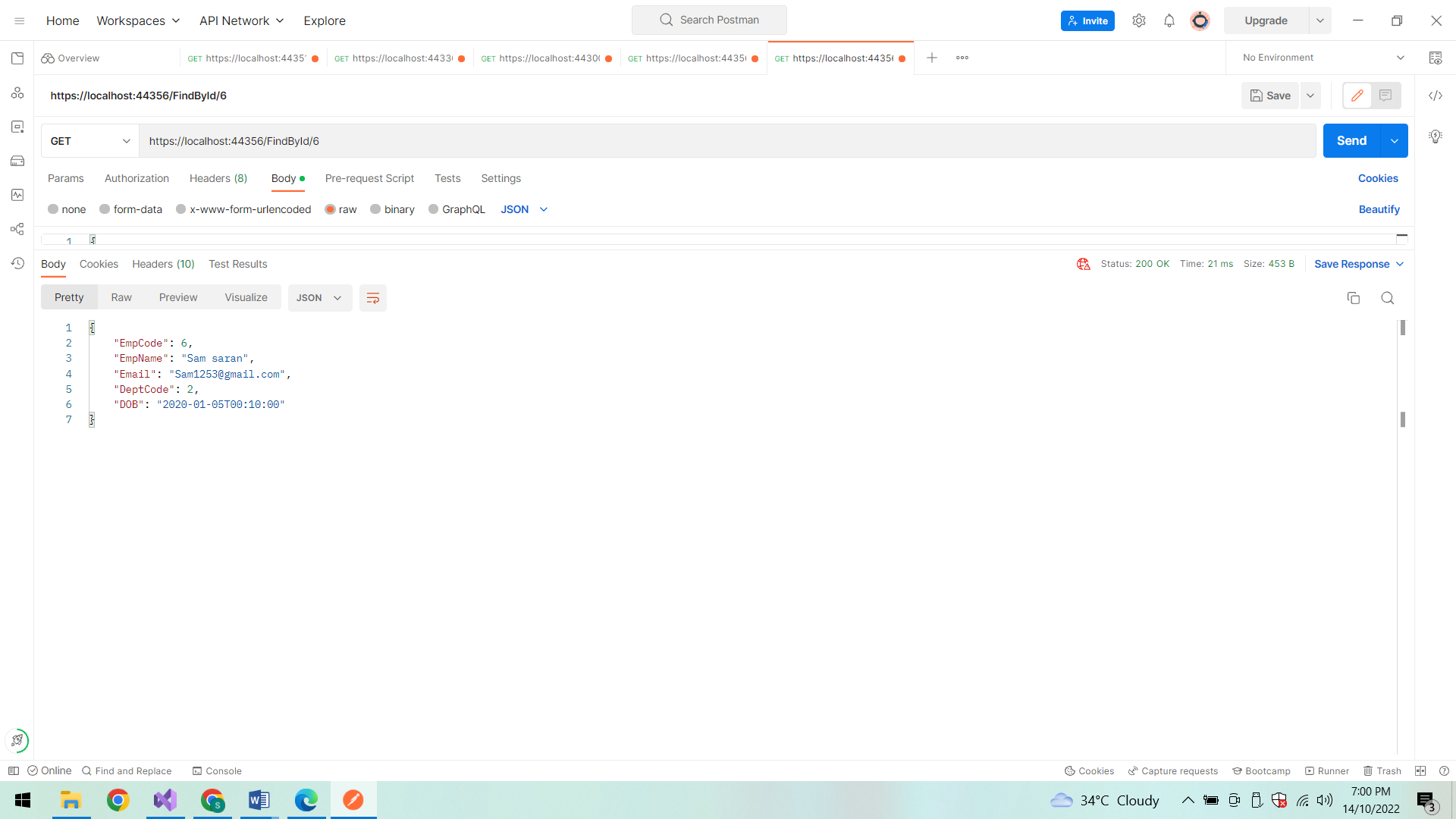
**After Updation:**

****

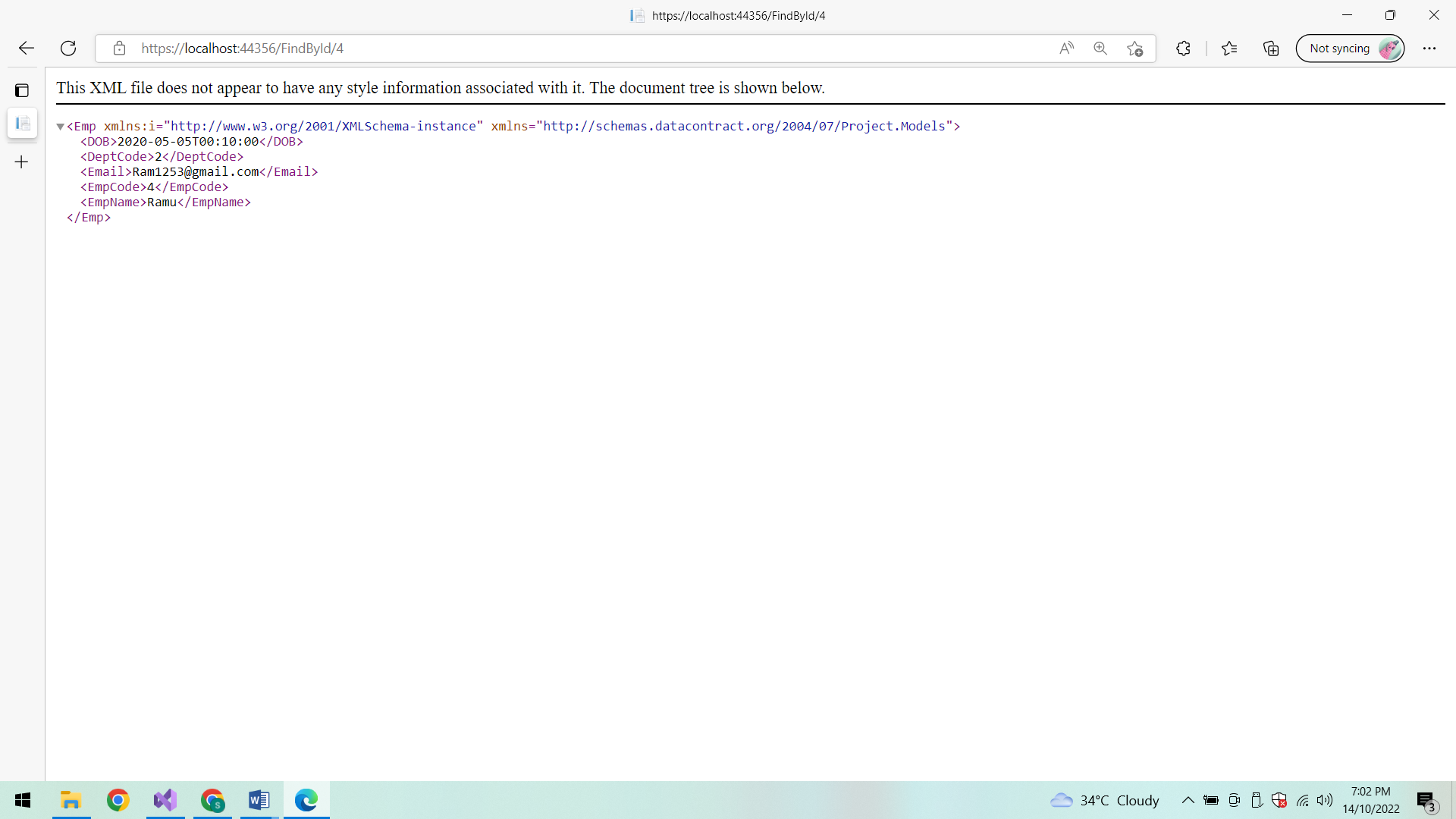
**FindById:**

Search Employee details based on EmpCode

****

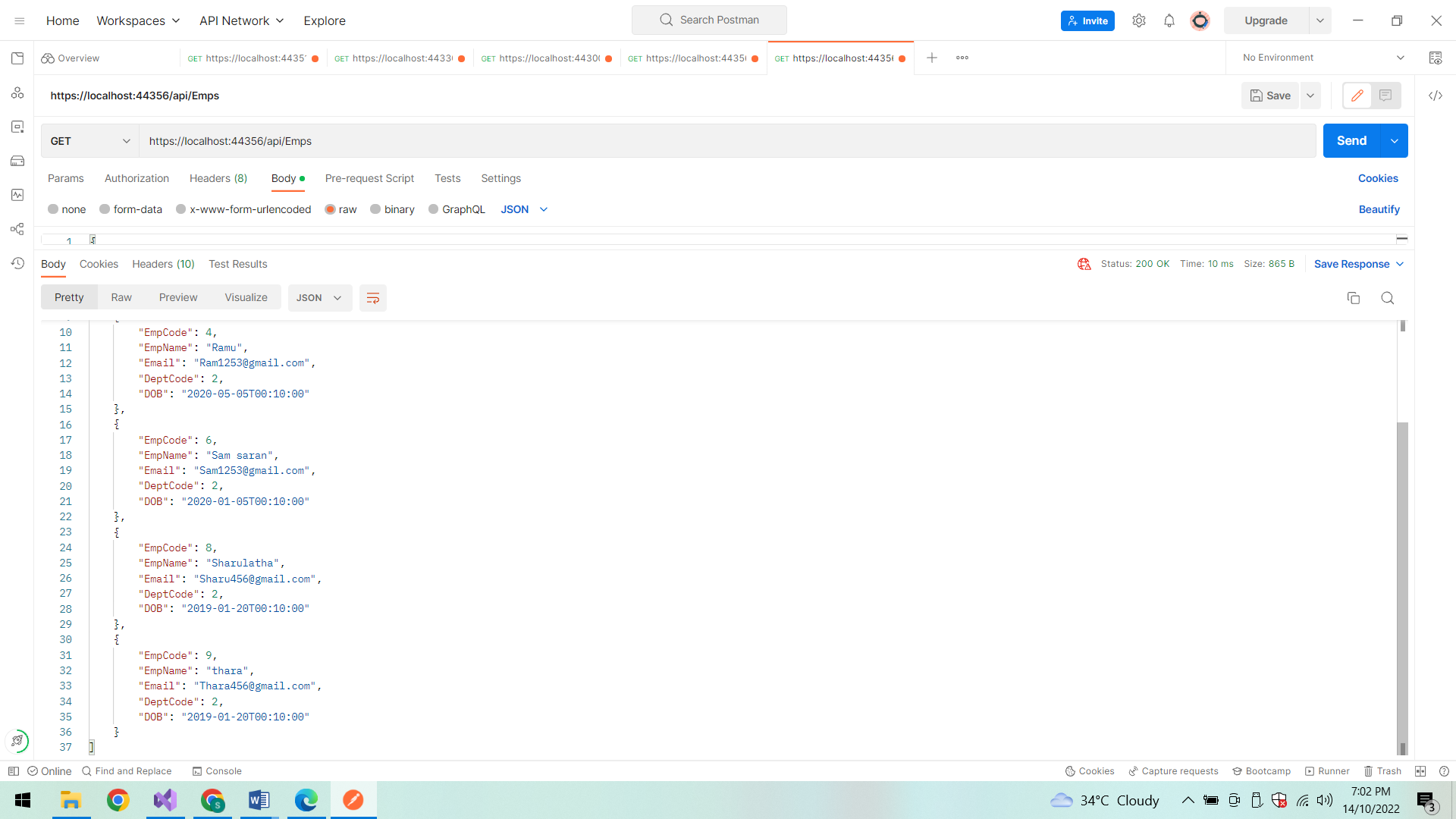
****

**FindById :**

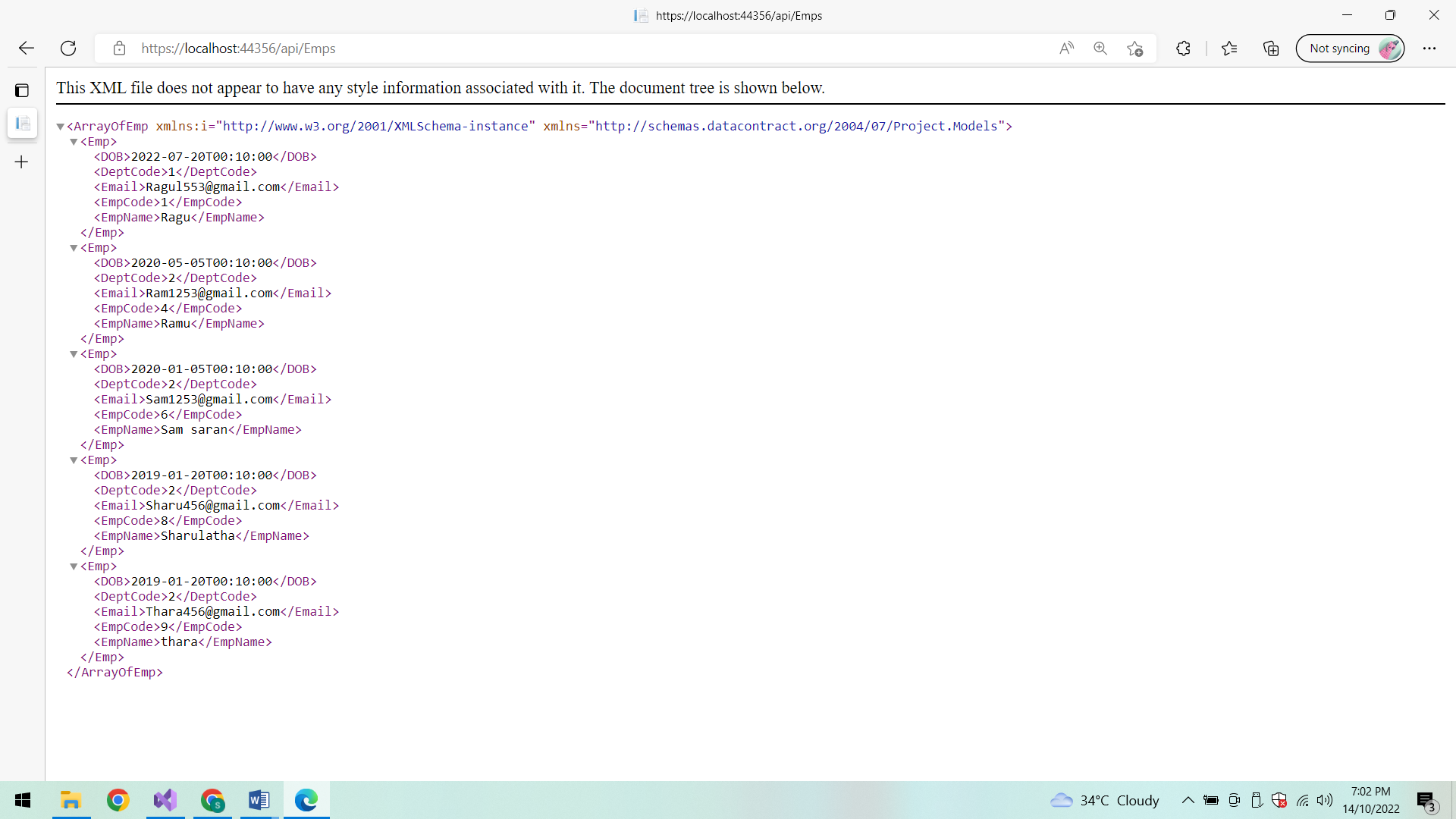
****

**Show All Employee details:**

**In PostMan:**

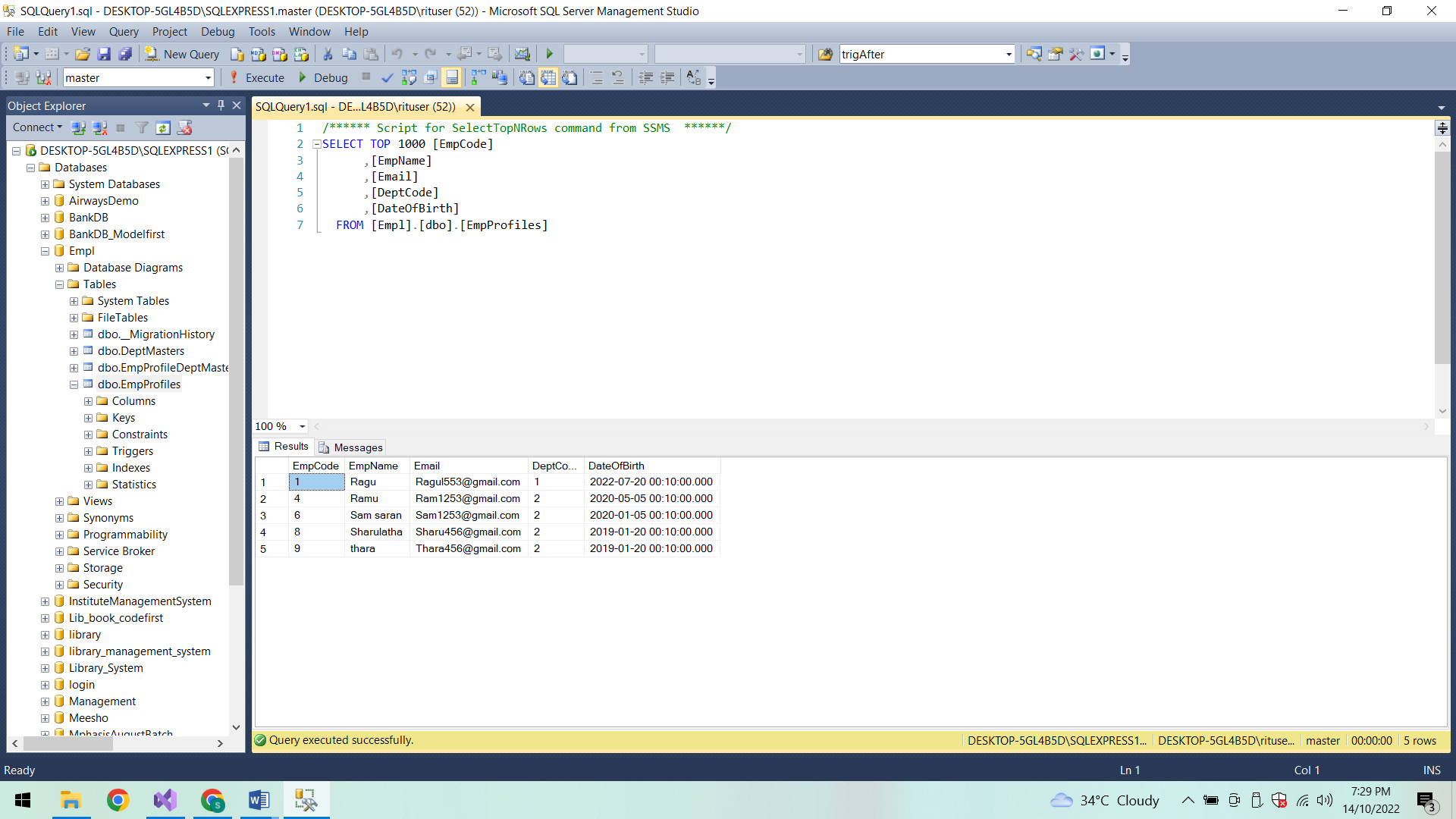
****

**In LocalHost:**

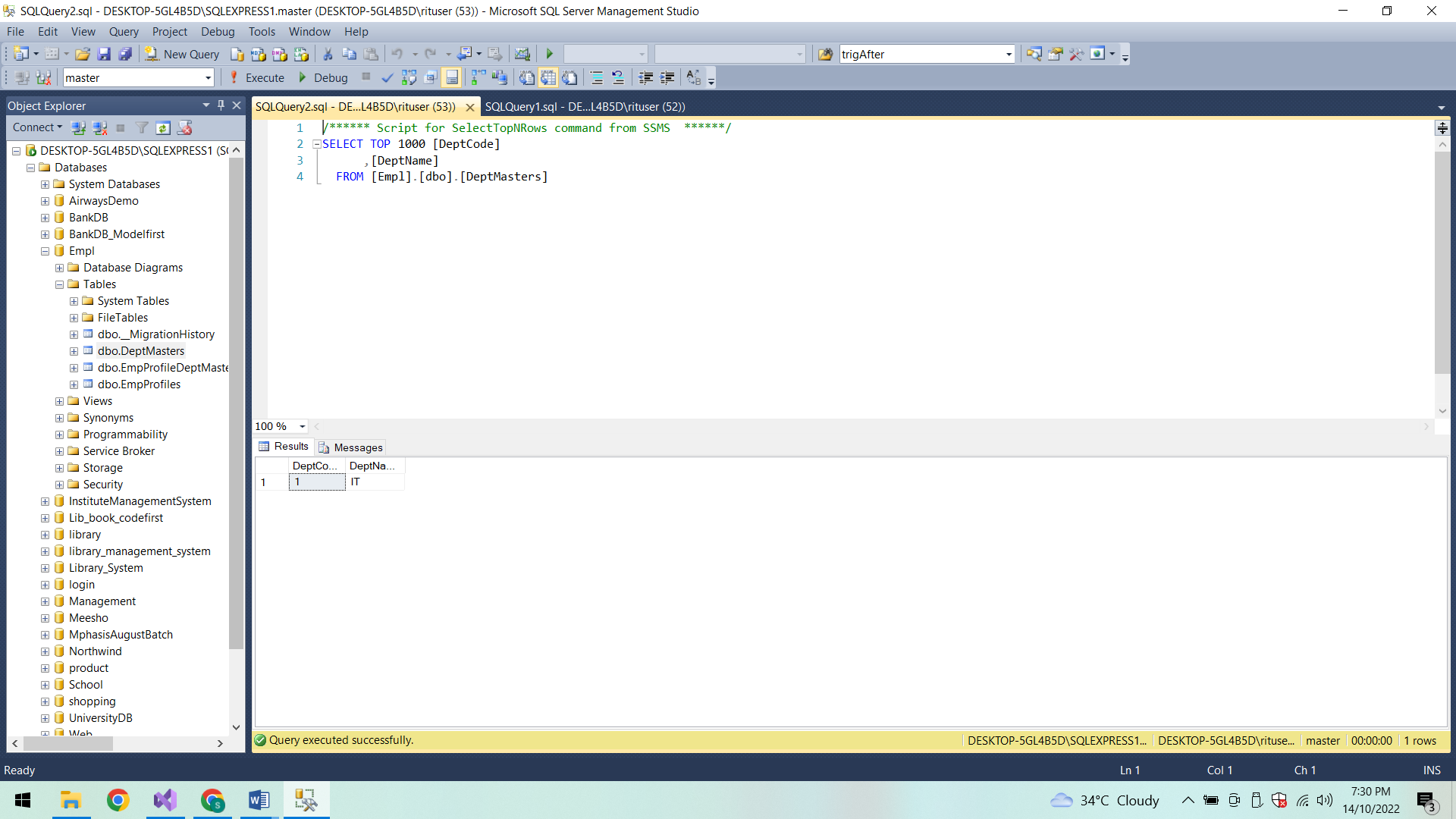
****

**Database:**

**EmpProfiles:**



**DeptMasters:**



**Conclusion:**

Thus the Employee Management System is created successfully.