**Employee Management System - Implementing CRUD Operations**

1. **Basic CRUD Operations:**
   * Used **JpaRepository** methods to create, read, update, and delete employees and departments.
   * Implemented RESTful endpoints for these operations using **EmployeeController** and **DepartmentController**.
     + - * **@RestController** indicates that this class is a RESTful controller.
         * **@RequestMapping("/api/employees") & @RequestMapping("/api/departments")** specifies the base URL for all endpoints related to employees and departments, respectively.

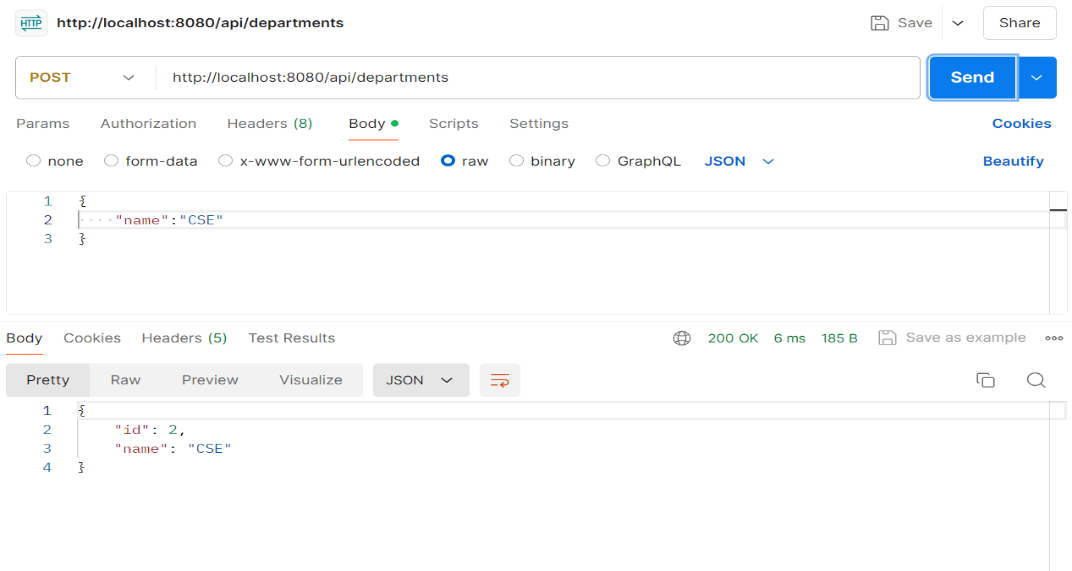
**CRUD Endpoints:**

* + - * **POST /api/employees or /api/departments** Creates a new employee or department.
      * **GET /api/employees or /api/departments** Retrieves a list of all employees or departments.
      * **GET /api/employees/{id} or /api/departments/{id}** Retrieves an employee or department by ID.
      * **PUT /api/employees/{id} or /api/departments/{id}** Updates an existing employee or department.
      * **DELETE /api/employees/{id} or /api/departments/{id}** Deletes an employee or department by ID.

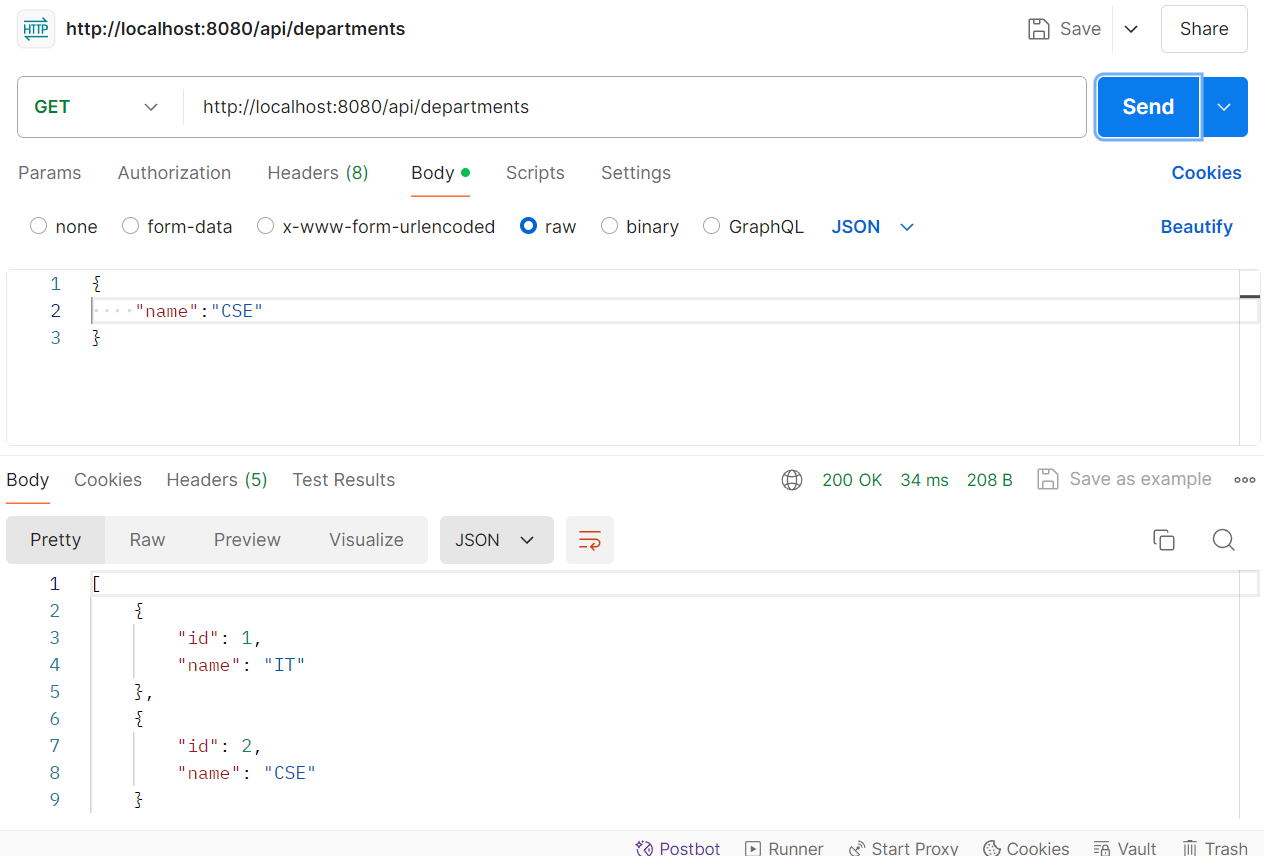
These controllers allows to manage employees and departments via standard RESTful CRUD operations.

Screenshots of PostMan:

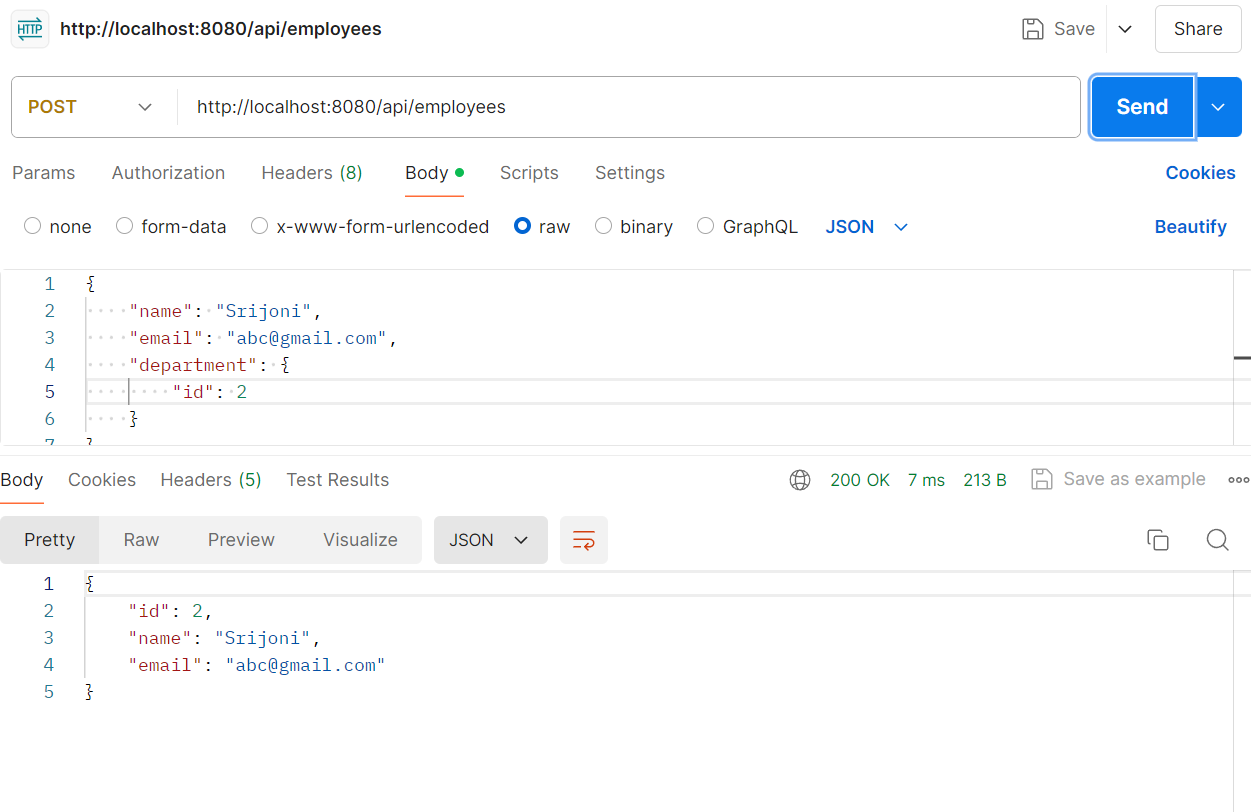
POST on Department:



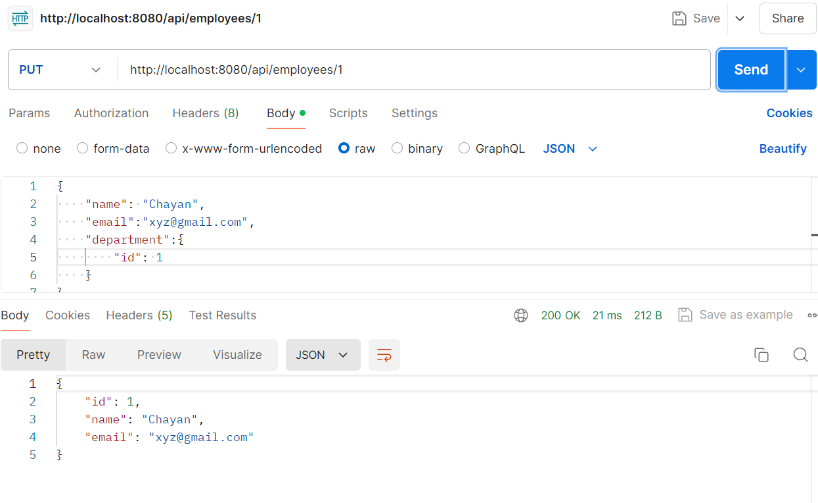
GET on Department:



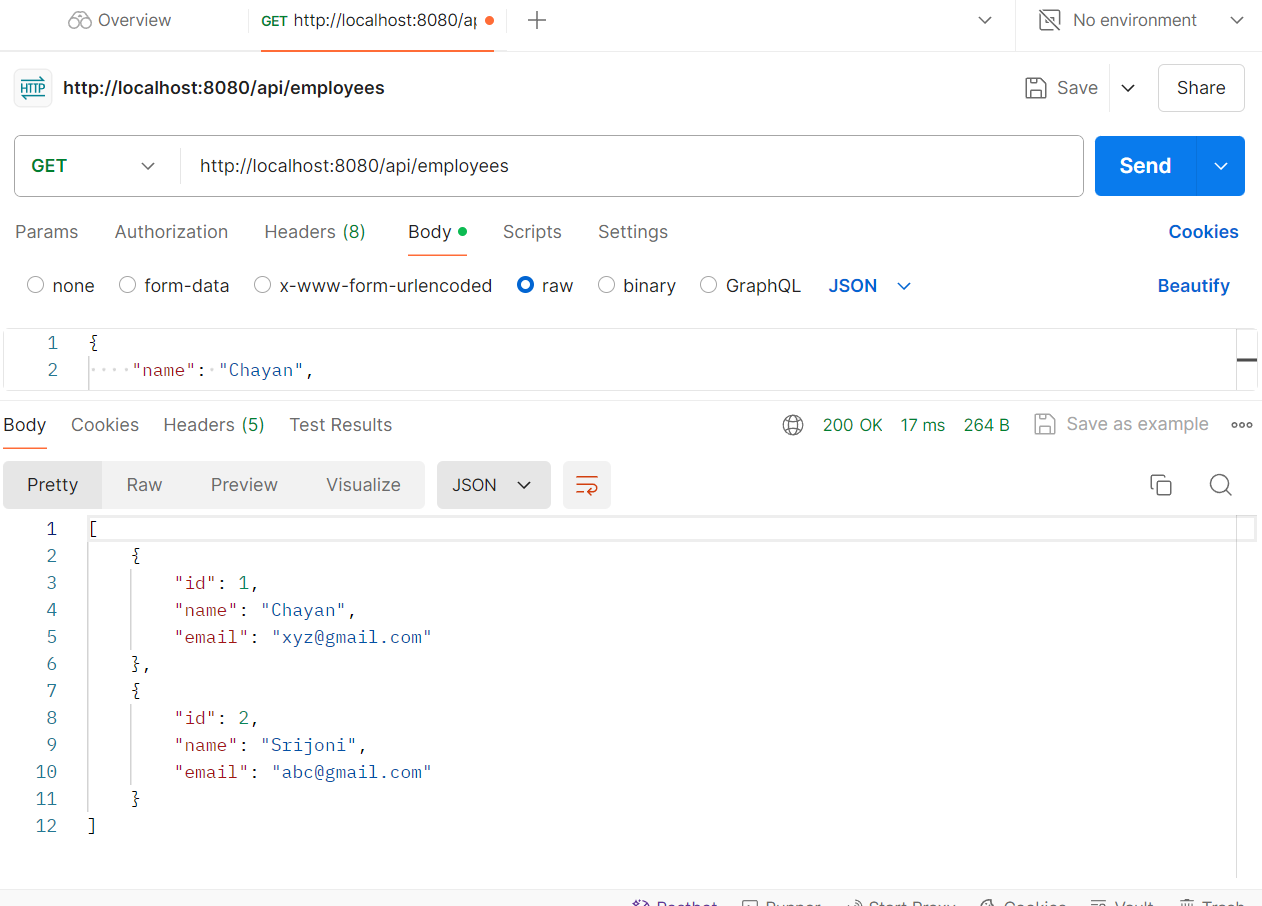
POST on Employee:



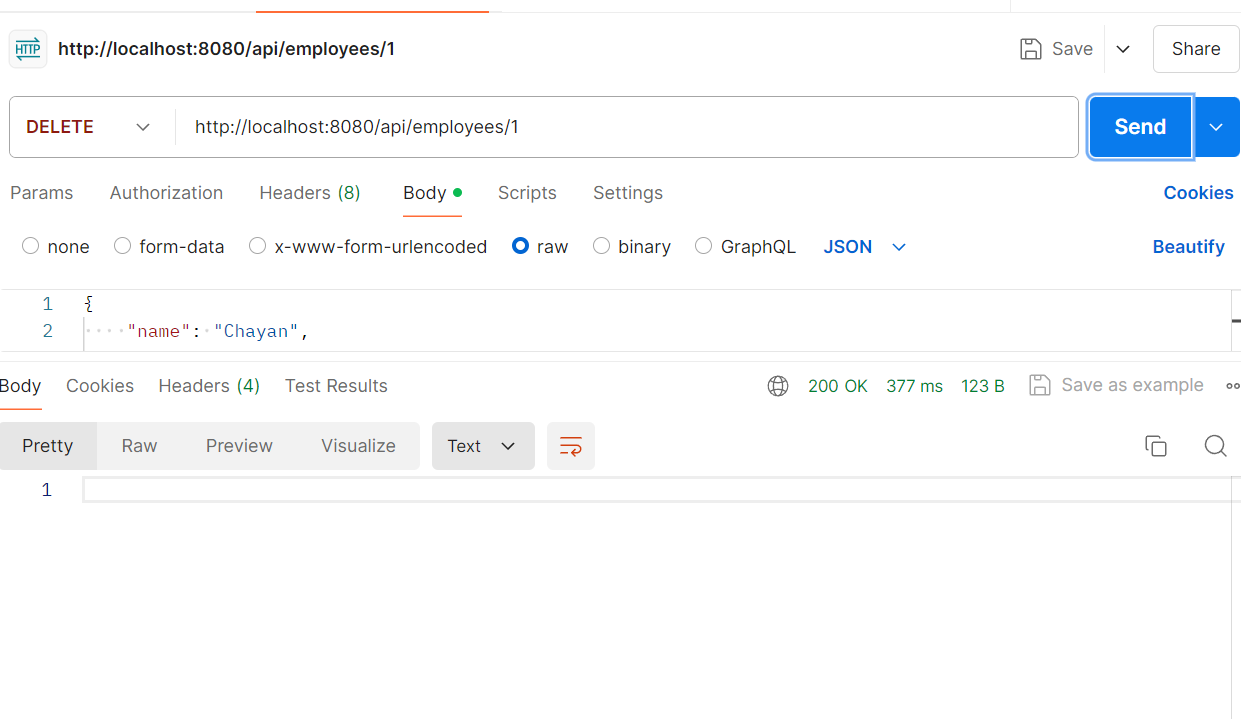
PUT on Employee id 1:



GET on Employee:



DELETE on Employee:



Checking if DELETE Worked on Employee by using GET:

