Employee operations project endpoints

1. GetAllEmployees - http://localhost:8081/getAllEmployees
2. Get employee by a specific name - http://localhost:8081/getEmployee/{name}
3. Create a new employee - http://localhost:8081/employee
4. Create list of employees- http://localhost:8081
5. Create an array of employees - http://localhost:8081/employee/createWithArray
6. Update an employee based on name -http://localhost:8081/employee/{employeeName}
7. Delete an employee by name- [http://localhost:8081/deleteEmployee/{name}](http://localhost:8081/deleteEmployee/%7bname%7d)

Project description:-

EmployeeOperations is an application which is designed with the help of a spring boot framework, this application enables us to do the following operations. The above mentioned rest endpoints were created to help with performing below operations

1. GetAllEmployees
2. Get employee by a specific name
3. Create a new employee
4. Create list of employees
5. Create an array of employees
6. Update an employee based on name
7. Delete an employee by name

Framework details:-

1. The project with below details, was created from the spring boot initializr

<groupId>com.org</groupId>

<artifactId>employeeOpertions</artifactId>

<version>0.0.1-SNAPSHOT</version>

<name>employeeOpertions</name>

<description>Demo project for Spring Boot</description>

1. Selected the dependencies Wed, actuator, devtools and spring security to help with developing the business logic as per the assignment
2. Created a package com.org.employeeOperations and then created a employeeConroller class which basically has the endpoints mapping methods
3. In the same package I have a class with the name “employeeController” which was created by spring boot framework, this is where the application starts from.
4. I’ve created another package with the name “com.org.employeeOperations.entities” in this package a new class called “EmloyeeModel.java” was created and the all the associated employee data would be placed.

* This class was primarily having a constructor to initialize the values.
* Implemented getters and setters for all the fields.

1. Have created another package with the name “com.org.employeeOperations.services” in this package a new class called “EmployeeService.java” is created and this class is responsible for the business logic that has to be performed when a particular endpoint is hit.

Application start up details:-

This application is locally hosted and uses the port 8081, since the port 8080 is occupied by Jenkins in my local.

**Feature:** verification of all employee related

**Scenario:** to query for all employees

Given: i have the endpoint

When: i hit the endpoint with get operation

Then: i should be able to get the details of all employees

**Scenario:** to query a particular employee by name

Given: i have the endpoint

When: i hit the endpoint with get operation by passing name as path param

Then: i should be able to get the detasils of all employees

**Scenario:** to validate the functionality of creating a new employee

Given: i have the baseURI and the appliction is running

When: i hit the endpoint by passing requet body and post method

Then: i should be able to verfiy the successmessage

**Scenario:** to validate the functionality of creating a list new employees

Given: i have the baseURI and the application is running

When: i hit the endpoint by passing requet body contains the list of employee details and post method

Then: i should be able to verfiy the response body

**Scenario:** to validate the functionality of creating an array new employees

Given: i have the baseURI and the application is running

When: i hit the endpoint by passing requet body contains the array of employee details and post method

Then: i should be able to verfiy the response body

**Scenario:** to validate the functionality of updating a new employee

Given: i have the baseURI and the appliction is running

When: i hit the endpoint by passing requet body and the path param with update method

Then: i should be able to verfiy the successmessage

**Scenario:** to validate the functionality of deleting an employee

Given: i have the baseURI and the appliction is running

When: i hit the endpoint with path param with delete method

Then: i should be able to verfiy the successmessage