Below are examples of how to test each endpoint using Postman. Replace {id} with actual IDs from your database.

**A. Regions**

1. **Create a Region**
   * **Method**: POST
   * **URL**: http://localhost:8080/api/regions
   * **Body (Raw - JSON)**:

json

Copy

{

"name": "North America"

}

* + **Response**: The created region with an ID.

1. **Get All Regions**
   * **Method**: GET
   * **URL**: http://localhost:8080/api/regions
   * **Response**: List of all regions.
2. **Get Region by ID**
   * **Method**: GET
   * **URL**: http://localhost:8080/api/regions/{id}
   * **Response**: The region with the specified ID.
3. **Update a Region**
   * **Method**: PUT
   * **URL**: http://localhost:8080/api/regions/{id}
   * **Body (Raw - JSON)**:

json

Copy

{

"name": "Europe"

}

* + **Response**: The updated region.

1. **Delete a Region**
   * **Method**: DELETE
   * **URL**: http://localhost:8080/api/regions/{id}
   * **Response**: HTTP 204 (No Content).

**B. Countries**

1. **Create a Country**
   * **Method**: POST
   * **URL**: http://localhost:8080/api/countries
   * **Body (Raw - JSON)**:

json

Copy

{

"name": "United States",

"region": {

"id": 1

}

}

* + **Response**: The created country with an ID.

1. **Get All Countries**
   * **Method**: GET
   * **URL**: http://localhost:8080/api/countries
   * **Response**: List of all countries.
2. **Get Country by ID**
   * **Method**: GET
   * **URL**: http://localhost:8080/api/countries/{id}
   * **Response**: The country with the specified ID.
3. **Update a Country**
   * **Method**: PUT
   * **URL**: http://localhost:8080/api/countries/{id}
   * **Body (Raw - JSON)**:

json

Copy

{

"name": "Canada",

"region": {

"id": 1

}

}

* + **Response**: The updated country.

1. **Delete a Country**
   * **Method**: DELETE
   * **URL**: http://localhost:8080/api/countries/{id}
   * **Response**: HTTP 204 (No Content).

**C. Locations**

1. **Create a Location**
   * **Method**: POST
   * **URL**: http://localhost:8080/api/locations
   * **Body (Raw - JSON)**:

json

Copy

{

"streetAddress": "123 Main St",

"postalCode": "12345",

"city": "New York",

"stateProvince": "NY",

"country": {

"id": 1

}

}

* + **Response**: The created location with an ID.

1. **Get All Locations**
   * **Method**: GET
   * **URL**: http://localhost:8080/api/locations
   * **Response**: List of all locations.
2. **Get Location by ID**
   * **Method**: GET
   * **URL**: http://localhost:8080/api/locations/{id}
   * **Response**: The location with the specified ID.
3. **Update a Location**
   * **Method**: PUT
   * **URL**: http://localhost:8080/api/locations/{id}
   * **Body (Raw - JSON)**:

json

Copy

{

"streetAddress": "456 Elm St",

"postalCode": "67890",

"city": "Los Angeles",

"stateProvince": "CA",

"country": {

"id": 1

}

}

* + **Response**: The updated location.

1. **Delete a Location**
   * **Method**: DELETE
   * **URL**: http://localhost:8080/api/locations/{id}
   * **Response**: HTTP 204 (No Content).

**D. Departments**

1. **Create a Department**
   * **Method**: POST
   * **URL**: http://localhost:8080/api/departments
   * **Body (Raw - JSON)**:

json

Copy

{

"name": "HR",

"location": {

"id": 1

}

}

* + **Response**: The created department with an ID.

1. **Get All Departments**
   * **Method**: GET
   * **URL**: http://localhost:8080/api/departments
   * **Response**: List of all departments.
2. **Get Department by ID**
   * **Method**: GET
   * **URL**: http://localhost:8080/api/departments/{id}
   * **Response**: The department with the specified ID.
3. **Update a Department**
   * **Method**: PUT
   * **URL**: http://localhost:8080/api/departments/{id}
   * **Body (Raw - JSON)**:

json

Copy

{

"name": "Finance",

"location": {

"id": 1

}

}

* + **Response**: The updated department.

1. **Delete a Department**
   * **Method**: DELETE
   * **URL**: http://localhost:8080/api/departments/{id}
   * **Response**: HTTP 204 (No Content).

**E. Jobs**

1. **Create a Job**
   * **Method**: POST
   * **URL**: http://localhost:8080/api/jobs
   * **Body (Raw - JSON)**:

json

Copy

{

"title": "Software Engineer",

"minSalary": 50000,

"maxSalary": 100000

}

* + **Response**: The created job with an ID.

1. **Get All Jobs**
   * **Method**: GET
   * **URL**: http://localhost:8080/api/jobs
   * **Response**: List of all jobs.
2. **Get Job by ID**
   * **Method**: GET
   * **URL**: http://localhost:8080/api/jobs/{id}
   * **Response**: The job with the specified ID.
3. **Update a Job**
   * **Method**: PUT
   * **URL**: http://localhost:8080/api/jobs/{id}
   * **Body (Raw - JSON)**:

json

Copy

{

"title": "Senior Software Engineer",

"minSalary": 70000,

"maxSalary": 120000

}

* + **Response**: The updated job.

1. **Delete a Job**
   * **Method**: DELETE
   * **URL**: http://localhost:8080/api/jobs/{id}
   * **Response**: HTTP 204 (No Content).

**F. Employees**

1. **Create an Employee**
   * **Method**: POST
   * **URL**: http://localhost:8080/api/employees
   * **Body (Raw - JSON)**:

json

Copy

{

"firstName": "John",

"lastName": "Doe",

"email": "john.doe@example.com",

"phoneNumber": "123-456-7890",

"salary": 60000,

"job": {

"id": 1

},

"department": {

"id": 1

}

}

* + **Response**: The created employee with an ID.

1. **Get All Employees**
   * **Method**: GET
   * **URL**: http://localhost:8080/api/employees
   * **Response**: List of all employees.
2. **Get Employee by ID**
   * **Method**: GET
   * **URL**: http://localhost:8080/api/employees/{id}
   * **Response**: The employee with the specified ID.
3. **Update an Employee**
   * **Method**: PUT
   * **URL**: http://localhost:8080/api/employees/{id}
   * **Body (Raw - JSON)**:

json

Copy

{

"firstName": "Jane",

"lastName": "Doe",

"email": "jane.doe@example.com",

"phoneNumber": "987-654-3210",

"salary": 65000,

"job": {

"id": 1

},

"department": {

"id": 1

}

}

* + **Response**: The updated employee.

1. **Delete an Employee**
   * **Method**: DELETE
   * **URL**: http://localhost:8080/api/employees/{id}
   * **Response**: HTTP 204 (No Content).

**G. Job Histories**

1. **Create a Job History**
   * **Method**: POST
   * **URL**: http://localhost:8080/api/job-histories
   * **Body (Raw - JSON)**:

json

Copy

{

"startDate": "2023-01-01",

"endDate": "2023-12-31",

"employee": {

"id": 1

},

"job": {

"id": 1

},

"department": {

"id": 1

}

}

* + **Response**: The created job history with an ID.

1. **Get All Job Histories**
   * **Method**: GET
   * **URL**: http://localhost:8080/api/job-histories
   * **Response**: List of all job histories.
2. **Get Job History by ID**
   * **Method**: GET
   * **URL**: http://localhost:8080/api/job-histories/{id}
   * **Response**: The job history with the specified ID.
3. **Update a Job History**
   * **Method**: PUT
   * **URL**: http://localhost:8080/api/job-histories/{id}
   * **Body (Raw - JSON)**:

json

Copy

{

"startDate": "2023-06-01",

"endDate": "2023-12-31",

"employee": {

"id": 1

},

"job": {

"id": 1

},

"department": {

"id": 1

}

}

* + **Response**: The updated job history.

1. **Delete a Job History**
   * **Method**: DELETE
   * **URL**: http://localhost:8080/api/job-histories/{id}
   * **Response**: HTTP 204 (No Content).

**4. Verify Responses**

* Check the **Status Code** (e.g., 200 for success, 201 for created, 204 for deleted).
* Verify the **Response Body** for the expected data.

**5. Save Requests in Postman**

* Save each request in a Postman collection for future use.
* Use environments in Postman to manage different environments (e.g., local, staging, production).