**1. Get All Admins**

**Request Type**: GET  
**Endpoint**: <http://localhost:8085/admins>

**Test Case**:

* **Description**: Retrieve a list of all admins.
* **Expected Response**:
  + **Status**: 200 OK
  + **Body** (example):

json

[

{

"id": "1",

"username": "admin1",

"email": "admin1@example.com"

},

{

"id": "2",

"username": "admin2",

"email": "admin2@example.com"

}

]

**2. Get Admin by ID**

**Request Type**: GET  
**Endpoint**: [http://localhost:8085/admins/{id}](http://localhost:8085/admins/%7bid%7d)

**Test Case**:

* **Description**: Retrieve details of a specific admin by ID.
* **Example ID**: 1
* **Expected Response**:
  + **Status**: 200 OK
  + **Body** (example):

json

{

"id": "1",

"username": "admin1",

"email": "admin1@example.com"

}

**3. Insert Admin into Database**

**Request Type**: POST  
**Endpoint**: http://localhost:8085/admins  
**Test Case**:

* **Description**: Insert a new admin into the database.
* **Request Body**:

json

{

"username": "admin",

"email": "admin1@example.com",

"password": "adminpass"

}

* **Expected Response**:
  + **Status**: 200
  + **Body** (example):

{

"id": "1",

"username": "admin",

"email": "admin1@example.com"

}

}

**4. Update Existing Admin**

**Request Type**: PUT  
**Endpoint**: http://localhost:8085/admins/{id}  
**Test Case**:

* **Description**: Update the details of an existing admin.
* **Example ID**: 1
* **Request Body**:

json

{

"username": "adminupdated",

"email": "adminupdated@example.com",

"password": "adminpassupdated"

}

* **Expected Response**:
  + **Status**: 200 OK
  + **Body** (example):

json

{

"id": "1",

"username": "adminupdated",

"email": "adminupdated@example.com"

}

**5. Delete Existing Admin**

**Request Type**: DELETE  
**Endpoint**: [http://localhost:8085/admins/{id}](http://localhost:8085/admins/%7bid%7d)

**Test Case**:

* **Description**: Delete an admin from the database.
* **Example ID**: 1
* **Expected Response**:
  + **Status**: 204 No content

**NEGATIVE TEST CASES**

**1. Get Admin by Invalid ID**

**Request Type:** GET  
**Endpoint:** http://localhost:8085/admins/invalid-id  
**Test Case:**

* **Description:** Attempt to retrieve an admin with an invalid or non-existent ID.
* **Expected Response:**
  + **Status:** 404 Not Found
  + **Body (example):**

{

"timestamp": "2024-09-12T11:55:40.621+00:00",

"status": 404,

"error": "Not Found",

"trace": "com.ecommerce.admin.exception.ResourceNotFoundException: Admin not found with id: 99 (TaskThread.java:63)\r\n\tat java.base/java.lang.Thread.run(Thread.java:1583)\r\n",

"message": "Admin not found with id: 99",

"path": "/admins/99"

}

**1. DELETE Admin by Invalid ID**

**Request Type:** DELETE  
**Endpoint:** http://localhost:8085/admins/invalid-id  
**Test Case:**

* **Description:** Attempt to DELETE an admin with an invalid or non-existent ID.
* **Expected Response:**
  + **Status:** 404 Not Found
  + **Body (example):**

{

"timestamp": "2024-09-12T12:00:48.575+00:00",

    "status": 404,

    "error": "Not Found",

    "trace": "com.ecommerce.admin.exception.ResourceNotFoundException: Admin not found with id: 99 (TaskThread.java:63)\r\n\tat java.base/java.lang.Thread.run(Thread.java:1583)\r\n",

"message": "Admin not found with id: 99",

"path": "/admins/99"

}