

Name: API | GET /Employees/{id} returns 200 OK for nonexistent employee.

Description: When sending a GET request to /Prod/api/Employees/{id} with a valid UUID that doesn't match any existing employee, the API returns a 200 OK status with an empty body instead of 404 Not Found.

Setup:

- API version: 1.0.0 (assumed)
- Environment: testing
- Authorization: Basic VGVzdFVzZXI3NTc6TyFoRXxaSmdSXTNY
- Nonexistent employee id: 83e2574b-9241-4d47-8620-8b30a2e374bb

Evidence:

The screenshot shows a REST client interface with the following details:

- Method:** GET
- URL:** `https://wmxrww14uc.execute-api.us-east-1.amazonaws.com/Prod/api/Employees/83e2574b-9241-4d47-8620-8b30a2e374bb`
- Headers:** Content-Type: application/json
- Authorization:** Basic VGVzdFVzZXI3NTc6TyFoRXxaSmdSXTNY
- Response:** 200 OK, 257 ms
- Body:** Raw view showing an empty response.

NOTE: It also happened with DELETE request to `https://wmxrww14uc.execute-api.us-east-1.amazonaws.com/Prod/api/Employees/{nonexistent-id}`

The screenshot shows a REST client interface with the following details:

- Method:** DELETE
- URL:** `https://wmxrww14uc.execute-api.us-east-1.amazonaws.com/Prod/api/Employees/8e26b431-cd86-45d9-a361-26c75a4ae7c9`
- Body:** raw
- Response:** 200 OK, 814 ms
- Body:** Raw view showing an empty response.

Repeatability: Reproducible.

Steps to reproduce:

1. Make GET request to <https://wmxrwwq14uc.execute-api.us-east-1.amazonaws.com/Prod/api/Employees/{nonexistent-id}> with nonexistent employee id.

Actual results: Endpoint returns status code 200 and blank response body.

Expected results: Endpoint should return status code 404 not found and a message inside response body.

Name: API | POST /Prod/api/Employees accept special characters in name fields.

Description: When submitting a POST request to <https://wmxrww14uc.execute-api.us-east-1.amazonaws.com/Prod/api/Employees> with special characters in the firstName or lastName fields (e.g., "Test\$#"), the API accepts the input and returns 200 OK, allowing the record to be saved. E.g body:

```
{  
  "firstName": "Test$#",  
  "lastName": "Testing3",  
  "dependants": 1  
}
```

Setup:

- API version: 1.0.0 (assumed)
- Environment: testing
- Authorization: Basic VGVzdFVzZXI3NTc6TyFoRXxaSmdSXTNY

Evidence:

The screenshot displays a REST client interface with the following details:

- Method:** POST
- URL:** <https://wmxrww14uc.execute-api.us-east-1.amazonaws.com/Prod/api/Employees/>
- Body Type:** raw
- Request Body (JSON):**

```
1 {  
2   "firstName": "Test$#",  
3   "lastName": "Testing3",  
4   "dependants": 1  
5 }
```
- Status:** 200 OK (873 ms)
- Response Body (JSON):**

```
1 {  
2   "partitionKey": "TestUser757",  
3   "sortKey": "4f4ee75e-01d0-4932-90a7-c5f13006da29",  
4   "username": "TestUser757",  
5   "id": "4f4ee75e-01d0-4932-90a7-c5f13006da29",  
6   "firstName": "Test$#",  
7   "lastName": "Testing3",  
8   "dependants": 1,  
9   "salary": 52000,  
10  "gross": 2000,  
11  "benefitsCost": 57.69231,  
12  "net": 1942.3077  
13 }
```

Repeatability: Reproducible.

Steps to reproduce:

1. Make a POST request to <https://wmxrwwq14uc.execute-api.us-east-1.amazonaws.com/Prod/api/Employees> using a body with special characters inputted in First name field.

Actual results: Endpoint returns status code 200 and allows you to add the new employee.

Expected results: Endpoint should return status code of 400 Bad Request and mention that special characters are not allowed.

Name: API | POST /Prod/api/Employees accepts payloads with unexpected fields.

Description: When sending a POST request to <https://wmxrww14uc.execute-api.us-east-1.amazonaws.com/Prod/api/Employees> that includes an undefined property (e.g., "extra": "value"), the API still returns 200 OK and adds the new employee, despite the schema specifying **"additionalProperties": false**. E.g body:

```
{
  "firstName": "Test5",
  "lastName": "Testing5",
  "dependants": 1,
  "extra": "value"
}
```

Setup:

- API version: 1.0.0 (assumed)
- Environment: testing
- Authorization: Basic VGVzdFVzZXI3NTc6TyFoRXxaSmdSXTNY

Evidence:

The screenshot shows a REST client interface with the following details:

- Method:** POST
- URL:** <https://wmxrww14uc.execute-api.us-east-1.amazonaws.com/Prod/api/Employees/>
- Body Type:** raw
- Request Body (JSON):**

```
1 {
2   "firstName": "Test5",
3   "lastName": "Testing5",
4   "dependants": 1,
5   "extra": "value"
6 }
```
- Status:** 200 OK
- Response Body (JSON):**

```
1 {
2   "partitionKey": "TestUser757",
3   "sortKey": "0e57d46c-1fac-4aa8-b2ea-8b460268e54e",
4   "username": "TestUser757",
5   "id": "0e57d46c-1fac-4aa8-b2ea-8b460268e54e",
6   "firstName": "Test5",
7   "lastName": "Testing5",
8   "dependants": 1,
9   "salary": 52000,
10  "gross": 2000,
11  "benefitsCost": 57.69231,
12  "net": 1942.3077
13 }
```

Repeatability: Reproducible.

Steps to reproduce:

1. Make a POST request to <https://wmxrwwq14uc.execute-api.us-east-1.amazonaws.com/Prod/api/Employees> using a body with an unexpected field.

Actual results: Endpoint returns status code 200 and allows you to add the new employee.

Expected results: Endpoint should return status code 400 Bad Request and mention that unexpected fields are not allowed.

Name: API | PUT /Prod/api/Employees returns 405 Method Not Allowed instead of 400 for invalid data type.

Description: When submitting a PUT request to <https://wmxrww14uc.execute-api.us-east-1.amazonaws.com/Prod/api/Employees> with an invalid data type in the dependants field (e.g., "dependants": "one" instead of an integer), the API returns 405 Method Not Allowed instead of 400 Bad Request. E.g body:

```
{  
  "id": "0e57d46c-1fac-4aa8-b2ea-8b460268e54e",  
  "firstName": "Test5",  
  "lastName": "Testing55",  
  "dependants": "one"  
}
```

Setup:

- API version: 1.0.0 (assumed)
- Environment: testing
- Authorization: Basic VGVzdGVzZXI3NTc6TyFoRXxaSmdSXTNY

Evidence:

The screenshot shows a REST client interface with the following details:

- Method:** PUT
- URL:** <https://wmxrww14uc.execute-api.us-east-1.amazonaws.com/Prod/api/Employees/>
- Body Type:** raw (selected)
- Body Content:**

```
1 {  
2   "id": "0e57d46c-1fac-4aa8-b2ea-8b460268e54e",  
3   "firstName": "Test5",  
4   "lastName": "Testing55",  
5   "dependants": "one"  
6 }
```
- Response:** 405 Method Not Allowed (322 ms)
- Raw View:** 1

Repeatability: Reproducible.

Steps to reproduce:

1. Make a PUT request to <https://wmxrwwq14uc.execute-api.us-east-1.amazonaws.com/Prod/api/Employees> using a body with dependants field value as string.

Actual results: Endpoint returns status code 405 Method Not Allowed and blank response body.

Expected results: Endpoint should return status code 400 Bad Request and a message inside response body.

Name: API | PUT /Prod/api/Employees returns wrong values for salary, gross and net after updated employee.

Description: When submitting a PUT request to <https://wmxrww14uc.execute-api.us-east-1.amazonaws.com/Prod/api/Employees> with a valid data, the fields salary, gross and net had wrong values than the expected ones.

Setup:

- API version: 1.0.0 (assumed)
- Environment: testing
- Authorization: Basic VGVzdFVzZXI3NTc6TyFoRXxaSmdSXTNY
- Id employee: 8f70a667-d068-4ad7-aec7-82ce474f1946

Evidence:

The screenshot displays a REST client interface with a PUT request to the endpoint `https://wmxrww14uc.execute-api.us-east-1.amazonaws.com/Prod/api/Employees/`. The request body is a JSON object with the following fields: `id`, `firstName`, `lastName`, and `dependants`. The response status is `200 OK`, and the response body is a JSON object containing `sortKey`, `username`, `id`, `firstName`, `lastName`, `dependants`, `salary`, `gross`, `benefitsCost`, and `net`. The `salary`, `gross`, and `net` fields show values that appear to be incorrectly calculated or formatted.

```
1 {
2   "id": "8f70a667-d068-4ad7-aec7-82ce474f1946",
3   "firstName": "Test",
4   "lastName": "Testing",
5   "dependants": 1
6 }
7
```

Body Cookies Headers (12) Test Results 200 OK

```
13 {
14   "sortKey": "f5ae4db8-62f6-40b4-a4e3-19a299cbdc91",
15   "username": "TestUser757",
16   "id": "f5ae4db8-62f6-40b4-a4e3-19a299cbdc91",
17   "firstName": "Test",
18   "lastName": "Testing",
19   "dependants": 1,
20   "salary": 0,
21   "gross": 0,
22   "benefitsCost": 57.69231,
23   "net": -57.69231
24 }
```

Repeatability: Reproducible.

Steps to reproduce:

1. Make a PUT request to <https://wmxrwwq14uc.execute-api.us-east-1.amazonaws.com/Prod/api/Employees> using a updated body with different values.

Actual results: Endpoint returns status code 200, however the value of salary, gross and net shows different values than the expected ones.

Expected results: Endpoint should return status code 200 and correct updated data for the employee.