

PRUEBAS UNITARIAS

PROCESO DE RECLUTAMIENTO: 14042021-2215

Q2 Desarrollador Back Senior

Postulante: David Culebra López

INTRODUCCION

En esta documentación, realizaremos todas las pruebas unitarias realizadas, para que el usuario sepa que esta introduciendo mal los datos (parámetros).

Nota: Utilizaremos el programa **Postman**

PRUEBAS INSERTAR UN REGISTRO DYNAMODB

Datos:

- <https://ogahdcddb3.execute-api.us-east-1.amazonaws.com/dev/insertar>
- Endpoint POST

Pruebas:

- Si no enviamos ningún parámetro, el resultado es el siguiente:



The screenshot shows the 'Body' tab of a Postman response. The response is in JSON format, displaying a message and parameters. The message states that the user must send the corresponding field to insert into the database, and the parameters are null.

```
1  {
2    "mensaje": "Para insertar en la BD debe enviar el campo correspondiente",
3    "parametros": null
4  }
```

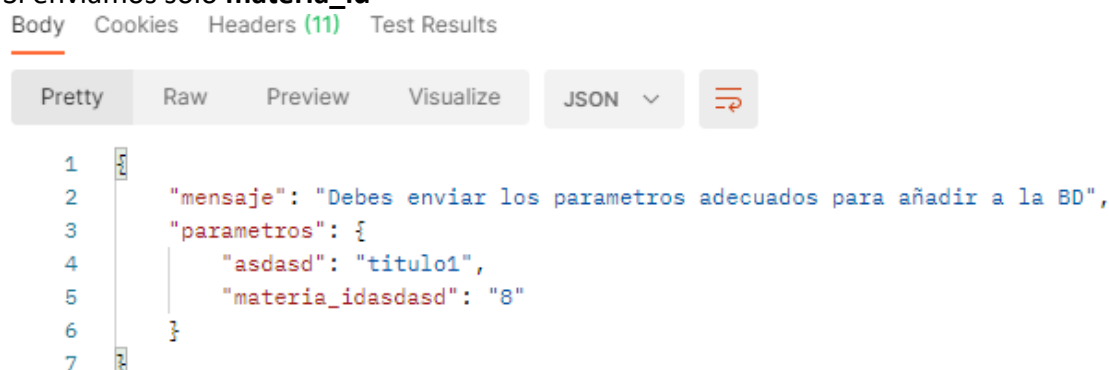
- Si enviamos otros parámetros:



The screenshot shows the 'Body' tab of a Postman response. The response is in JSON format, displaying a message and parameters. The message states that the user must send the appropriate parameters to add to the database, and the parameters are an object with 'asdasd' and 'materia_idasdasd' fields.

```
1  {
2    "mensaje": "Debes enviar los parametros adecuados para añadir a la BD",
3    "parametros": {
4      "asdasd": "titulo1",
5      "materia_idasdasd": "8"
6    }
7  }
```

- Si enviamos solo **materia_id**



The screenshot shows the 'Body' tab of a Postman response. The response is in JSON format, displaying a message and parameters. The message states that the user must send the appropriate parameters to add to the database, and the parameters are an object with 'asdasd' and 'materia_idasdasd' fields.

```
1  {
2    "mensaje": "Debes enviar los parametros adecuados para añadir a la BD",
3    "parametros": {
4      "asdasd": "titulo1",
5      "materia_idasdasd": "8"
6    }
7  }
```

- Si enviamos materia_id y titulo, pero materia_id es una cadena:

```
Body Cookies Headers (11) Test Results
Pretty Raw Preview Visualize JSON ↕
1
2  "mensaje": "materia_id no es numero entero",
3  "parametros": {
4    "materia_id": "eawds5",
5    "titulo": "titulo1"
6  }
7
```

- Si enviamos materia_id y titulo, vacíos:

```
Body Cookies Headers (11) Test Results
Pretty Raw Preview Visualize JSON ↕
1
2  "mensaje": "materia_id no es numero entero",
3  "parametros": {
4    "materia_id": "eawds5",
5    "titulo": "titulo1"
6  }
7
```

- Si agregamos una clave que ya este en la base de datos:

```
Body Cookies Headers (11) Test Results
Pretty Raw Preview Visualize JSON ↕
1
2  "mensaje": "Ya existe una materia_id con ese ID",
3  "parametros": {
4    "materia_id": "5",
5    "titulo": "asdasd"
6  }
7
```

- Si añadimos una nueva materia_id con su titulo, se añade correctamente:

```
Body Cookies Headers (11) Test Results
Pretty Raw Preview Visualize JSON ↕
1
2  "mensaje": "Registro Guardado Correctamente",
3  "parametros": {
4    "materia_id": "9",
5    "titulo": "Ciencias"
6  }
7
```

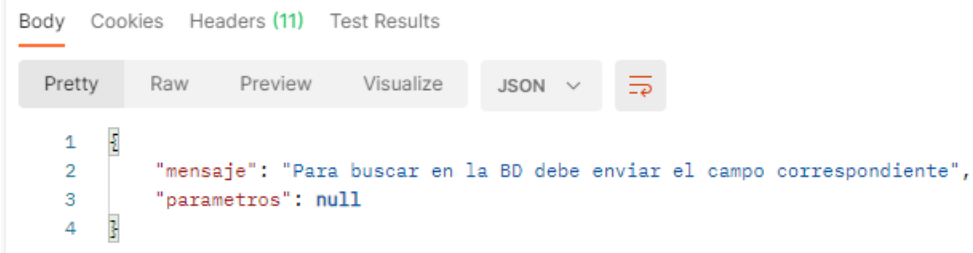
PRUEBAS VISUALIZAR UN REGISTRO DYNAMODB

Datos:

- <https://ogahdcdqb3.execute-api.us-east-1.amazonaws.com/dev/obtener>
- Endpoint GET

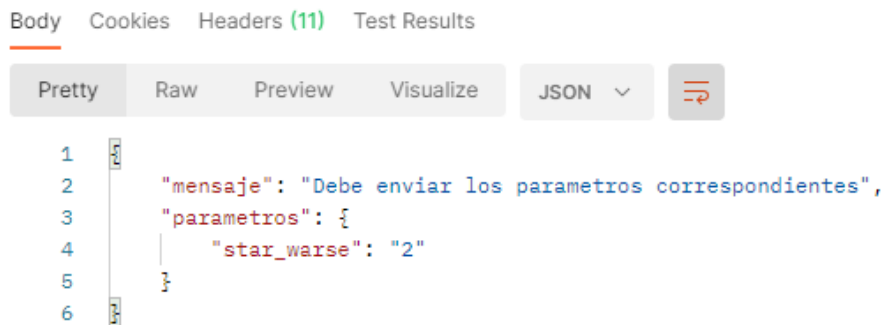
Pruebas:

- Si no enviamos nada



```
Body  Cookies  Headers (11)  Test Results
Pretty  Raw  Preview  Visualize  JSON  [icon]
1
2  "mensaje": "Para buscar en la BD debe enviar el campo correspondiente",
3  "parametros": null
4
```

- Si enviamos otro parámetro que no sea materia_id



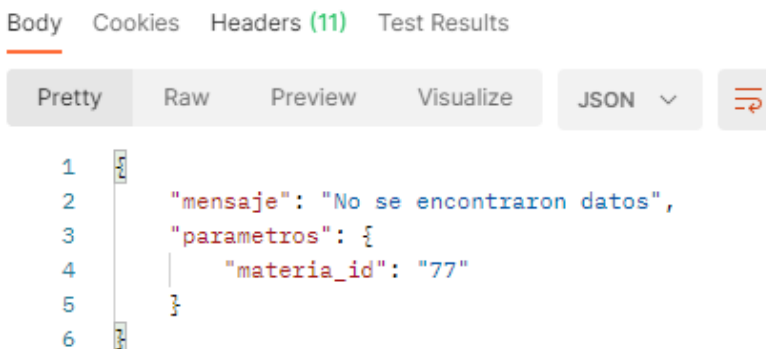
```
Body  Cookies  Headers (11)  Test Results
Pretty  Raw  Preview  Visualize  JSON  [icon]
1
2  "mensaje": "Debe enviar los parametros correspondientes",
3  "parametros": {
4    "star_warse": "2"
5  }
6
```

- Si enviamos materia_id, pero en cadena de texto



```
Body  Cookies  Headers (11)  Test Results
Pretty  Raw  Preview  Visualize  JSON  [icon]
1
2  "mensaje": "materia_id debe ser numero entero",
3  "parametros": {
4    "materia_id": "eseert"
5  }
6
```

- Si enviamos materia_id con un registro que no existe



```
Body  Cookies  Headers (11)  Test Results
Pretty  Raw  Preview  Visualize  JSON  [icon]
1
2  "mensaje": "No se encontraron datos",
3  "parametros": {
4    "materia_id": "77"
5  }
6
```

- Si enviamos materia_id con un registro que si existe

Body Cookies Headers (11) Test Results

Pretty Raw Preview Visualize JSON ▾

```

1  {
2    "mensaje": "Datos encontrados",
3    "parametros": {
4      "materia_id": "2"
5    },
6    "busqueda": {
7      "titulo": "titulo1",
8      "materia_id": 2
9    }
10 }

```

PRUEBAS VISUALIZAR UN REGISTRO SWAPI

Datos:

- <https://ogahdcdqb3.execute-api.us-east-1.amazonaws.com/dev/obtener>
- Endpoint GET

Pruebas:

- Todas las pruebas anteriores de **VISUALIZAR UN REGISTRO DYNAMODB**
- Si enviamos star_wars en cadena de texto

Body Cookies Headers (11) Test Results

Pretty Raw Preview Visualize JSON ▾

```

1  {
2    "mensaje": "star_wars debe ser numero entero",
3    "parametros": {
4      "star_wars": "hola"
5    }
6  }

```

- Si enviamos star_wars y no lo encuentra

Body Cookies Headers (11) Test Results

Pretty Raw Preview Visualize JSON ▾

```

1  {
2    "mensaje": "Star War NO encontrado",
3    "parametros": {
4      "star_wars": "50"
5    },
6    "busqueda": {
7      "detail": "Not found"
8    }
9  }

```

- Si enviamos star_wars y si lo encuentra

Body Cookies Headers (11) Test Results

Pretty

Raw

Preview

Visualize

JSON



```
1  {
2    "mensaje": "Star War encontrado",
3    "parametros": {
4      "star_wars": "5"
5    },
6    "busqueda": {
7      "name": "Sentinel-class landing craft",
8      "model": "Sentinel-class landing craft",
9      "manufacturer": "Sienar Fleet Systems, Cyngus Spaceworks",
10     "cost_in_credits": "240000",
11     "length": "38",
12     "max_atmosphering_speed": "1000",
13     "crew": "5",
14     "passengers": "75",
15     "cargo_capacity": "180000",
16     "consumables": "1 month",
17     "hvoerdrive rating": "1.0".
```