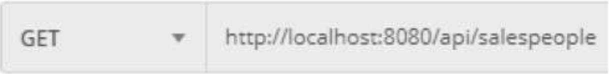
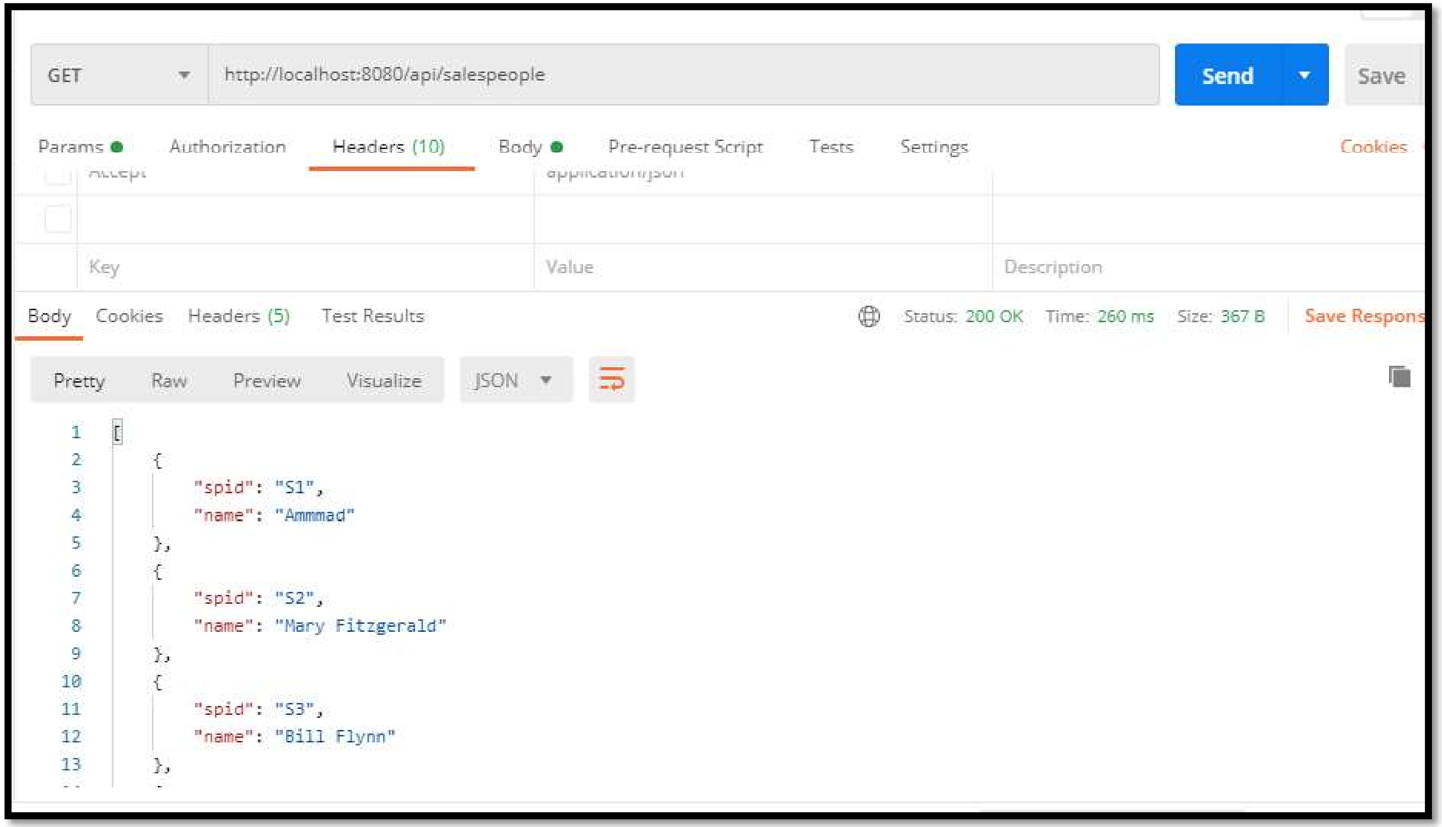
4.1.1 GET /api/salespeople

Returns an array of JSON objects containing the *spid* and *name* of all salespeople.





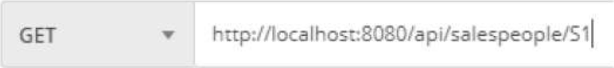
Status: 200 OK Time: 260 ms Size: 367 B

```
[
    {
        "spid": "S1",
        "name": "Ammmad"
    },
    {
        "spid": "S2",
        "name": "Mary Fitzgerald"
    },
    {
        "spid": "S3",
        "name": "Bill Flynn"
    },
```

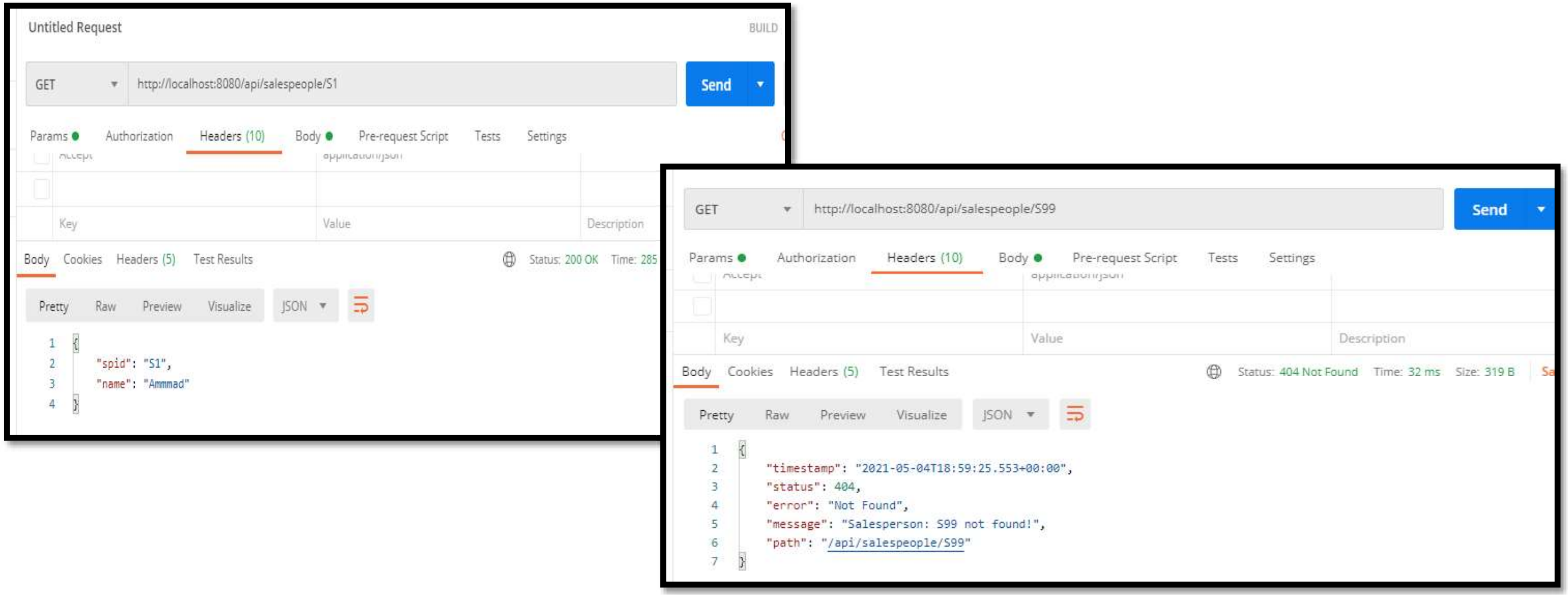
4.1.2 GET /api/salespeople/{spid}

Returns a JSON object containing the *spid* and *name* of the salesperson whose spid is equal to the spid in the URL.

If no salesperson with the specified spid exists, a HTTP 404 response should be returned with the error message: “Salesperson {spid} not found.” where {spid} is the spid in the URL.



```
http://localhost:8080/api/salespeople/S1
```



GET http://localhost:8080/api/salespeople/S1

Status: 200 OK Time: 285

```
{
    "spid": "S1",
    "name": "Ammmad"
}
```

GET http://localhost:8080/api/salespeople/S99

Status: 404 Not Found Time: 32 ms Size: 319 B

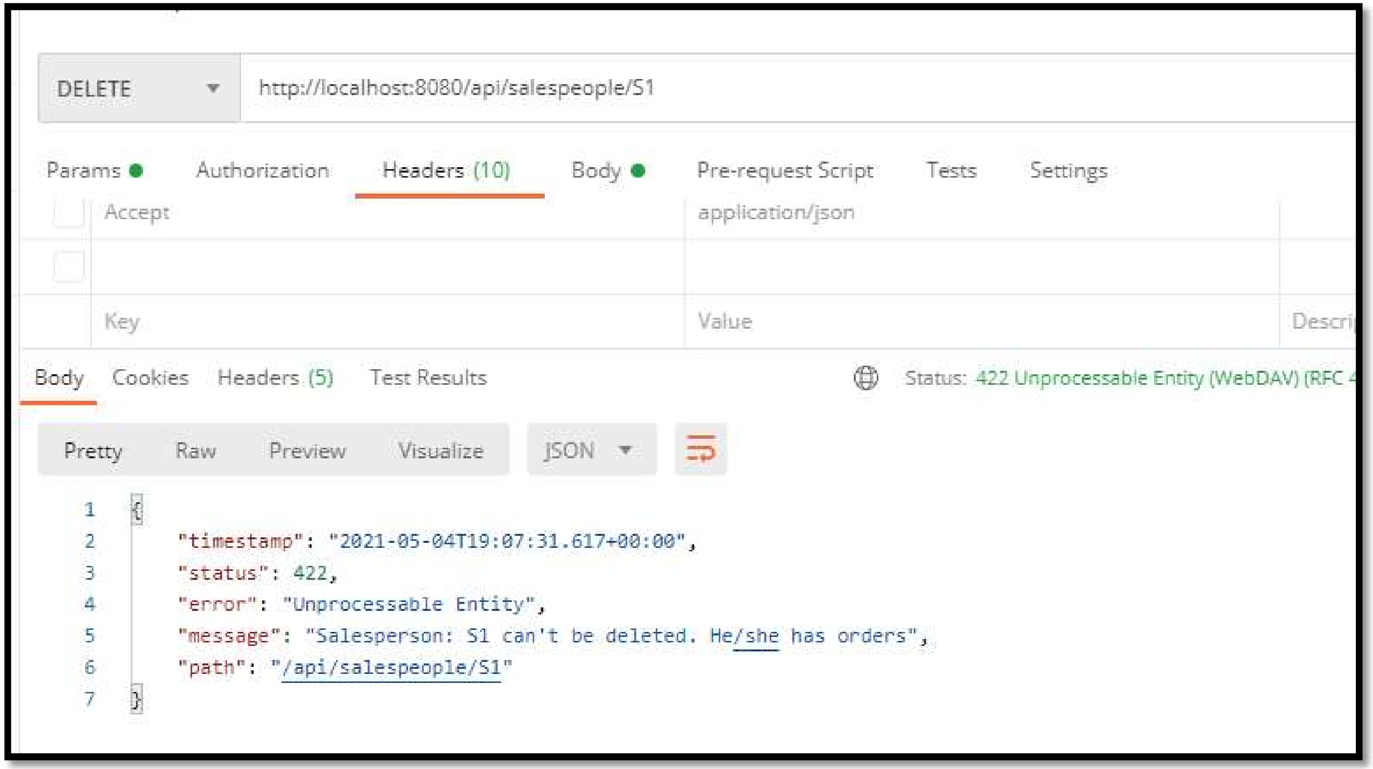
```
{
    "timestamp": "2021-05-04T18:59:25.553+00:00",
    "status": 404,
    "error": "Not Found",
    "message": "Salesperson: S99 not found!",
    "path": "/api/salespeople/S99"
}
```

4.1.3 DELETE /api/salespeople/{spid}

Deletes the salesperson whose *spid* is equal to the spid in the URL and returns a HTTP 200 response.

If no salesperson with the specified spid exists, a HTTP 200 response should still be returned.

If the salesperson with the specified spid exists, but has associated orders, a HTTP 422 response should be returned with the error message:
"salesperson {spid} can't be deleted. He/she has orders." where {spid} is the spid in the URL.



DELETE http://localhost:8080/api/salespeople/S1

Accept application/json

Status: 422 Unprocessable Entity (WebDAV) (RFC 4

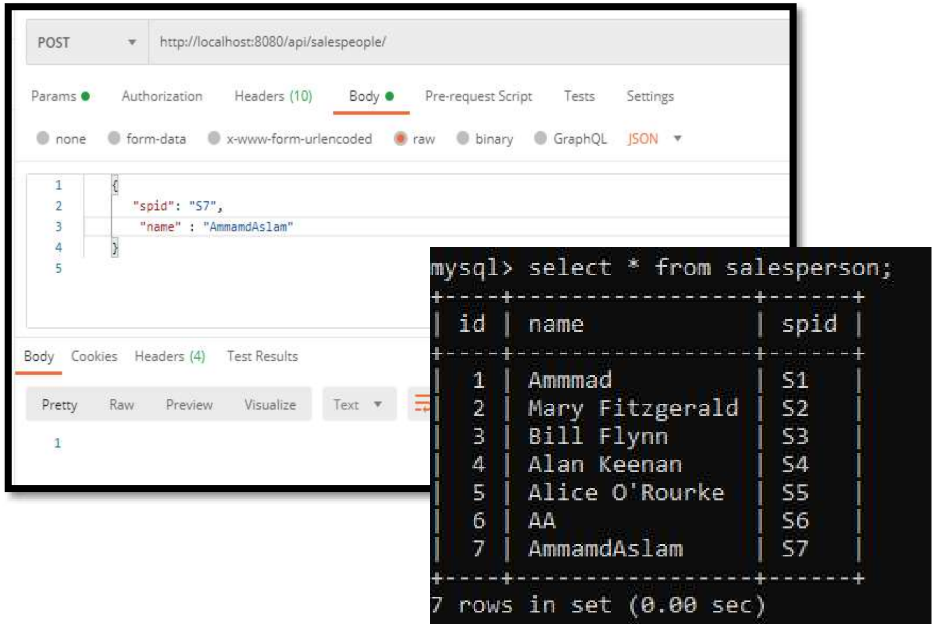
```
{
    "timestamp": "2021-05-04T19:07:31.617+00:00",
    "status": 422,
    "error": "Unprocessable Entity",
    "message": "Salesperson: S1 can't be deleted. He/she has orders",
    "path": "/api/salespeople/S1"
}
```

4.1.4 POST /api/salespeople

Persists the salesperson object specified in the request body to the database. The salesperson object must contain *spid* and *name*.

If either of these attributes are not present a HTTP 500 response should be returned with an appropriate error message.

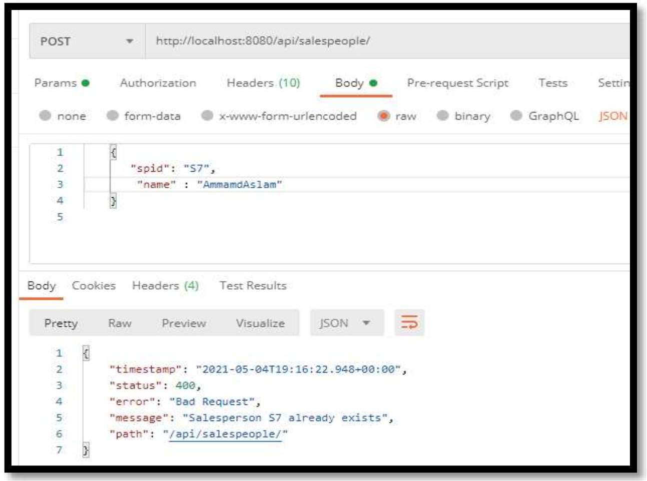
If a salesperson with the specified spid already exists, a HTTP 422 response should be returned with the following error message: “salesperson: {spid} already exists” where {spid} is the spid in the request body.



http://localhost:8080/api/salespeople/

```
{
  "spid": "S7",
   "name" : "AmmamdAslam"
}
```

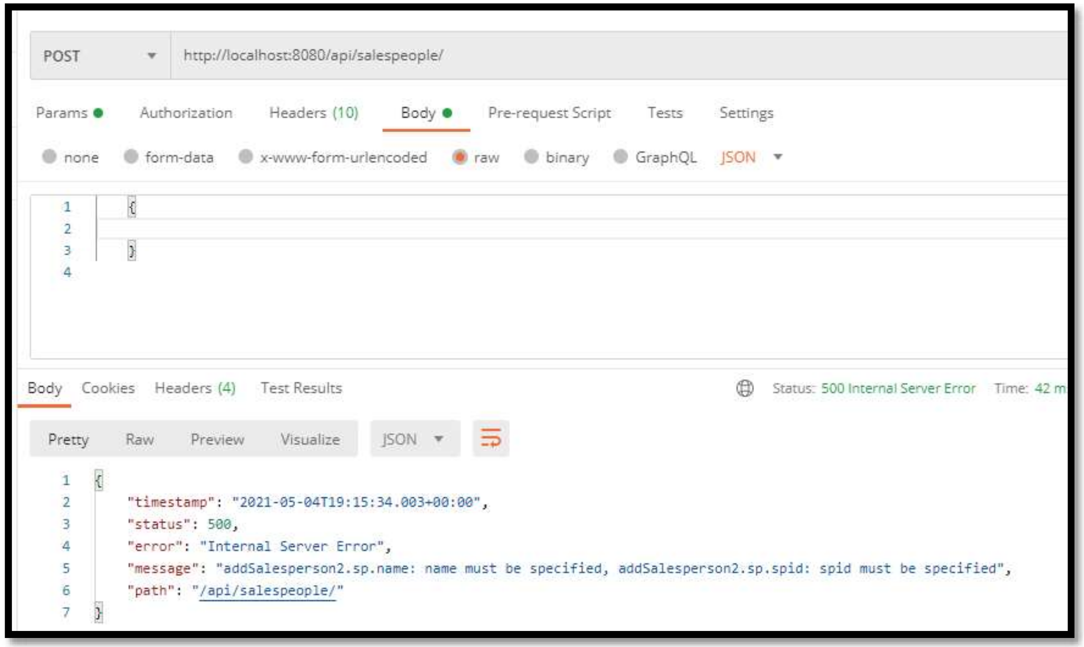
```
mysql> select * from salesperson;
+----+------------------+------+
| id | name             | spid |
+----+------------------+------+
|  1 | Ammmad           | S1   |
|  2 | Mary Fitzgerald  | S2   |
|  3 | Bill Flynn       | S3   |
|  4 | Alan Keenan      | S4   |
|  5 | Alice O'Rourke   | S5   |
|  6 | AA               | S6   |
|  7 | AmmamdAslam      | S7   |
+----+------------------+------+
7 rows in set (0.00 sec)
```



http://localhost:8080/api/salespeople/

```
{
  "spid": "S7",
   "name" : "AmmamdAslam"
}
```

```
{
    "timestamp": "2021-05-04T19:16:22.948+00:00",
    "status": 400,
    "error": "Bad Request",
    "message": "Salesperson S7 already exists",
    "path": "/api/salespeople/"
}
```



http://localhost:8080/api/salespeople/

```
{

}
```

Status: 500 Internal Server Error   Time: 42 m

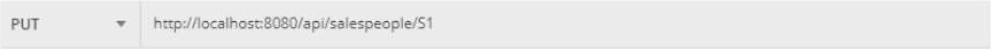
```
{
    "timestamp": "2021-05-04T19:15:34.003+00:00",
    "status": 500,
    "error": "Internal Server Error",
    "message": "addSalesperson2.sp.name: name must be specified, addSalesperson2.sp.spid: spid must be specified",
    "path": "/api/salespeople/"
}
```

4.1.5 PUT /api/salespeople/{spid}

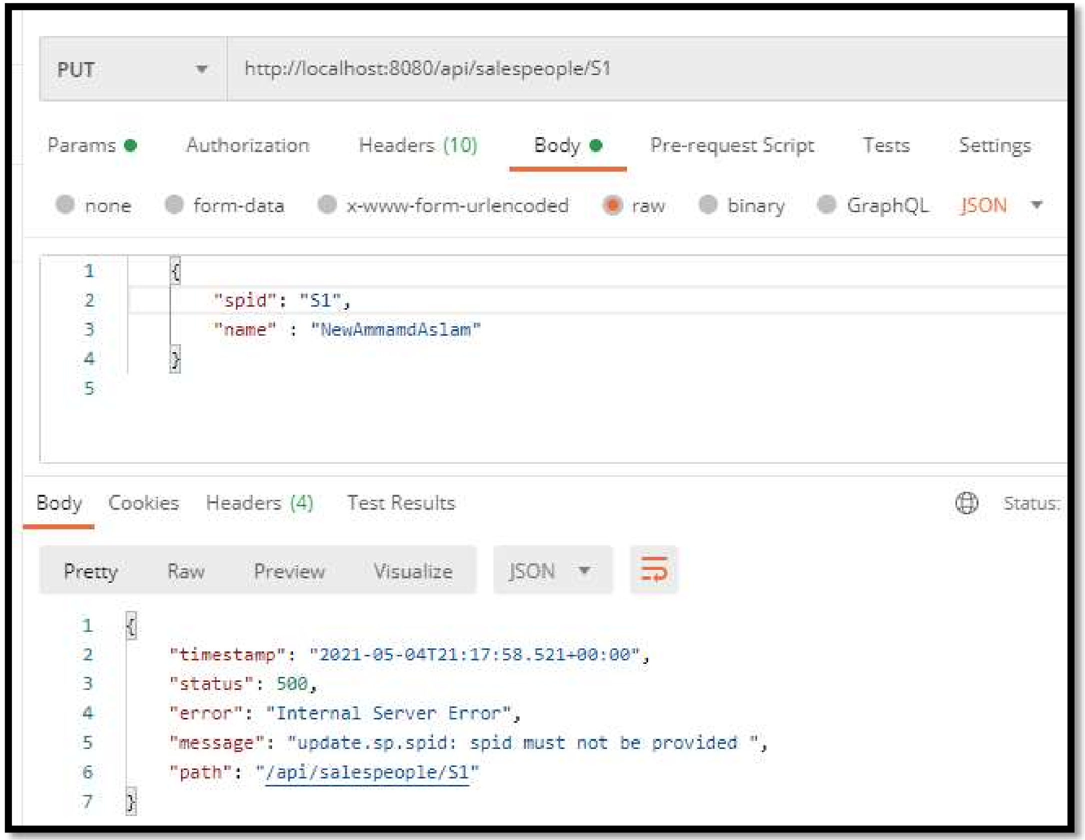
Updates the salesperson with the *spid* specified in the URL with the attributes specified in the request body. The request body should only contain the *name* attribute.

If any other attributes are present, a HTTP 500 response should be returned with an appropriate error message.

If no salesperson with the specified spid exists, a HTTP 404 response should be returned with the error message: “Salesperson {spid} not found.” where {spid} is the spid in the URL.



PUT http://localhost:8080/api/salespeople/S1

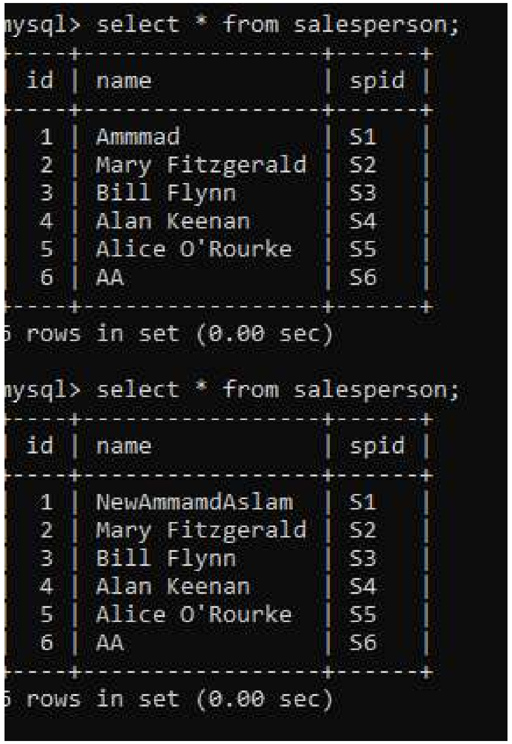


PUT http://localhost:8080/api/salespeople/S1

Params ●
Authorization
Headers (10)
Body ●
Pre-request Script
Tests
Settings
○ none ○ form-data ○ x-www-form-urlencoded ● raw ○ binary ○ GraphQL JSON

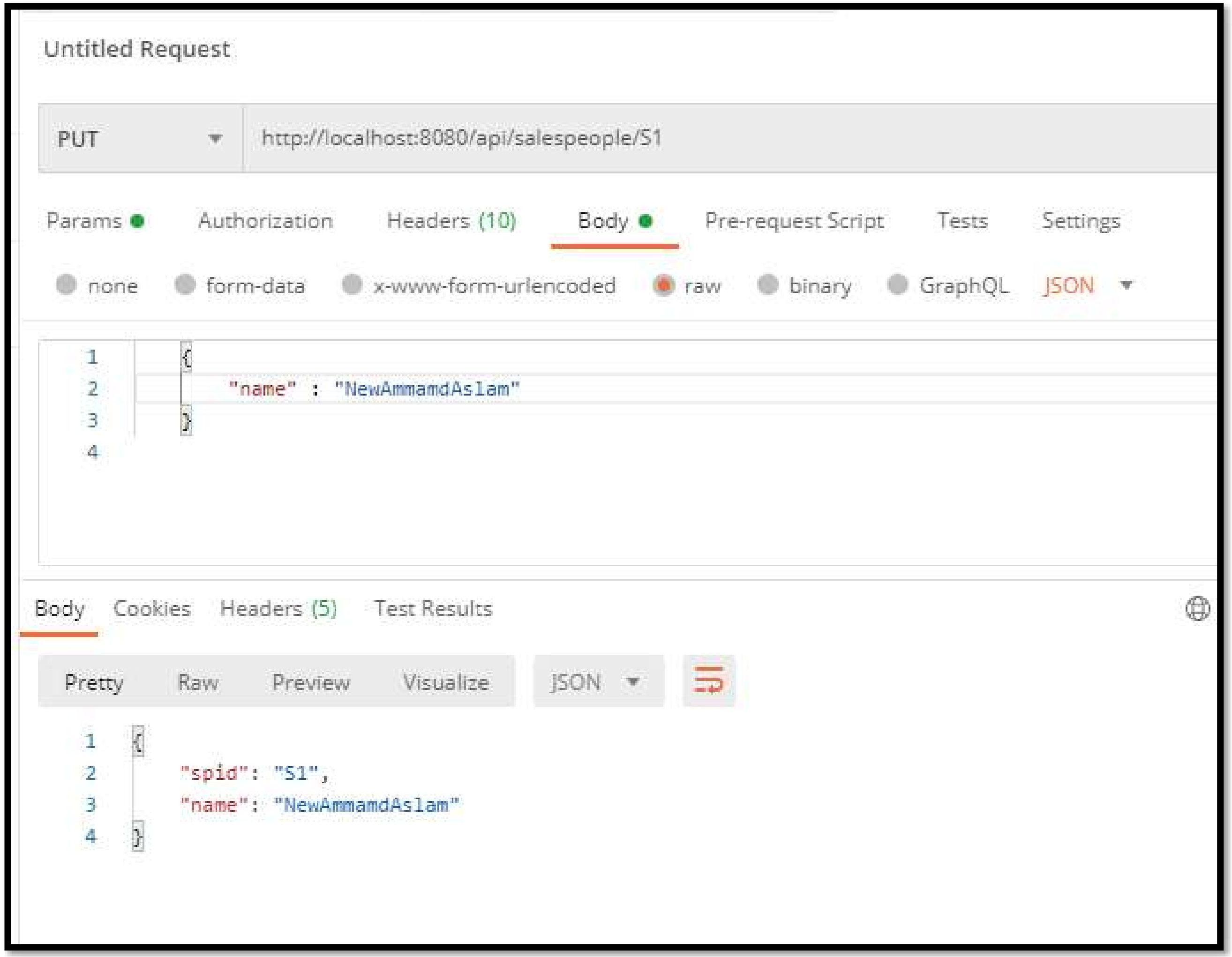
```
{
    "spid": "S1",
    "name" : "NewAmmamdAslam"
}
```

```
{
    "timestamp": "2021-05-04T21:17:58.521+00:00",
    "status": 500,
    "error": "Internal Server Error",
    "message": "update.sp.spid: spid must not be provided ",
    "path": "/api/salespeople/S1"
}
```



```
mysql> select * from salesperson;
+----+------------------+------+
| id | name             | spid |
+----+------------------+------+
|  1 | Ammmad           | S1   |
|  2 | Mary Fitzgerald  | S2   |
|  3 | Bill Flynn       | S3   |
|  4 | Alan Keenan      | S4   |
|  5 | Alice O'Rourke   | S5   |
|  6 | AA               | S6   |
+----+------------------+------+
6 rows in set (0.00 sec)

mysql> select * from salesperson;
+----+------------------+------+
| id | name             | spid |
+----+------------------+------+
|  1 | NewAmmamdAslam   | S1   |
|  2 | Mary Fitzgerald  | S2   |
|  3 | Bill Flynn       | S3   |
|  4 | Alan Keenan      | S4   |
|  5 | Alice O'Rourke   | S5   |
|  6 | AA               | S6   |
+----+------------------+------+
6 rows in set (0.00 sec)
```



PUT http://localhost:8080/api/salespeople/S1

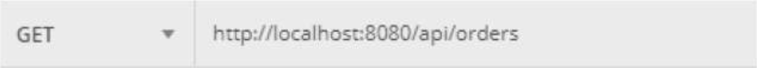
```
{
    "name" : "NewAmmamdAslam"
}
```

```
{
    "spid": "S1",
    "name": "NewAmmamdAslam"
}
```

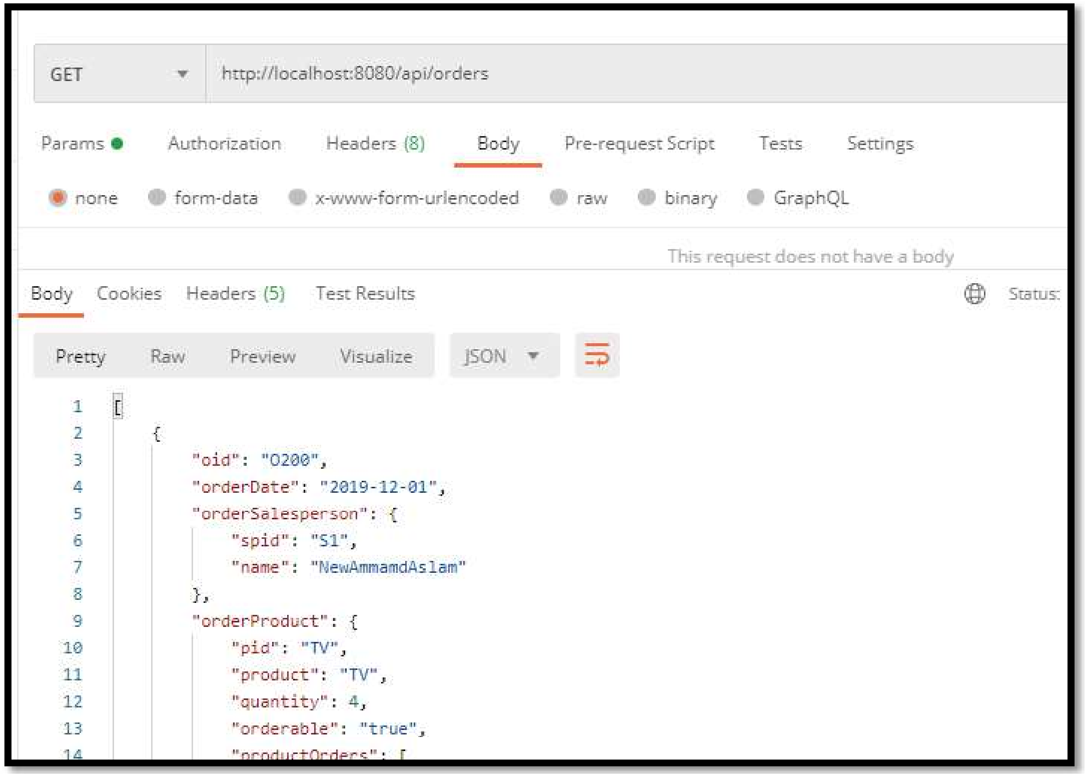
4.2 Order API endpoints

4.2.1 GET /api/orders

Returns an array of JSON objects containing the *oid*, *orderDate*, *orderSalesperson.spid*, *orderSalesperson.name*, *orderProduct.pid*, *orderProduct.product*, *orderProduct.quantity*, and *orderProduct.orderable* attributes of all orders.



GET http://localhost:8080/api/orders



GET http://localhost:8080/api/orders

This request does not have a body

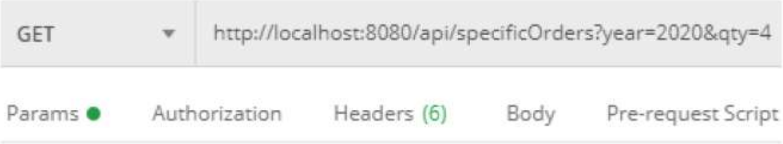
```
[
    {
        "oid": "O200",
        "orderDate": "2019-12-01",
        "orderSalesperson": {
            "spid": "S1",
            "name": "NewAmmamdAslam"
        },
        "orderProduct": {
            "pid": "TV",
            "product": "TV",
            "quantity": 4,
            "orderable": "true",
            "productOrders": [
```

4.2.2 GET /api/specificOrders?year=<*year*>&qty=<*quantity*>

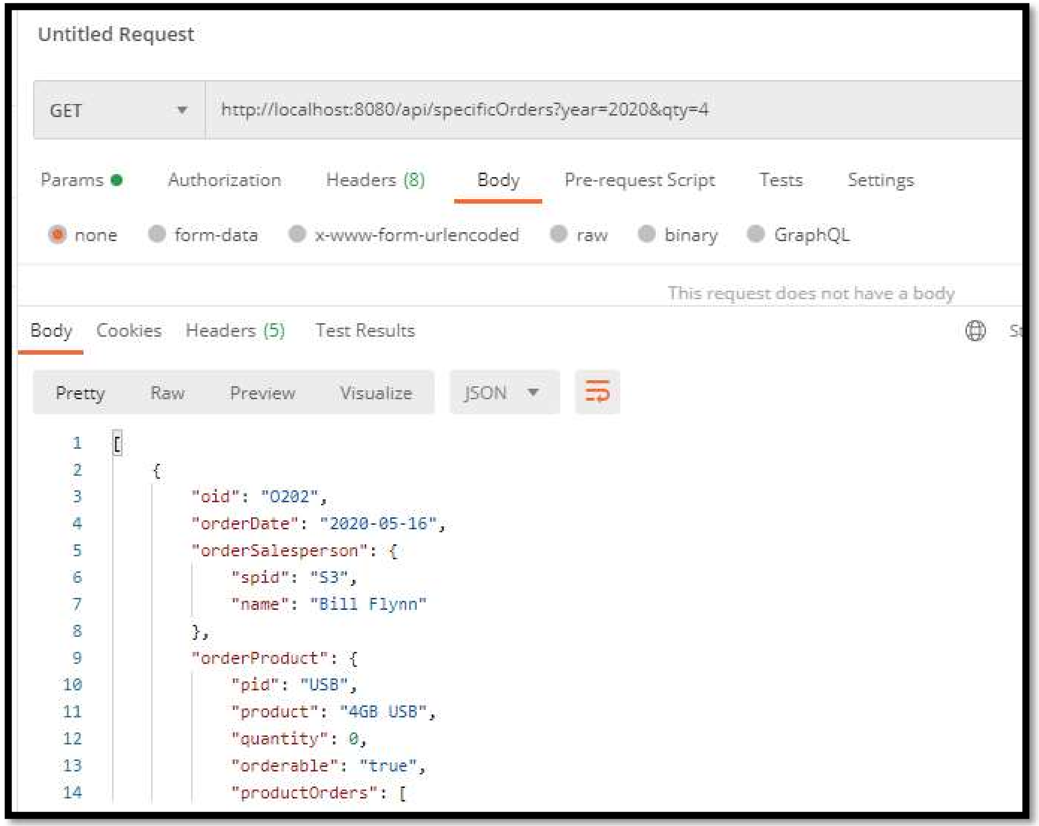
Return the *oid*, *orderDate*, *orderSalesperson.spid*, *orderSalesperson.name*, *orderProduct.pid*, *orderProduct.product*, *orderProduct.quantity*, and *orderProduct.orderable* attributes of all orders where the order year = *year* and order quantity > *qty*.

or example, `/api/specificOrders?year=2020&qty=4` returns all details of orders in the year 2020 that had an order quantity of greater than 4.

Both *year* and *qty* parameters must supplied to this endpoint and their values must always be integers.



GET ▾ http://localhost:8080/api/specificOrders?year=2020&qty=4



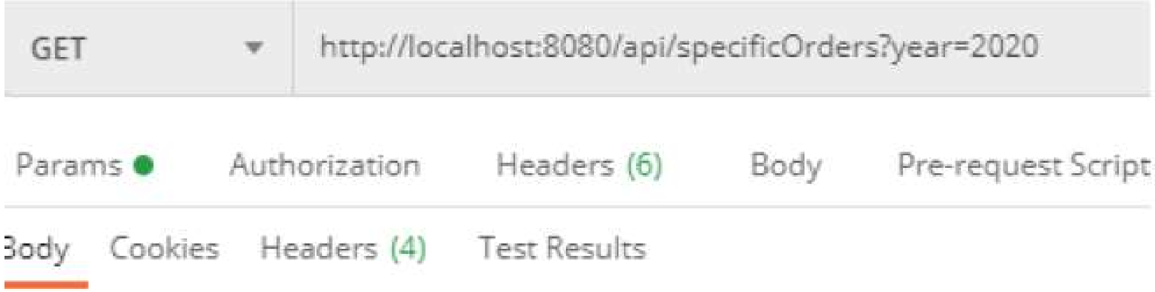
Untitled Request

GET ▾ http://localhost:8080/api/specificOrders?year=2020&qty=4

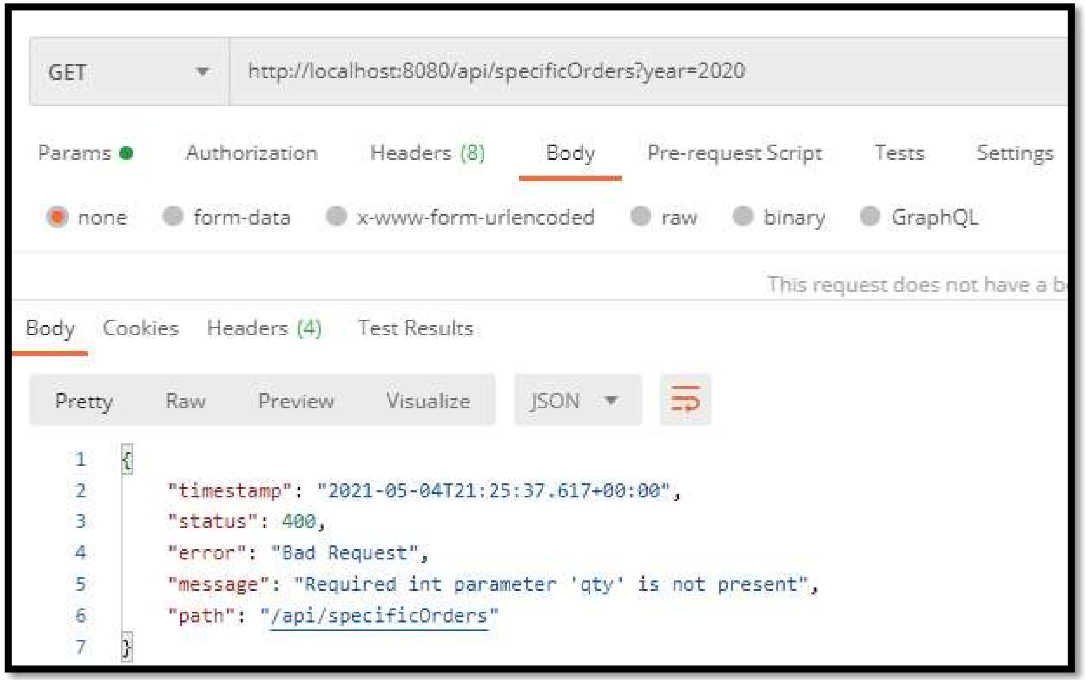
● none ○ form-data ○ x-www-form-urlencoded ○ raw ○ binary ○ GraphQL

This request does not have a body

```
[
    {
        "oid": "O202",
        "orderDate": "2020-05-16",
        "orderSalesperson": {
            "spid": "S3",
            "name": "Bill Flynn"
        },
        "orderProduct": {
            "pid": "USB",
            "product": "4GB USB",
            "quantity": 0,
            "orderable": "true",
            "productOrders": [
```



http://localhost:8080/api/specificOrders?year=2020



http://localhost:8080/api/specificOrders?year=2020

● none

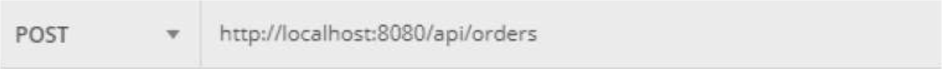
This request does not have a b

```
{
    "timestamp": "2021-05-04T21:25:37.617+00:00",
    "status": 400,
    "error": "Bad Request",
    "message": "Required int parameter 'qty' is not present",
    "path": "/api/specificOrders"
}
```

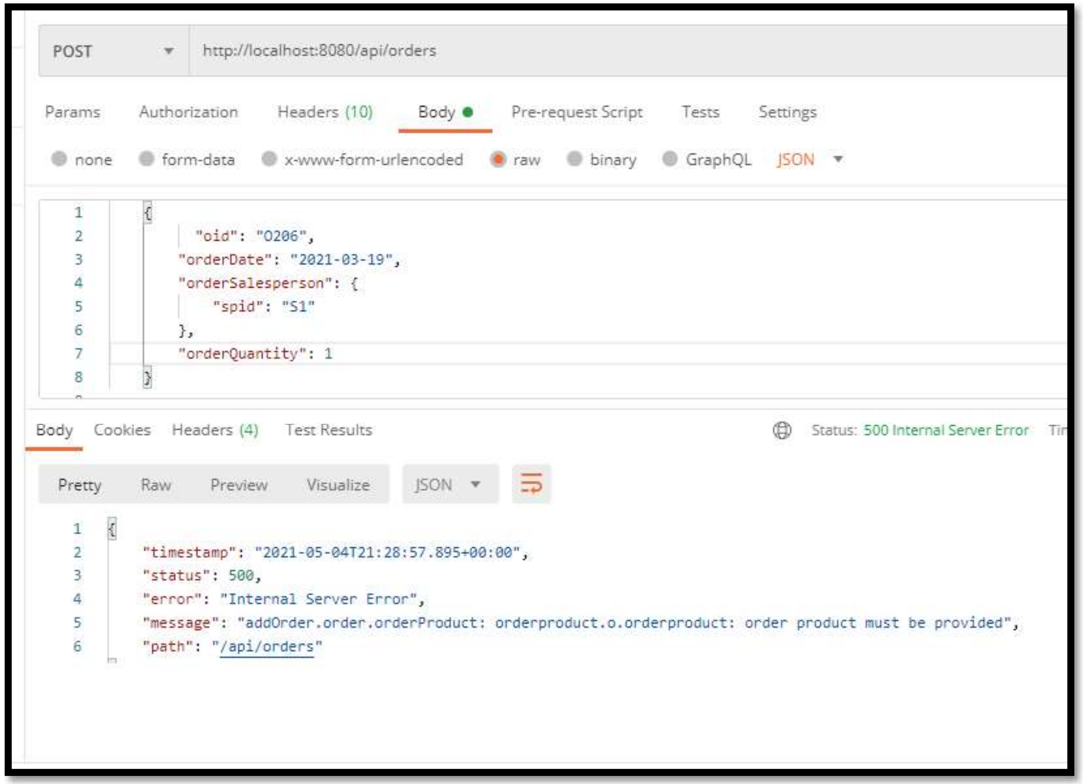
4.2.3 POST /api/orders

Persists the order object specified in the request body to the database. The order object must contain the *oid*, *orderDate*, *orderQuantity*, *orderSalesperson.spid* and *orderProduct.pid* attributes.

If any of these attributes is not present a HTTP 500 response should be returned with an appropriate error message. (NOTE: order*Quantity* should be > 0).



POST http://localhost:8080/api/orders



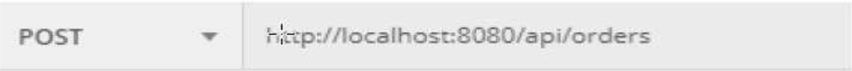
POST http://localhost:8080/api/orders

```
{
    "oid": "O206",
    "orderDate": "2021-03-19",
    "orderSalesperson": {
        "spid": "S1"
    },
    "orderQuantity": 1
}
```

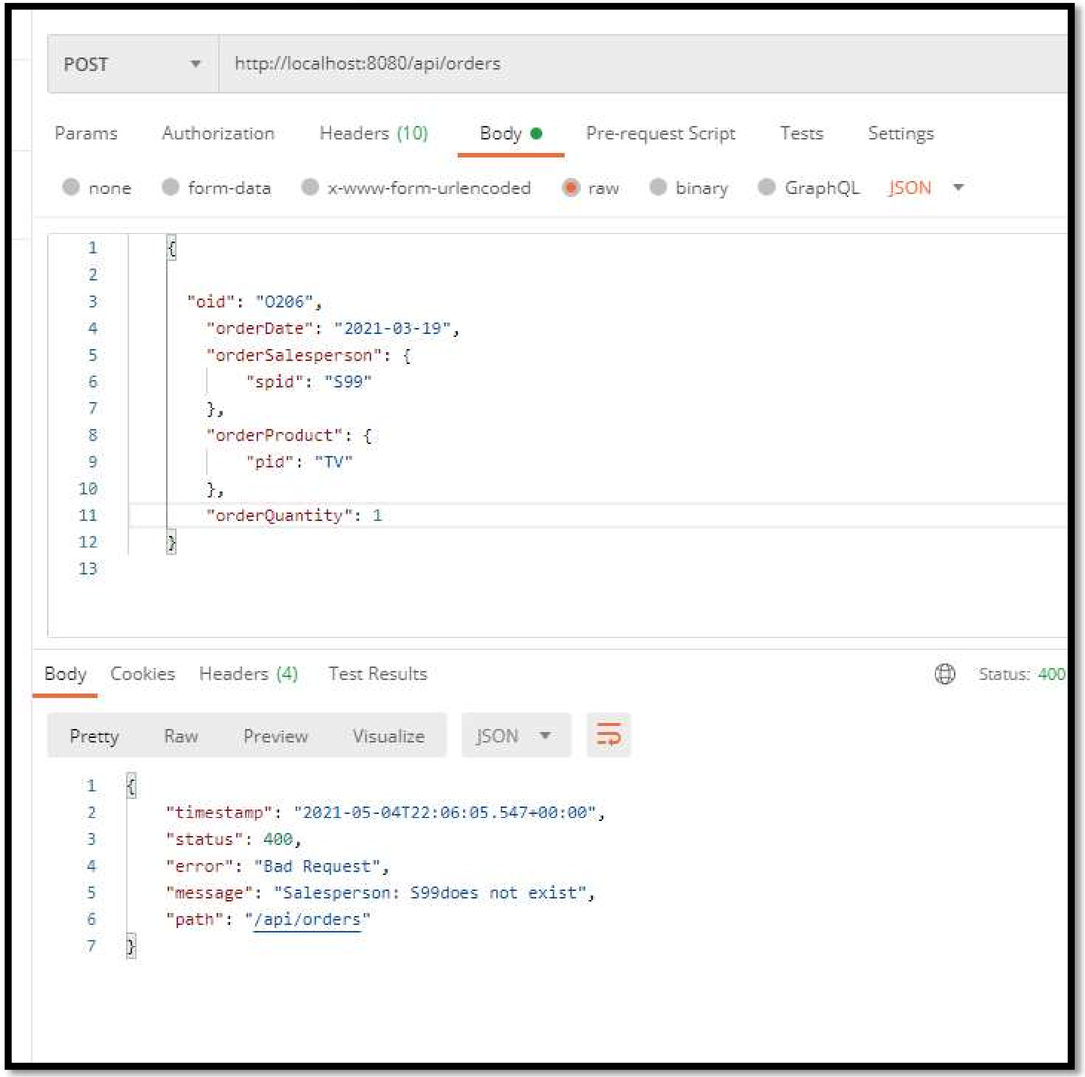
Status: 500 Internal Server Error

```
{
    "timestamp": "2021-05-04T21:28:57.895+00:00",
    "status": 500,
    "error": "Internal Server Error",
    "message": "addOrder.order.orderProduct: orderproduct.o.orderproduct: order product must be provided",
    "path": "/api/orders"
```

If *orderSalesperson.spid* is not a valid salesperson, a HTTP 422 response should be returned with the following error message: “Salesperson {orderSalesperson.spid} does not exist” where {orderSalesperson.spid} is the salesperson spid in request body.



POST http://localhost:8080/api/orders



POST http://localhost:8080/api/orders

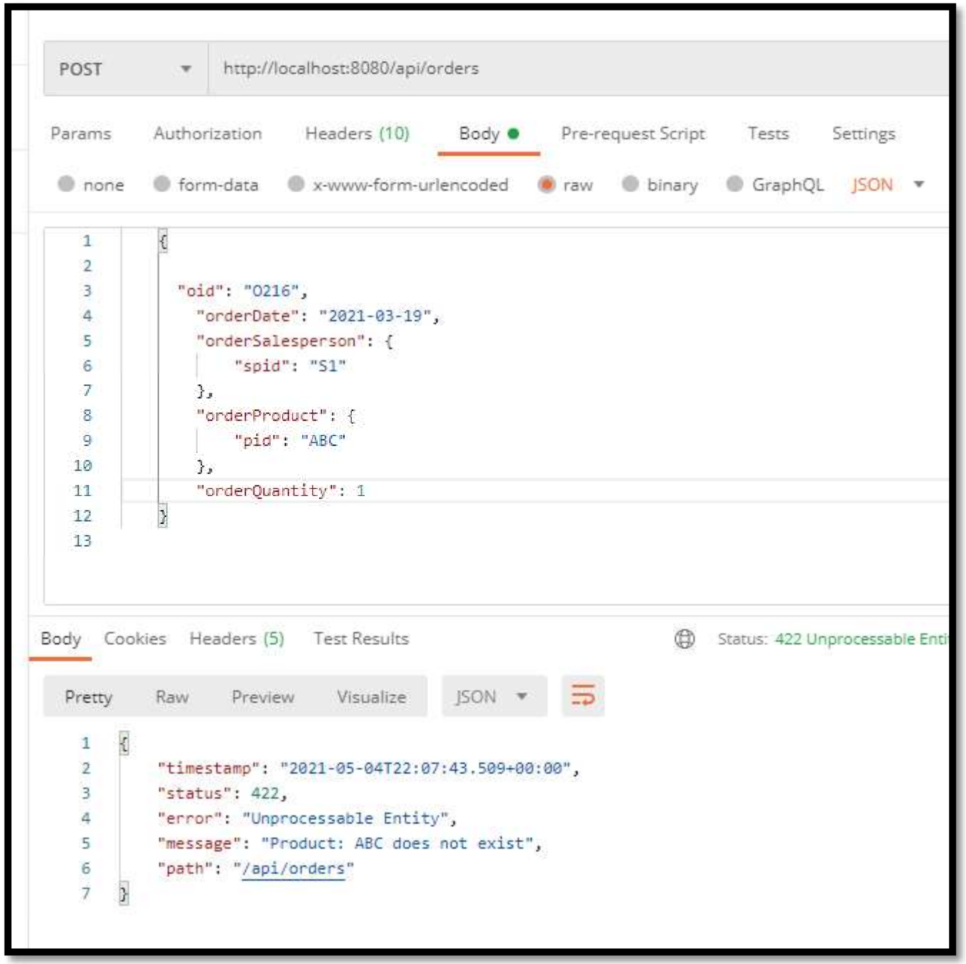
```
{

  "oid": "O206",
    "orderDate": "2021-03-19",
    "orderSalesperson": {
        "spid": "S99"
    },
    "orderProduct": {
        "pid": "TV"
    },
    "orderQuantity": 1
}

```

Status: 400

```
{
    "timestamp": "2021-05-04T22:06:05.547+00:00",
    "status": 400,
    "error": "Bad Request",
    "message": "Salesperson: S99does not exist",
    "path": "/api/orders"
}
```



POST http://localhost:8080/api/orders

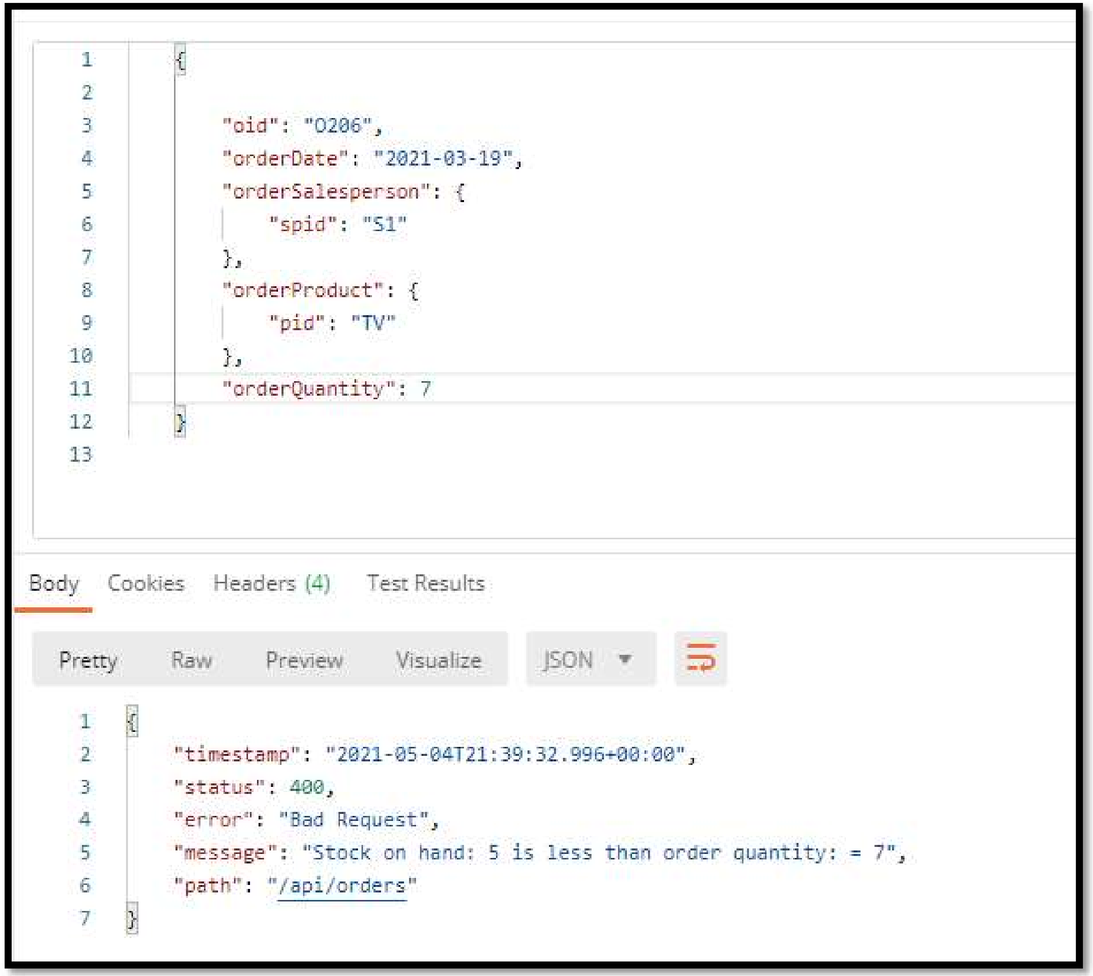
```
{

  "oid": "O216",
    "orderDate": "2021-03-19",
    "orderSalesperson": {
        "spid": "S1"
    },
    "orderProduct": {
        "pid": "ABC"
    },
    "orderQuantity": 1
}

```

Status: 422 Unprocessable Enti

```
{
    "timestamp": "2021-05-04T22:07:43.509+00:00",
    "status": 422,
    "error": "Unprocessable Entity",
    "message": "Product: ABC does not exist",
    "path": "/api/orders"
}
```



```
{

    "oid": "O206",
    "orderDate": "2021-03-19",
    "orderSalesperson": {
        "spid": "S1"
    },
    "orderProduct": {
        "pid": "TV"
    },
    "orderQuantity": 7
}

```

```
{
    "timestamp": "2021-05-04T21:39:32.996+00:00",
    "status": 400,
    "error": "Bad Request",
    "message": "Stock on hand: 5 is less than order quantity: = 7",
    "path": "/api/orders"
}
```