Grocery API Documentation

Endpoint: POST /auth/register

Description: To create a new user account

Payload

- email
    - Data type: string
    - Description: New user email
- password
    - Data type: string
    - Description: New user password

Response

```
{
    "status": 201,
    "message": "Registration successful"
}
```

SQL Code

```
INSERT INTO tbl_users (email, password)
VALUES ("ewyaks.5@gmail.com", "letmein123")
```

Endpoint: POST /auth/login

Description: To authenticate user

Payload

- email
    - Data type: string
    - Description: Registered user email
- password
    - Data type: string
    - Description: Registered user password

Response

```
{
    "status": 200,
    "message": "Login successful",
    "data": {
        "user_id": "8e22ed13-f793-4bc9-980b-257bd51e02d4"
    }
}
```

SQL Code

```
SEELCT user_id FROM tbl_users
WHERE email = "ewyaks.5" AND password = "letmein123"
```

Endpoint: GET /grocery/groceries

Description: To retrieve all the groceries associated with the specific user

Query Params

- userId

- Data type: string
- Description: To identifiy which groceries from a specific user to be retrieved

Response

```
{
    "status": 200,
    "message": "OK",
    "data": [
        {
            "grocery_id": "11f72c79-2a55-4cb1-b743-5424169a8d3a",
            "title": "Canned goods",
            "description": "Foods that are preserved by canning",
            "created_at":  "2023-11-20 11:00:44"
        },
        ...
    ]
}
```

SQL Code

```
SELECT grocery_id, title, description, created_at
FROM tbl_groceries
WHERE user_id = "8e22ed13-f793-4bc9-980b-257bd51e02d4"
```

Endpoint: POST /grocery/grocery

Description: To create a new grocery.

Payload

- title
    - Data type: string
    - Description: New title for the grocery
- description
    - Data type; string
    - Description: New description for the grocery

Response

```
{
    "status": 201,
    "message": "Grocery added successfully",
}
```

SQL Code

```
INSERT INTO tbl_groceries (title, description, user_id)
VALUES ("Canned goods", "Foods that are preserved by canning", "8e22ed13-f793-4bc9-980b-257bd51e02d4")
```

Endpoint: DELETE /grocery/grocery/:groceryId

Param

- groceryId
    - Data type: string
    - Description: Id of the grocery to be deleted

Response

```json
{
    "status": 200,
    "message": "Deletion successful"
}
```

SQL Code

```sql
DELETE tbl_grocery_items WHERE grocery_id = "11f72c79-2a55-4cb1-b743-5424169a8d3a";

DELETE tbl_groceries WHERE grocery_id = "11f72c79-2a55-4cb1-b743-5424169a8d3a";
```

Endpoint: GET /grocery/grocery_items

Description: Retrieves only the from the specific grocery from the database.

Query Params

- groceryId
    - Data type: string
    - Description: To identify which grocery items should be retrieved.

Response

```json
{
    "status": 200,
    "message": "OK",
    "data": [
        {
            "item_id": "1e59142c-bb00-492f-b53f-3ec45bb90363",
            "name": "Sardinas",
            "qty": 10,
            "price": 25.00,
            "created_at": "2023-11-20 10:57:54"
        },
        ...
    ]
}
```

SQL Code

```sql
SELECT item_id, name, qty, price, created_at FROM tbl_grocery_items
WHERE grocery_id = "11f72c79-2a55-4cb1-b743-5424169a8d3a"
```

Endpoint: POST /grocery/grocery_item

Description: To create new grocery item

Payload

- name
    - Data type: string
    - Description: New item name
- qty
    - Data type: number
    - Description: Quantity for the new item
- price
    - Data type: number,
    - Description: Price for the new item
- grocery_id
    - Data type: string,
    - Description: Grocery id for the new item

Response

```
{
    "status": 201,
    "message": "Item added successfully"
}
```

SQL Code

```sql
INSERT INTO tbl_grocery_items (name, qty, price, grocery_id)
VALUES ("Sardinas", 10, 25.00, "1e59142c-bb00-492f-b53f-3ec45bb90363")
```

Endpoint: PATCH /grocery/grocery_item/:item_id

Param

- itemId
  - Data type: string
  - Description: id of the item to be updated

Payload

- name
  - Data type: string
  - Description: New name for the item
- qty
  - Data type: number
  - Description: New quantity for the item
- price
  - Data type: number,
  - Description: New price for the item

Response

```
{
    "status": 200,
    "message": "Item updated successfully"
}
```

SQL Code

```sql
UPDATE tbl_grocery_items
SET name = "Sardines", qty = 20, price = 22.00
WHERE item_id = "1e59142c-bb00-492f-b53f-3ec45bb90363"
```

Endpoint: DELETE /grocery/grocery_item/:itemId

Param

- itemId
  - Data type: string
  - Description: id of the item to be deleted

Response

```
{
    "status": 200,
    "message": "item deleted successfully"
}
```

SQL Code

```sql
DELETE tbl_grocery_items
WHERE item_id = "1e59142c-bb00-492f-b53f-3ec45bb90363"
```