### Order Processing

#### Test Cases

##### 1. Creating an Order

* **Description**: Verify that a new order can be created with valid data.
* **Steps**:
  1. Send a POST request to /api/Order with a JSON containing the details of the new order.
  2. Verify that the response has a status code of 201 Created.
  3. Ensure that the order has been created correctly by checking the response or querying the order list.
* **Test Data**:

json

{

"orderItems": [

{

"productId": 1,

"quantity": 2

}

],

"orderDate": "2024-07-30T12:00:00Z"

}

* **Expected Result**: Response with status code 201 and the new order created correctly.

##### 2. Cancelling an Order

* **Description**: Verify that an existing order can be cancelled.
* **Steps**:
  1. Send a DELETE request to /api/Order/{id} with the ID of the order to be cancelled.
  2. Verify that the response has a status code of 204 No Content.
  3. Ensure that the order has been cancelled correctly by checking the order list or attempting to access the cancelled order.
* **Test Data**:

json

{

"id": 1

}

* **Expected Result**: Response with status code 204 and the order cancelled correctly.

##### 3. Updating Order Status

* **Description**: Verify that the status of an existing order can be updated.
* **Steps**:
  1. Send a PUT request to /api/Order/{id} with a JSON containing the updated order status.
  2. Verify that the response has a status code of 200 OK.
  3. Ensure that the order status has been updated correctly by checking the response or querying the order list.
* **Test Data**:

json

{

"status": "Shipped"

}

* **Expected Result**: Response with status code 200 and the order status updated correctly.

##### 4. Error Handling for Invalid Order Operations

* **Description**: Verify that invalid operations are handled correctly when creating, cancelling, or updating an order.
* **Steps**:
  1. Send a POST, DELETE, or PUT request to /api/Order with invalid data.
  2. Verify that the response has a status code of 400 Bad Request.
  3. Ensure that the error message is appropriate for the invalid data or operation attempt.
* **Test Data**:

json

{

"orderItems": [],

"orderDate": "invalid-date"

}

* **Expected Result**: Response with status code 400 and appropriate error message.