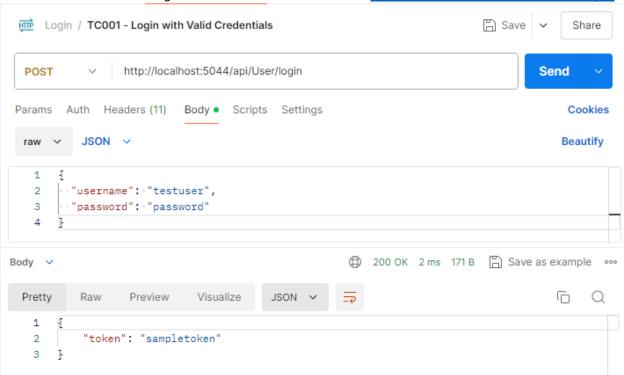
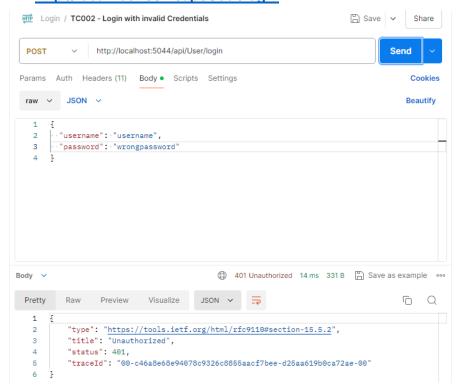
PART 3: TEST EXECUTION AND REPORTING

1. Test Execution - Backend (C# APIS):

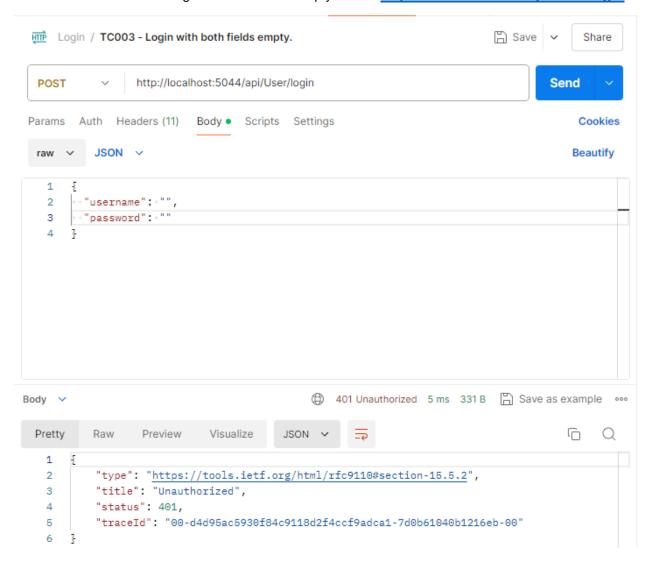
1. TC001: Login with Valid Credentials - POST http://localhost:5044/api/User/login



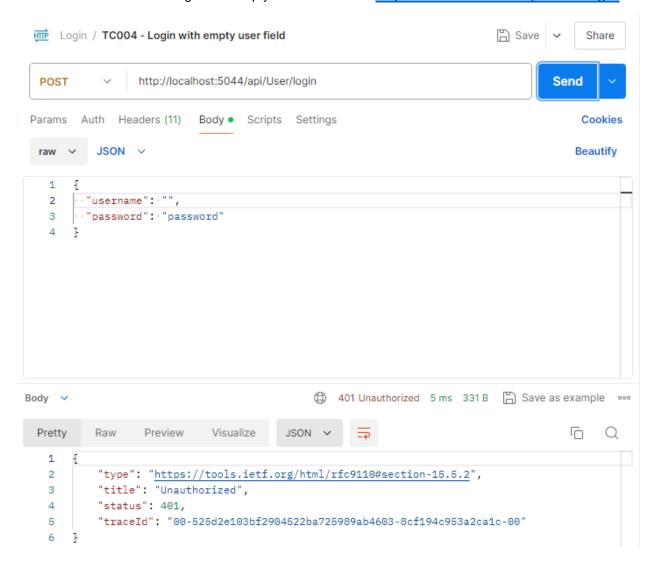
 TC002: Login with invalid Credentials - POST http://localhost:5044/api/User/login



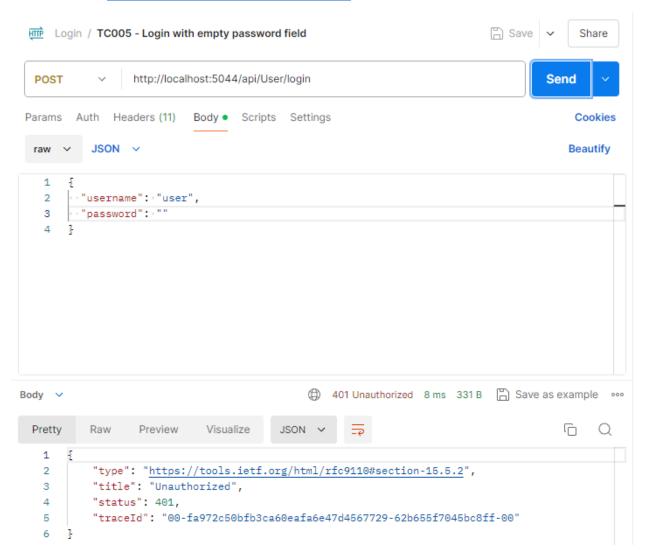
3. TC003: Login with both fields empty POST http://localhost:5044/api/User/login



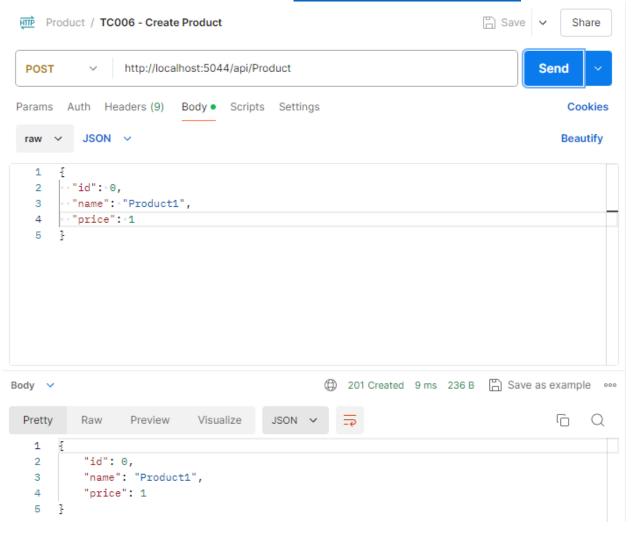
4. TC004: Login with empty user field POST http://localhost:5044/api/User/login



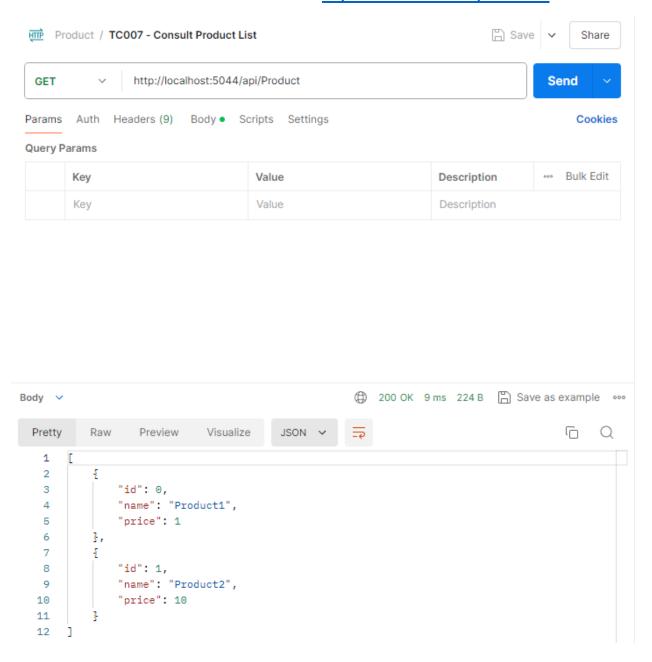
TC005: Login with empty password field POST http://localhost:5044/api/User/login



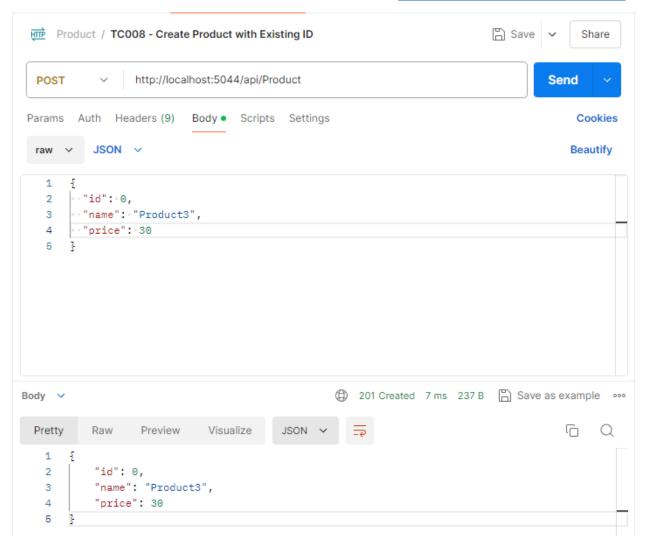
6. TC006: Create Product POST http://localhost:5044/api/Product



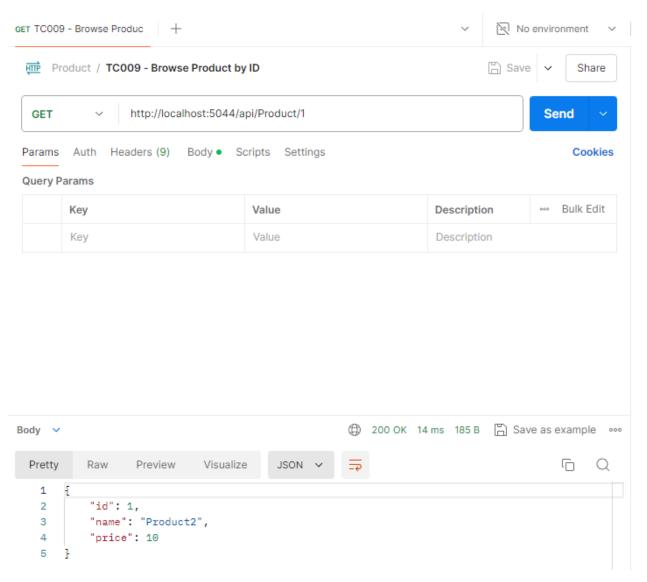
7. TC007: Consult Product List GET http://localhost:5044/api/Product



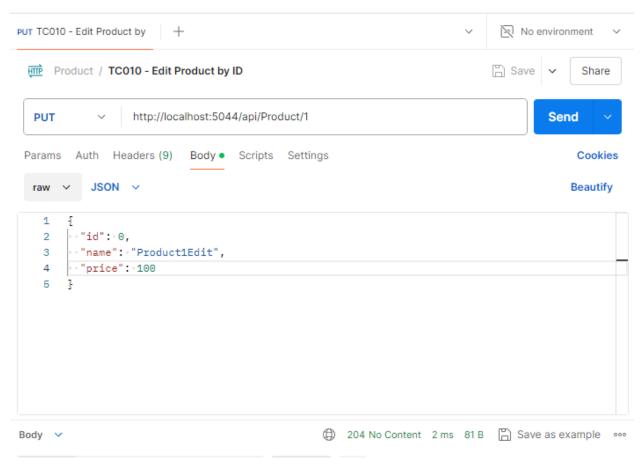
8. TC008: Create Product with Existing ID POST http://localhost:5044/api/Product



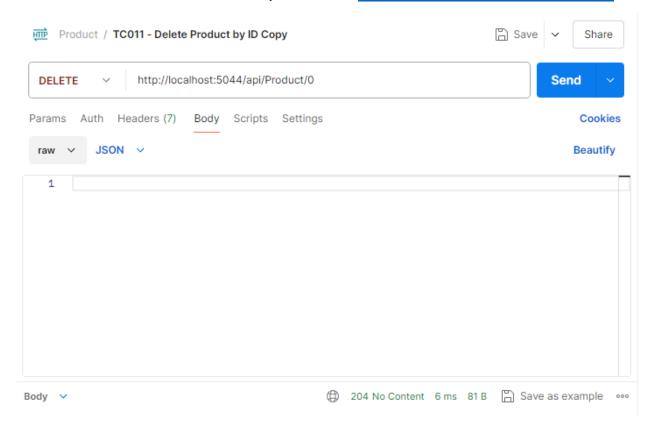
9. TC009: Browse Product by ID GET http://localhost:5044/api/Product/{ld}



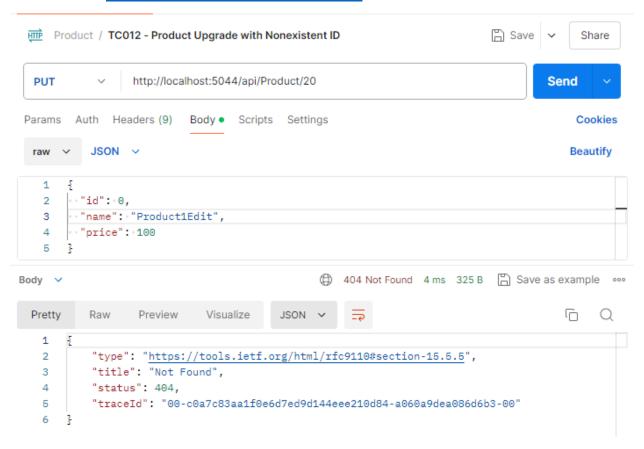
10.TC010: Edit Product by ID PUT http://localhost:5044/api/Product/{ld}



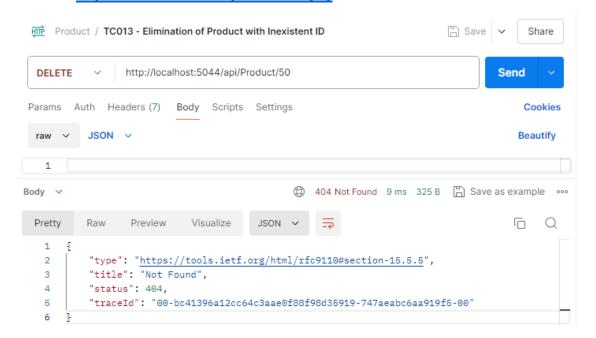
11.TC011: Delete Product by ID DELETE http://localhost:5044/api/Product/{ld}



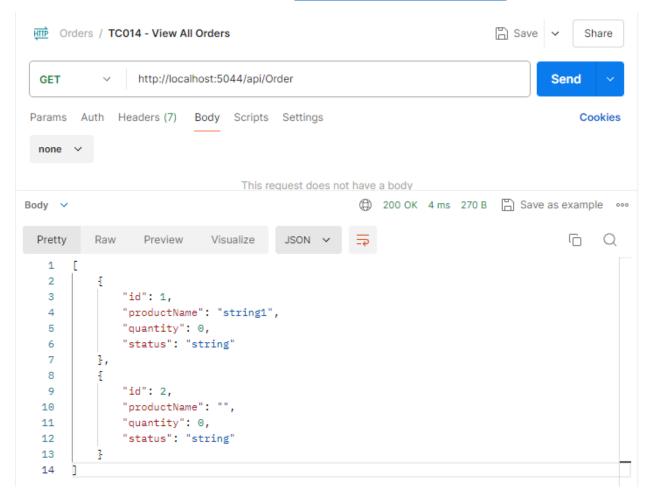
12.TC012: Product Upgrade with Nonexistent ID PUT http://localhost:5044/api/Product/{ld}



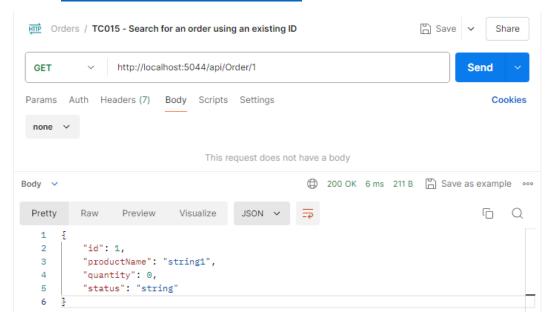
13.TC013: Elimination of Product with Inexistent ID DELETE http://localhost:5044/api/Product/{ld}



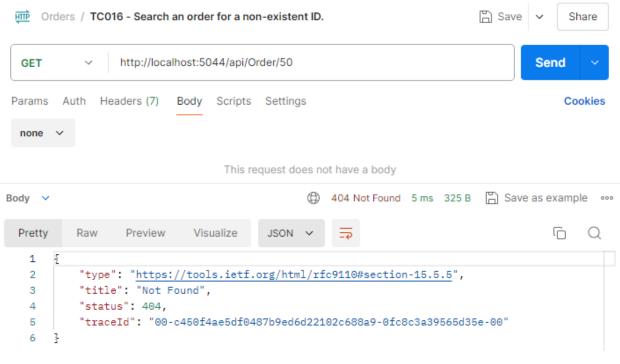
14.TC014: View all orders GET http://localhost:5044/api/Order



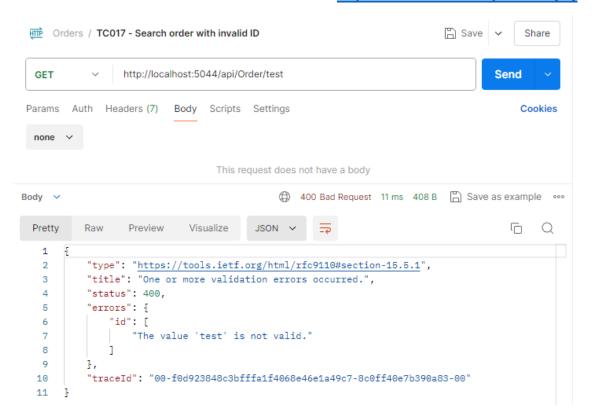
15.TC015: Search for an order using an existing ID GET http://localhost:5044/api/Order/{id}



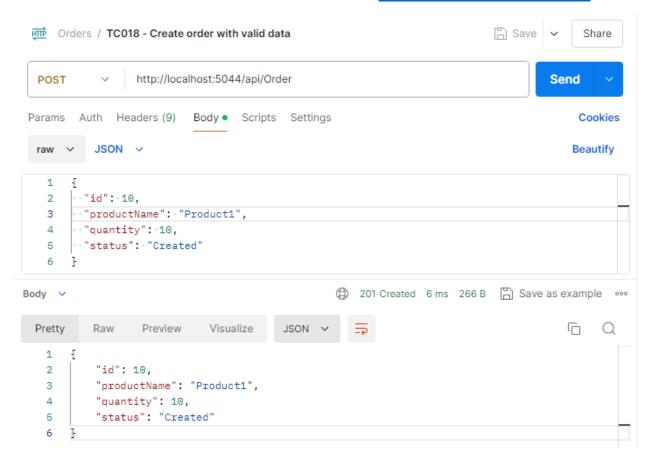
16.TC016: Search an order for a non-existent ID GET http://localhost:5044/api/Order/{id}



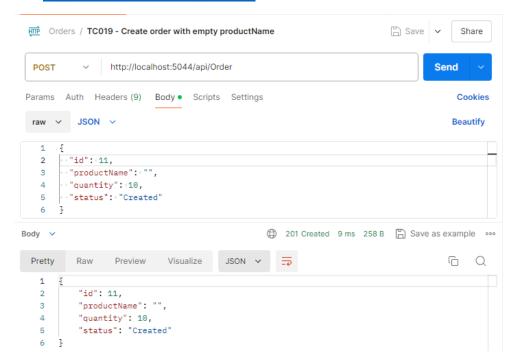
17.TC017: Search order with invalid ID GET http://localhost:5044/api/Order/{id}



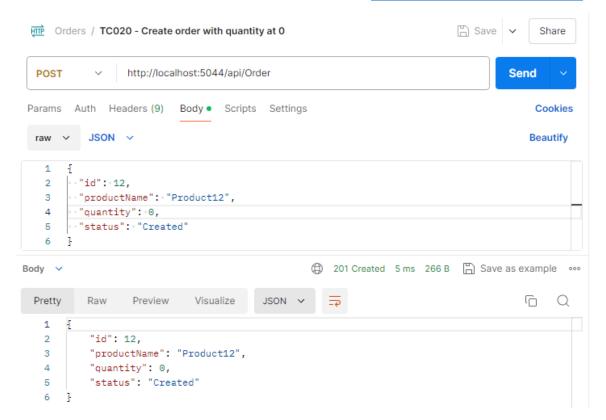
18.TC018: Create order with valid data POST http://localhost:5044/api/Order



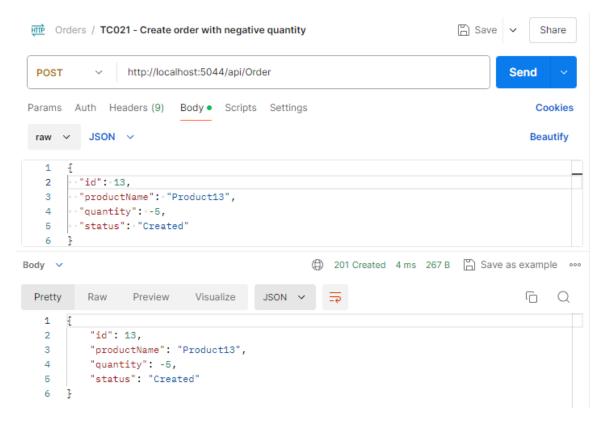
19.TC019: Create order with empty productName POST http://localhost:5044/api/Order



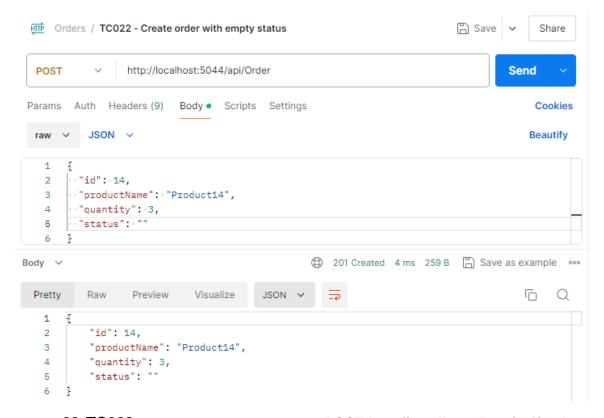
20.TC020: Create order with quantity at 0 POST http://localhost:5044/api/Order



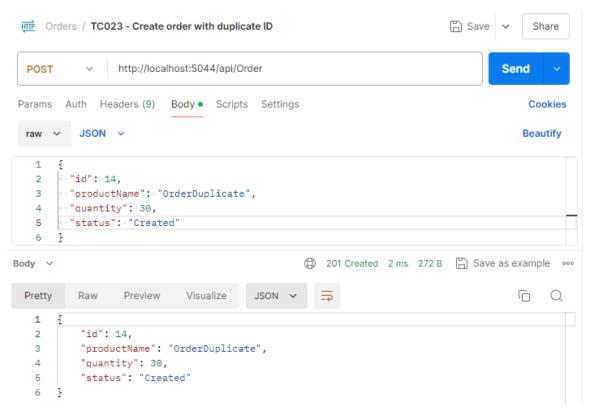
21.TC021: Create order with negative quantity POST http://localhost:5044/api/Order



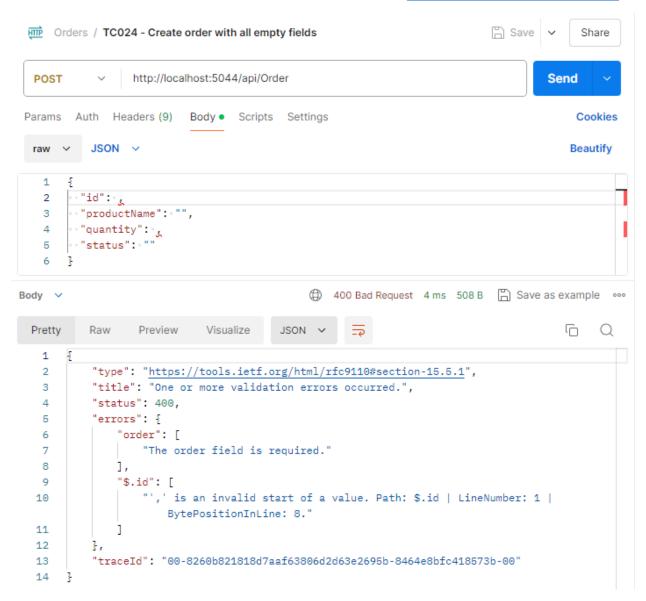
22.TC022: Create order with empty status POST http://localhost:5044/api/Order



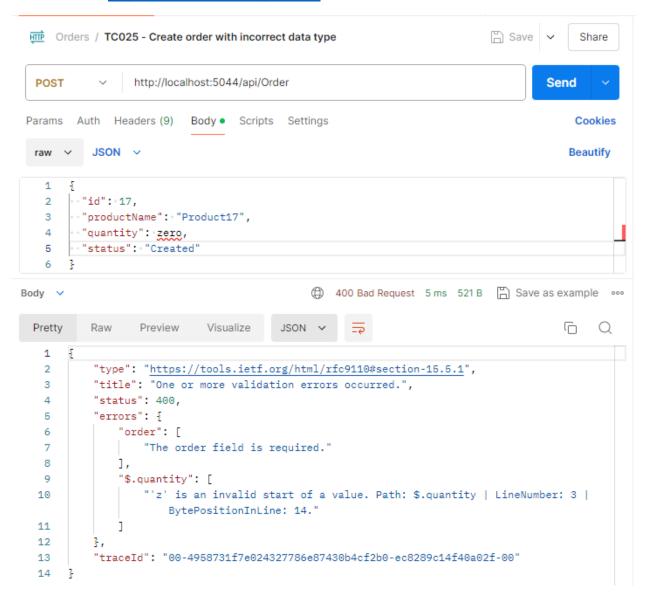
23. TC023: Create order with duplicate ID POST http://localhost:5044/api/Order



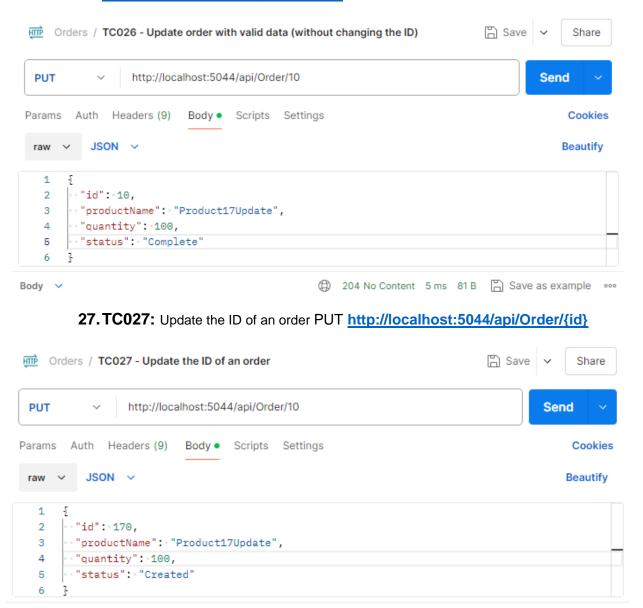
24.TC024: Create order with all empty fields POST http://localhost:5044/api/Order



25.TC025: Create order with incorrect data type POST http://localhost:5044/api/Order



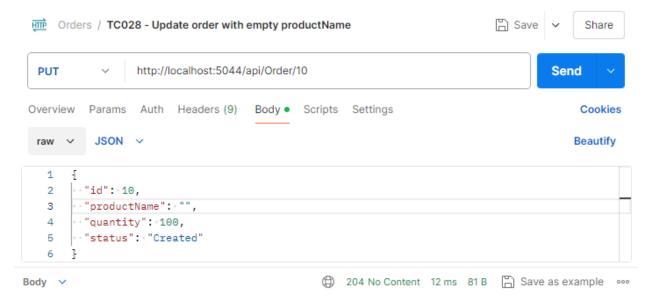
26.TC026: Update order with valid data (without changing the ID) PUT http://localhost:5044/api/Order/{id}



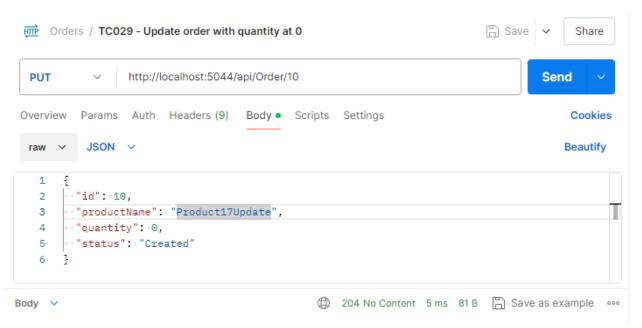
204 No Content 3 ms 81 B Save as example ...

Body V

28.TC028: Update order with empty productName PUT http://localhost:5044/api/Order/{id}

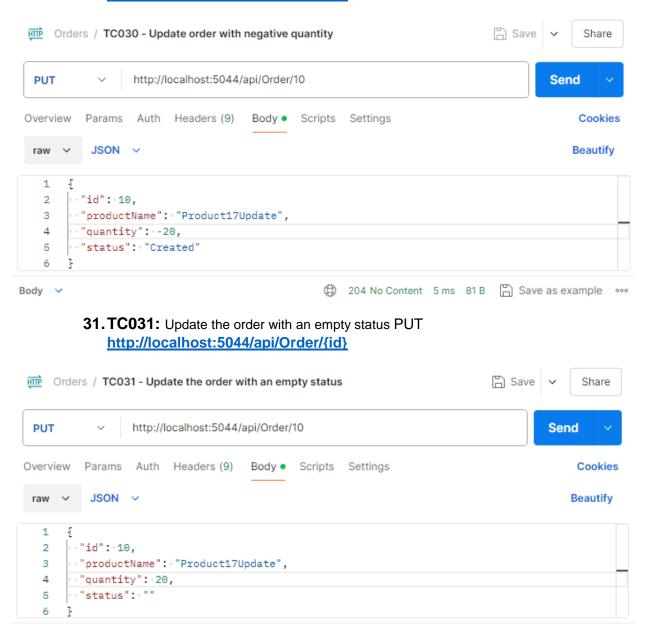


29.TC029: Update order with quantity at 0 PUT http://localhost:5044/api/Order/{id}



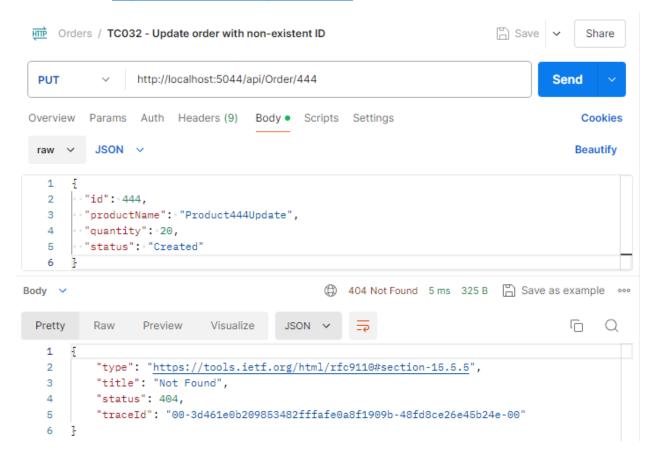
30.TC030: Update order with negative quantity PUT http://localhost:5044/api/Order/{id}

Body V

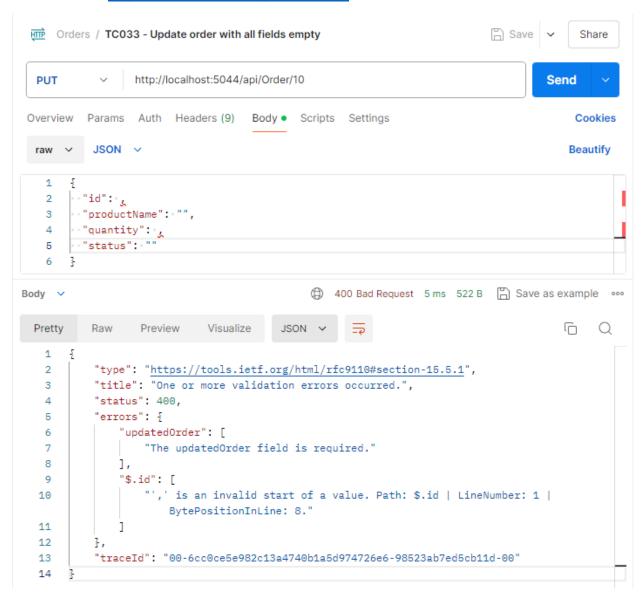


204 No Content 5 ms 81 B Save as example ...

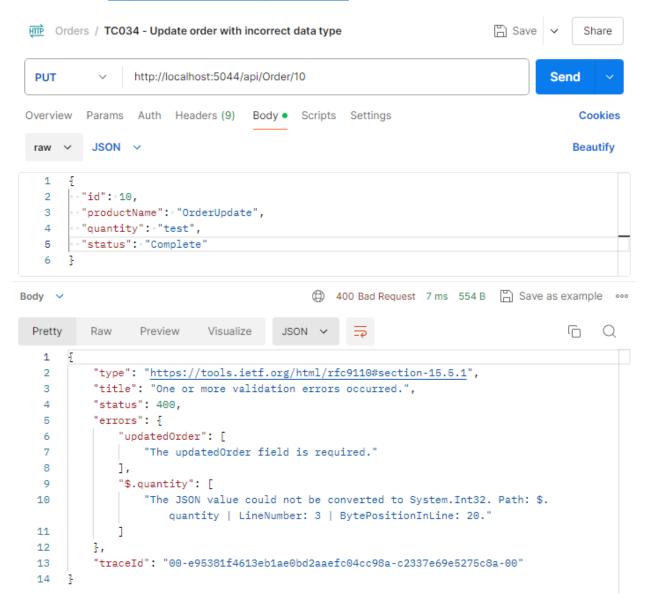
32.TC032: Update order with non-existent ID PUT http://localhost:5044/api/Order/{id}



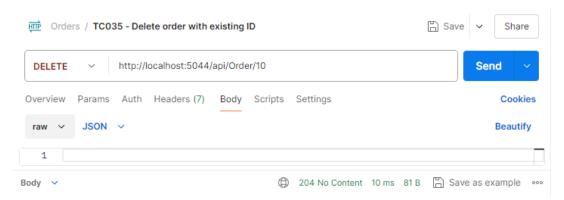
33.TC033: Update order with all fields empty PUT http://localhost:5044/api/Order/{id}



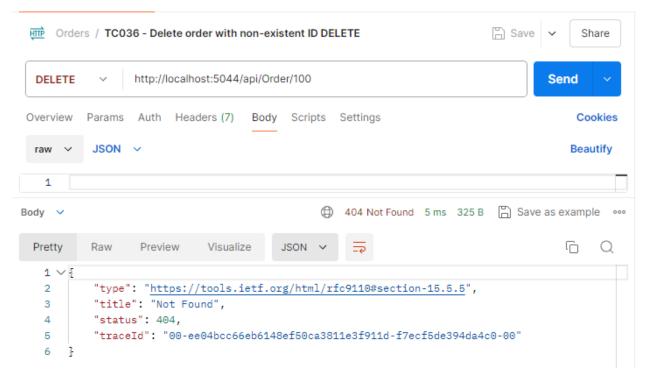
34.TC034: Update order with incorrect data type PUT http://localhost:5044/api/Order/{id}



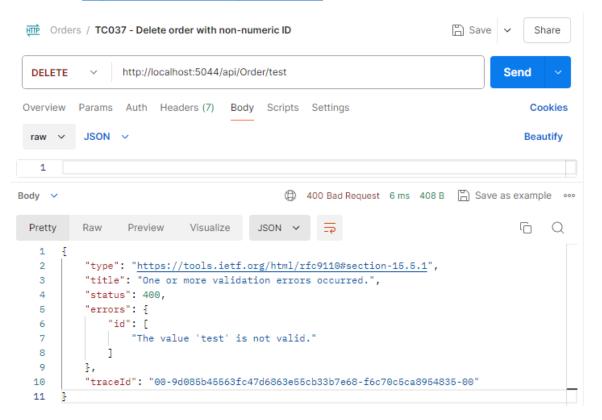
35.TC035: Delete order with existing ID DELETE http://localhost:5044/api/Order/{id}



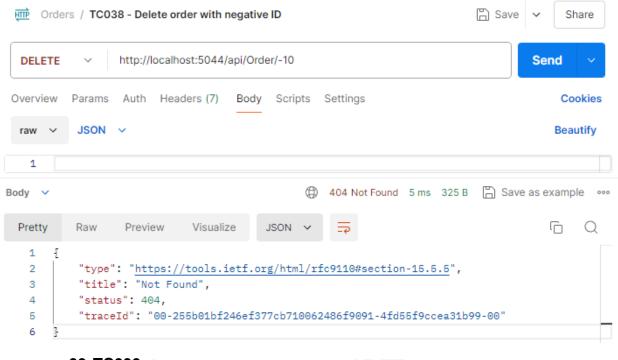
36.TC036: Delete order with non-existent ID DELETE http://localhost:5044/api/Order/{id}



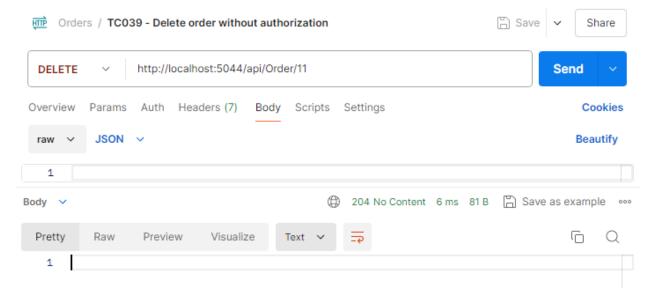
37.TC037: Delete order with non-numeric ID DELETE http://localhost:5044/api/Order/{id}



38.TC038: Delete order with negative ID DELETE http://localhost:5044/api/Order/{id}



39.TC039: Delete order without authorization DELETE http://localhost:5044/api/Order/{id}



2. Documentation of Results:

- Results Format:
 - Test Case ID: Identification of the test case.
 - o **Test Case Title:** Title of the test case to be executed.

- Result: Pass if the test case has been successful, Fail if it does not meet expectations.
- Comments: Any additional details, such as if there were unexpected behaviors or if the result was not as expected.

Test Case ID	Test Case Title	Result	Comments
TC001	Login with Valid Credentials	Pass	Authentication was successful and a token was received in the response body.
TC002	Login with Invalid Credentials	Pass	The system correctly rejected the authentication and returned a 401 (Unauthorized) error.
TC003	Login with both fields empty.	Pass	The system correctly returned a 401 (Unauthorized) error when attempting to log in with both fields empty.
TC004	Login with empty user field.	Pass	The system correctly returned a 401 (Unauthorized) error when trying to start the system.
TC005	Login with empty password field.	Pass	The system correctly returned a 401 (Unauthorized) error when attempting to log in with only the password field.
TC006	Create Product	Pass	The system successfully created the product with the data provided and returned a status 201 along with the details of the product created.
TC007	Consult Product List	Pass	The system successfully returned a list of products with status 200. The list contains the products with their respective id, name, and price.
TC008	Create Product with Existing ID	Fail	The system allowed the creation of a product with an ID that already existed, which is incorrect. A status 201 was received when a status 409 is expected.

			The product query with
TC009	Browse Product by ID	Pass	ID 1 was successful. A status 200 was
			received.
			The edition of the
			product with ID 0 was
			successful. A status
T0040	Edit Deciderat beed	Dana	204 was received,
TC010	Edit Product by ID	Pass	indicating that the operation was
			completed successfully
			with no additional
			content.
			The deletion of the
			product with the
			specified ID was
			successful. A status
TC011	Delete Product by ID	Pass	204 was received,
	_ = 5.5.5		indicating that the
			operation was
			completed successfully with no additional
			content.
			Product update with a
	Product Upgrade with Nonexistent ID		non-existent ID was
TC012		Pass	handled correctly. A
			404 status was
			received.
			The removal of the
			product with a non-
TC013	Elimination of Product	Pass	existent ID was
	with Inexistent ID		handled correctly. A
			404 status was received.
			The consultation of all
			products was
TC014	View all orders	Pass	successful. A status
			200 was received.
			The issue of the
	Search for an order		product with the
TC015	using an existing ID.	Pass	specified ID was
			successful. A status
			200 was received.
			The request to access a non-existent resource
TC016	Search an order for a	Pass	was handled correctly.
10010	non-existent ID.	1 433	A 404 status was
			received.
			The request to access
	Search order with invalid ID		a resource with an
TC017		Pass	invalid ID was handled
	IIIValia ID		correctly. A status 400
			was received.
TC018	Create order with valid	Pass	The creation of the
	data		order with all valid data

		1	
			was successful. A
			status 201 was received.
	+		Order creation with
			empty productName
			was not handled
			correctly. A 201 status
			was received,
TC019	Create order with	Fail	indicating that the order
1.00.0	empty productName	. a	was created even
			though the
			productName was
			empty, which should
			not be allowed.
			The creation of the
			order with quantity
			equal to 0 was not
	Create order with		handled correctly. A
TC020	quantity at 0	Fail	status 201 was
	quantity at 0		received, indicating that
			the order was created
			even though the
			quantity was 0. The creation of the
			order with negative
	Create order with negative quantity		quantity was not
		Fail	handled correctly. A
TC021			201 status was
			received, indicating that
			the order was created
			even though the
			quantity was -5.
			The creation of the
			order with an empty
			status was not handled
T0000	Create order with	e.n	correctly. A status 201
TC022	empty status	Fail	was received,
			indicating that the order was created even
			though the status was
			empty.
			The creation of the
			order with an empty
			status was not handled
	Croate order with		correctly. A status 201
TC023	Create order with duplicate ID	Fail	was received,
	auplicate ID		indicating that the order
			was created even
			though the status was
			empty.
			The API correctly handled the creation of
TC024	Create order with all	Pass	an order with all fields
10024	empty fields	1 433	empty. A status 400
			was received,

			indicating that there
			were validation errors.
TC025	Create order with incorrect data type	Pass	The API correctly handled the creation of an order with all fields empty. A status 400 was received, indicating that there were validation errors.
TC026	Update order with valid data (without changing the ID)	Pass	The API allowed the order to be updated with all valid data. A status 204 was received,
TC027	Update the ID of an order	Fail	The API allowed the modification of the order ID, which is not the expected behavior, since the order ID should not be modifiable. A status 204 was received, indicating that the operation was successful, which contradicts the ID immutability restriction.
TC028	Update order with empty productName	Fail	The API allowed the order update even though the productName was empty, which is not the expected behavior. A status 204 was received, indicating that the operation was successful, which contradicts the validation that should prevent updates with empty fields.
TC029	Update order with quantity at 0	Fail	The API allowed the order update even though quantity was 0, which is not the expected behavior. A status 204 was received, indicating that the operation was successful, which contradicts the validation that should prevent updates with quantity equal to 0.
TC030	Update order with negative quantity	Fail	The API allowed the order to be updated

			even though quantity was negative, which is not the expected
			behavior. A status 204 was received.
TC031	Update the order with an empty status	Fail	The API allowed the order update even though the status was empty, which is not the expected behavior. A 204 status was received, indicating that the operation was successful, which contradicts the validation that should prevent updates with empty status.
TC032	Update order with non- existent ID	Pass	The API responded correctly with a 404 status, indicating that the ID was not found, which is the expected behavior.
TC033	Update order with all fields empty	Pass	The system returned a status 400 with validation error messages.
TC034	Update order with incorrect data type	Pass	The system correctly returned a 400 status code with a validation error, indicating that the quantity field could not be converted to an integer value.
TC035	Delete order with existing ID	Pass	The system successfully updated the order and returned a 204 status code, indicating that the update was successful.
TC036	Delete order with non- existent ID	Pass	The system returned a 404 status code with a "Not Found" message, which is the expected behavior when attempting to update an order that doesn't exist.
TC037	Delete order with non- numeric ID	Pass	The API responded with a status 400, indicating that the value provided is invalid, which is the expected behavior.

TC038	Delete order with negative ID	Fail	The API responded with a 404 status, which is not correct. A 400 status was expected for a negative ID.
TC039	Delete order without authorization	Fail	The API responded with a 204 status, which is not the expected behavior. A 401 status was expected.

3. Defect Reporting:

- Defect Details:
 - o **Defect ID:** A unique identifier for each defect.
 - Associated Test Case ID: Identify in which test case the defect occurs.
 - o **Description:** Clear and concise explanation of the defect found.
 - o **Steps to Replicate:** Detailed step-by-step to replicate the defect.
 - o **Actual Result:** What happens when the scenario is run.
 - Expected Result: What should happen according to the requirements.
 - o **Severity:** Classification of the defect (e.g., Critical, Major, Minor).

Defect ID	Associated Test Case ID	Description	Steps to Replicate	Actual Result	Expected Result	Severity
DEF001	TC003, TC004, TC005	The system does not display a specific error message for empty fields in the login.	1. Access the login page. 2. Leave both fields empty, only fill in the user name, or only fill in the password. 3. Click on "Login".	The system returns a 401 (Unauthoriz ed) error.	The system should display a specific message indicating that all fields are mandatory.	Minor
DEF002	TC008	The system allows the creation of a product with an ID that already exists, which should not be allowed.	1. Send a POST request to /api/Product with an existing product ID. 2. Observe the system response	The system returns a 201 status.	The system should return an error with a 409 status and a message indicating that the ID already exists.	Major (as it allows the creation of duplicates, which can cause problems in product managemen t).

			which is a code 201.			
DEF003	TC019	The API allows the creation of an order with empty productNam e.	1. Send a POST request to /api/Order with empty productNam e.	A status 201 was received and the order was created with empty productNam e.	The API should reject the order creation and return a status 400 or 422, indicating that productNam e cannot be empty.	Major
DEF004	TC020	The API allows the creation of an order with quantity equal to 0.	1. Send a POST request to /api/Order with quantity equal to 0 and all other data valid.	A 201 status was received and the order was created with quantity equal to 0.	The API should reject the order creation and return a status 400 or 422, indicating that quantity cannot be 0.	Major
DEF005	TC021	API allows the creation of an order with negative quantity	1. Send a POST request to /api/Order with the amount in negative and all other valid data.	A 201 status was received and the order was created with negative quantity.	The API should reject the order creation and return a status 400 or 422, indicating that quantity cannot be negative.	Major
DEF006	TC022	The API allows the creation of an order with empty status.	1. Send a POST request to /api/Order with valid productNam e, quantity greater than zero and empty status.	A 201 status was received and the order was created with an empty status.	The API should reject the order creation and return a status 400 or 422.	Major
DEF007	TC028	The API allows the update of an order with empty	1. Send a PUT request to /api/Order/{i d} with an empty	The API responded with a 204 status, indicating that the	The API should reject the update and return an error with a	Major

		productNam e.	productNam e in the body.	operation was successful.	status of 400 or 422, indicating that productNam e cannot be empty.	
DEF008	TC029	The API allows the update of an order with quantity equal to 0.	1. Send a PUT request to /api/Order/{i d} with quantity equal to 0 in the body.	The API responded with a 204 status, indicating that the operation was successful.	The API should reject the update and return an error with a status of 400 or 422, indicating that quantity cannot be 0.	Major
DEF009	TC030	The API allows the update of an order with negative quantity.	1. Send a PUT request to /api/Order/{i d} with negative quantity in the body.	The API responded with a 204 status, indicating that the operation was successful.	The API should reject the update and return an error with a status of 400 or 422, indicating that quantity cannot be negative.	Major
DEF010	TC031	The API allows the update of an order with empty status.	1. Send a PUT request to /api/Order/{i d} with empty status in the body.	The API responded with a 204 status, indicating that the operation was successful.	The API should reject the update and return an error with a status of 400 or 422, indicating that status cannot be empty.	Major
DEF011	TC038	The API does not correctly validate a negative ID when trying to delete an order.	Send a DELETE request to /api/Order/- 1	The API returns a 404 status.	The API should return a 400 status.	Major
DEF012	TC039	The API does not correctly handle the deletion of an	Send a DELETE request to /api/Order/{i d} without a valid	The API returns a 204 status.	The API should return a 401 status.	Critical - This defect is critical because it allows sensitive

unauthorize	authorizatio		operations
d order,	n token.		to be
allowing the operation			performed without
with a 204			proper
status.			authorizatio
			n, which can
			compromise
			the security
			of the
			system.