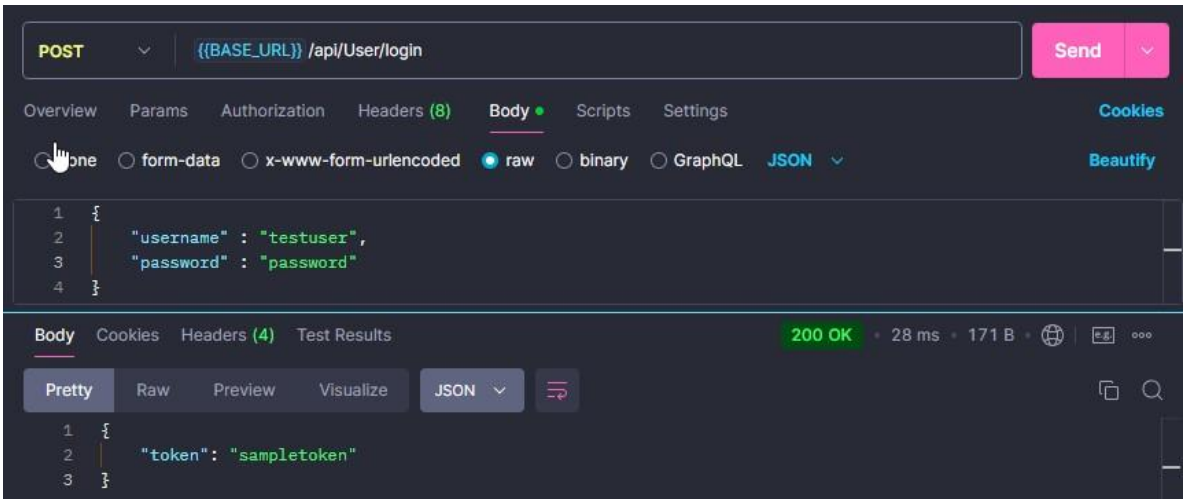
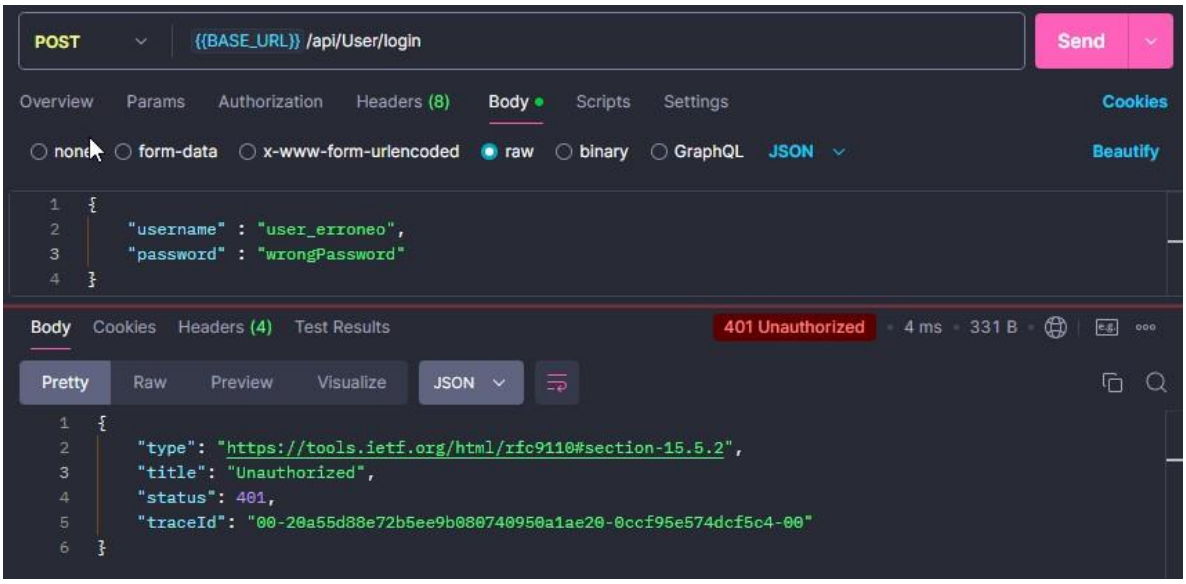
PART 3: TEST EXECUTION AND REPORTING

User Authentication (AU)-BACKEND (C# APIs):

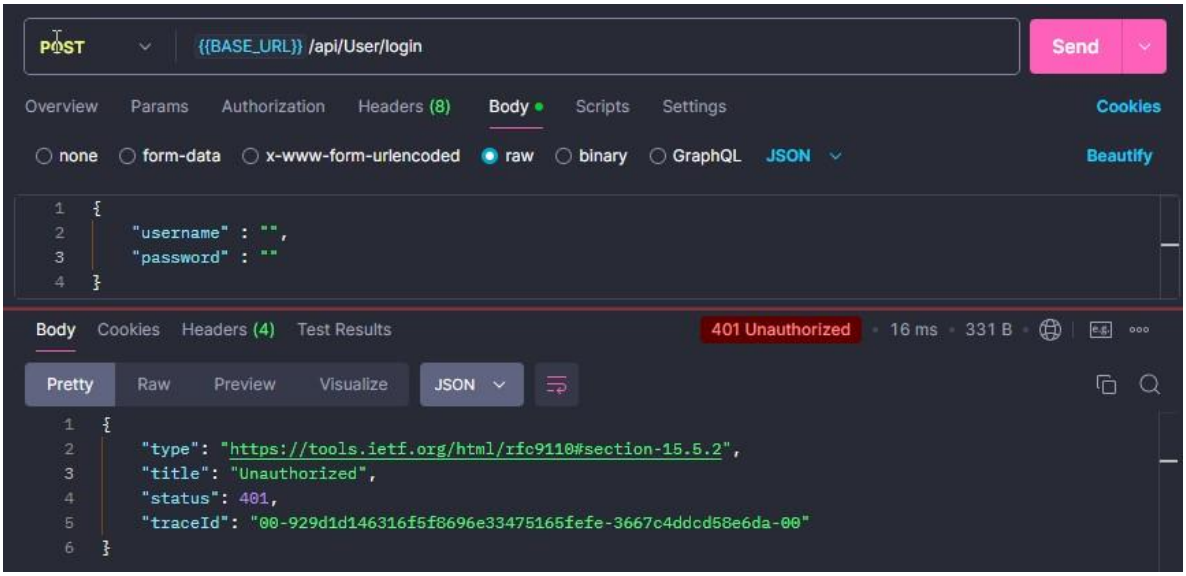
- **AU01- Verify login with valid credentials**



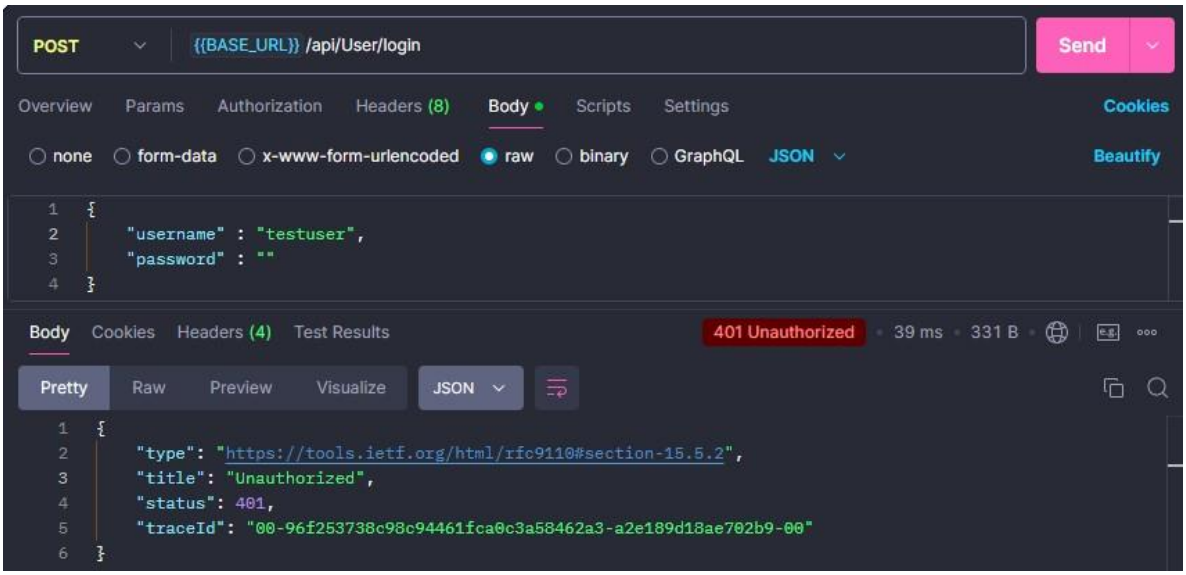
- **AU02 - Login attempt with wrong credentials**



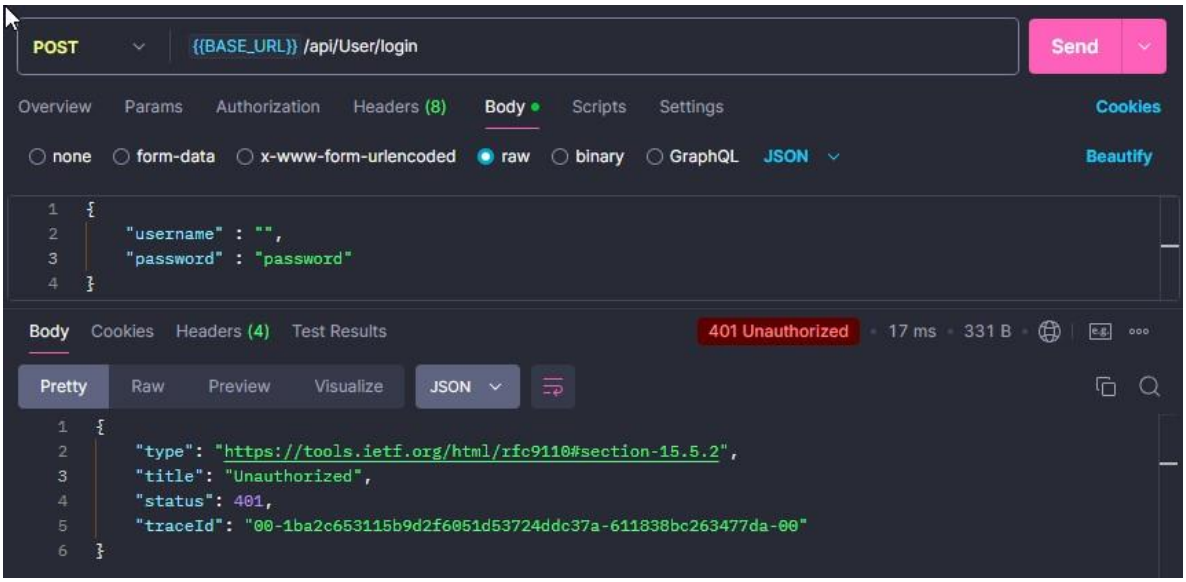
- **AU03 - Login with empty fields**



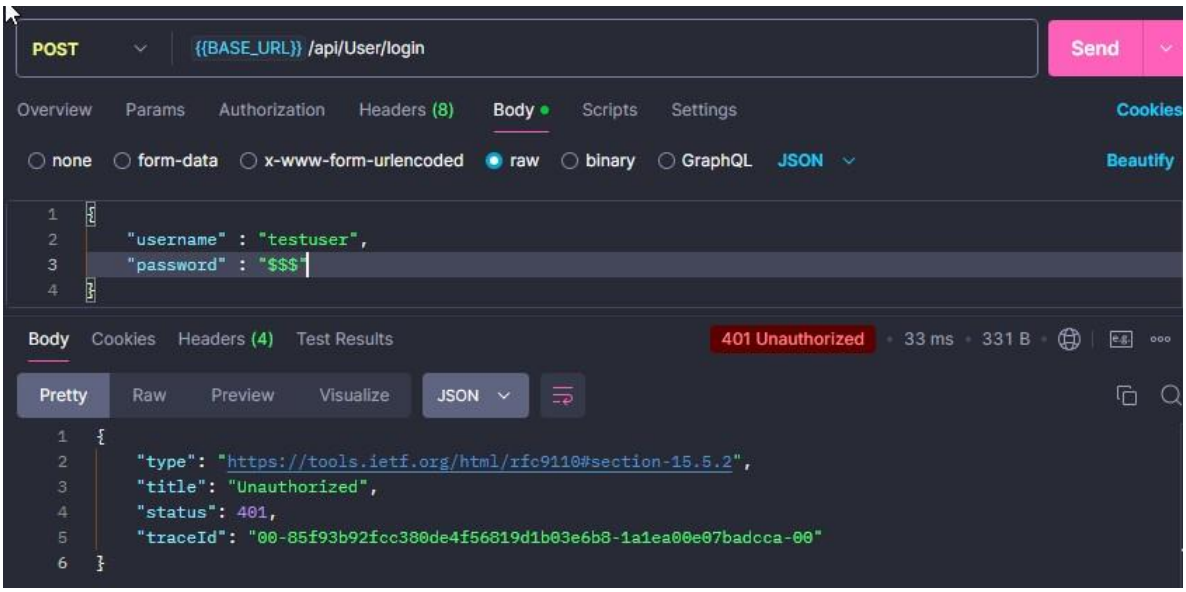
- **AU04- Login with correct username and empty password field.**



- **AU05 - Login with correct password and empty username field**

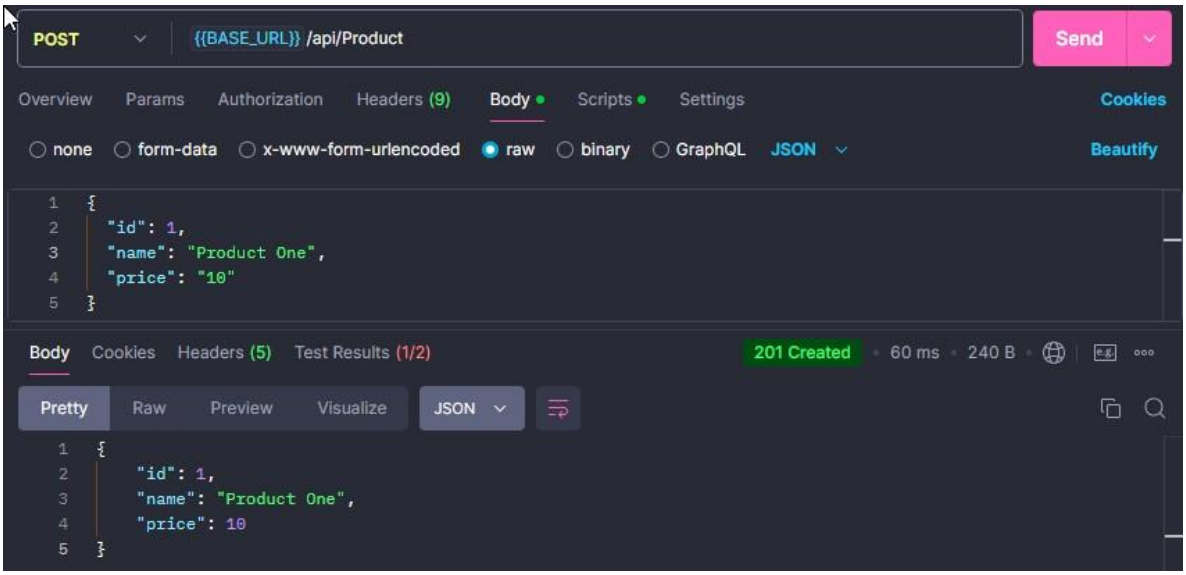


- **AU06 - Login attempt with invalid JSON formatting**

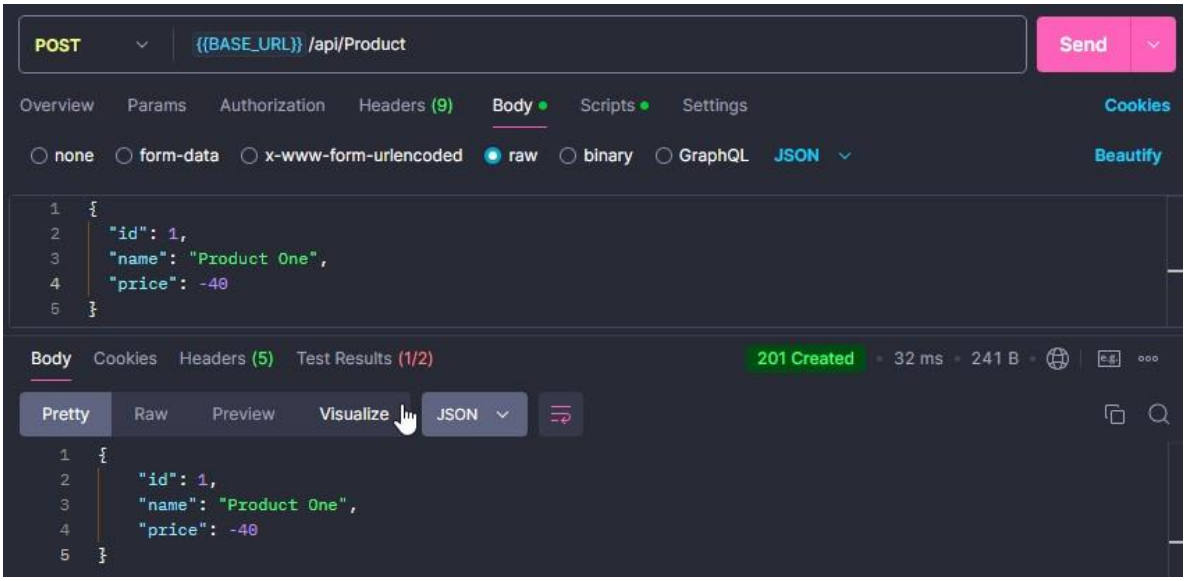


BACKEND (C# APIs): Product Management: (Create Product - POST)

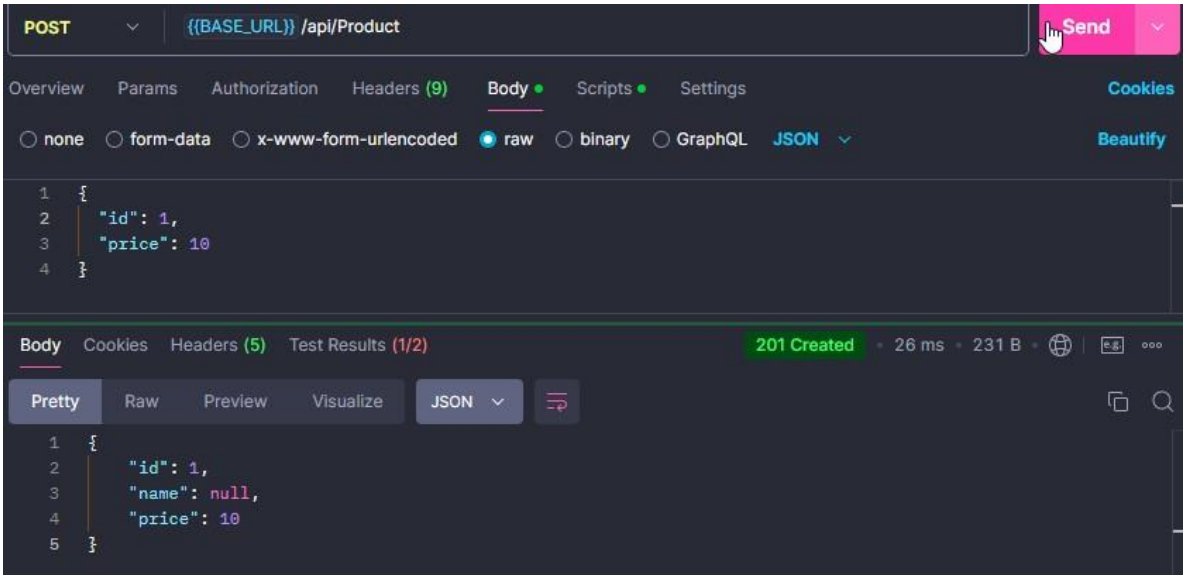
- **PC01-Create product with valid credentials**



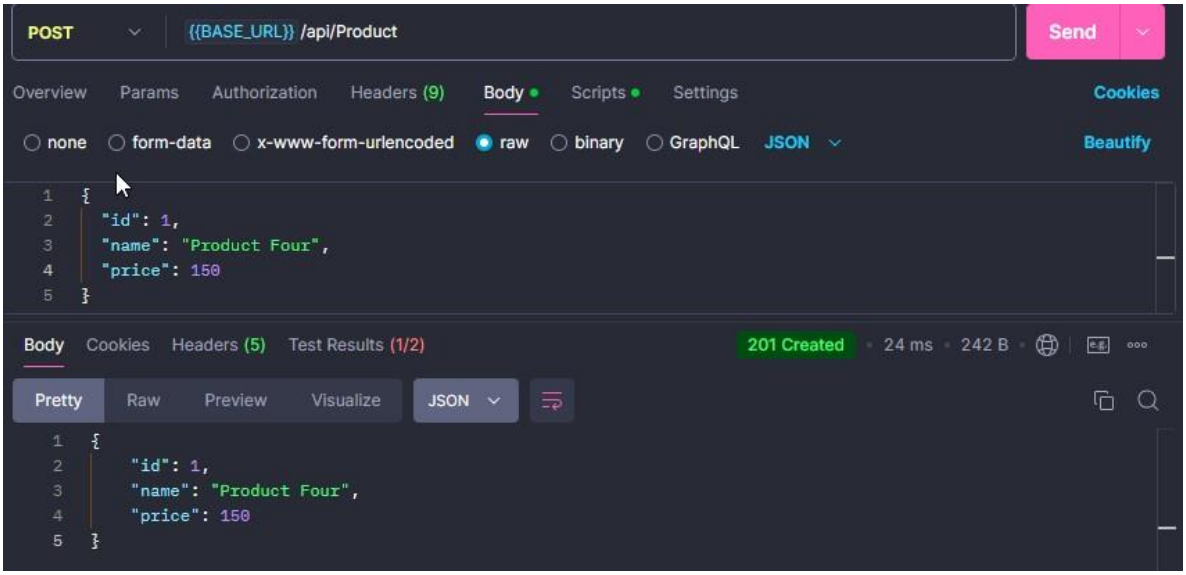
- **PC02- Create product with negative price**



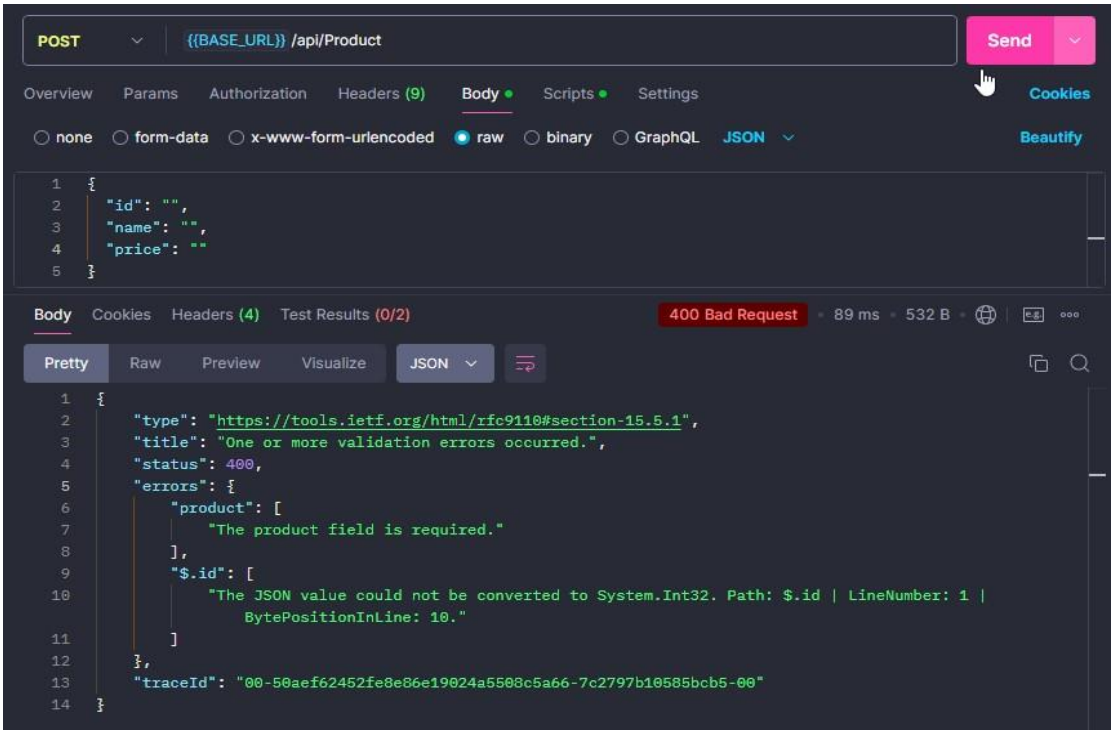
- **PC03-Create a product without the “name” field**



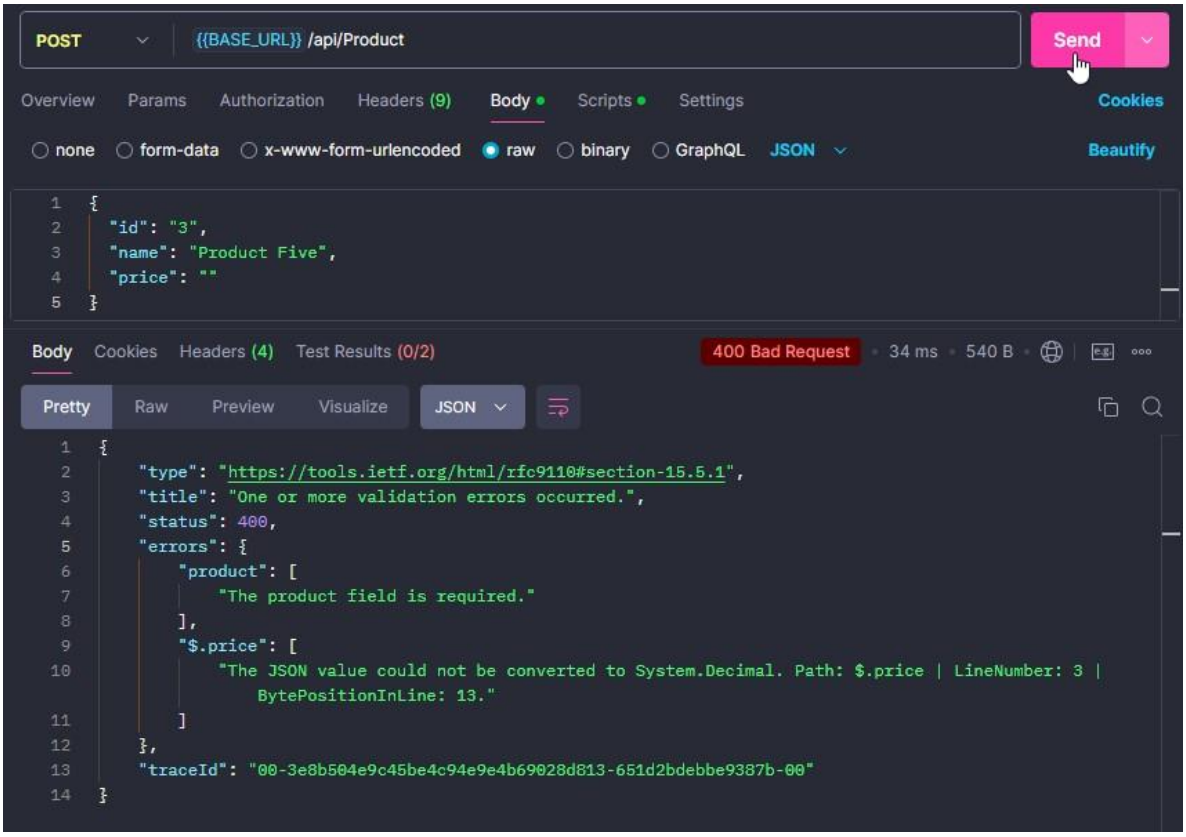
- **PC04-Attempting to create a product with a duplicate ID**



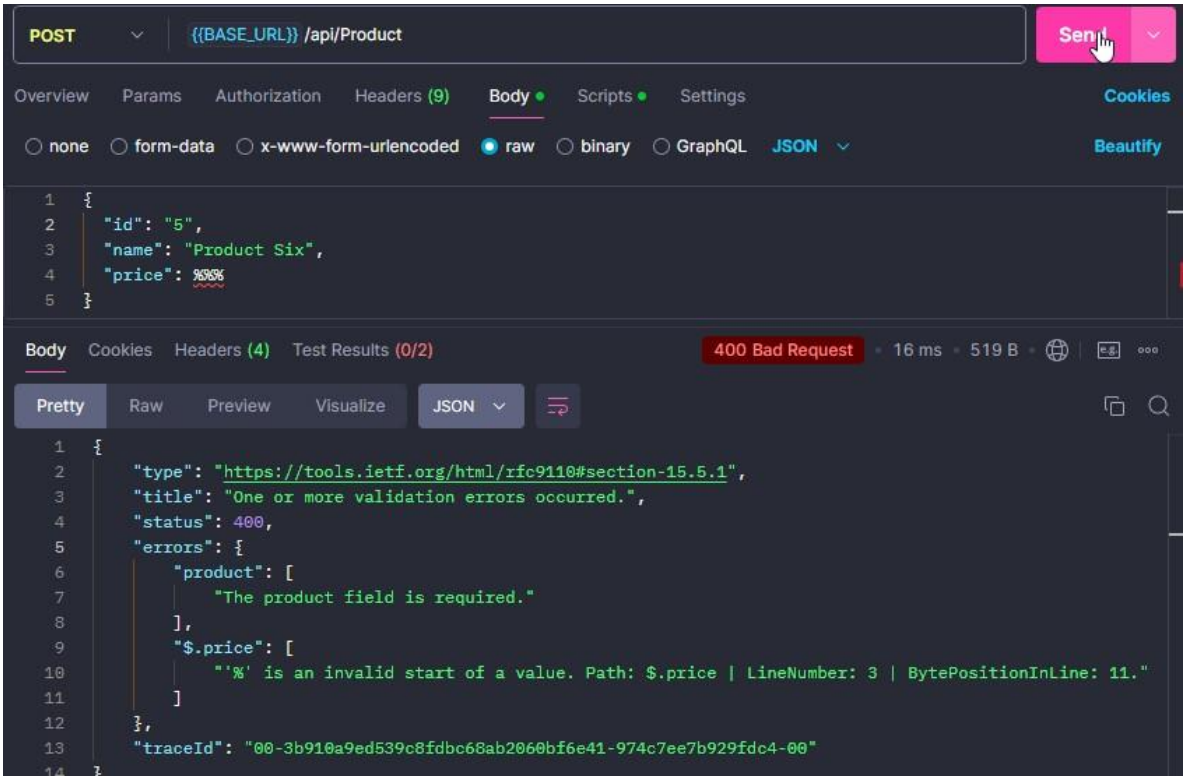
- **PC05- Create a product with empty fields**



- **PC06- Create a product without the “price” field**

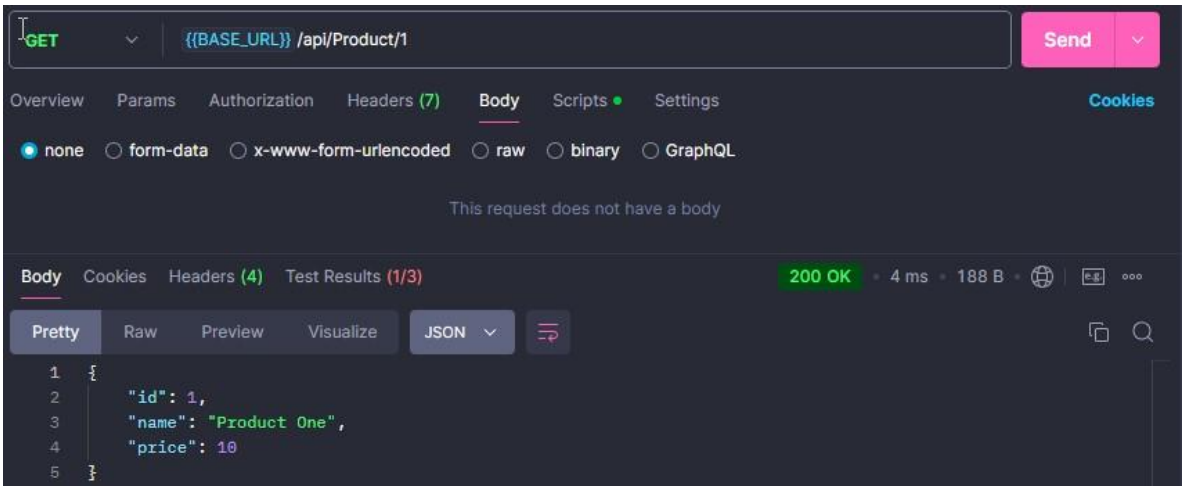


- **PC07- Create product with incorrectly formatted price**

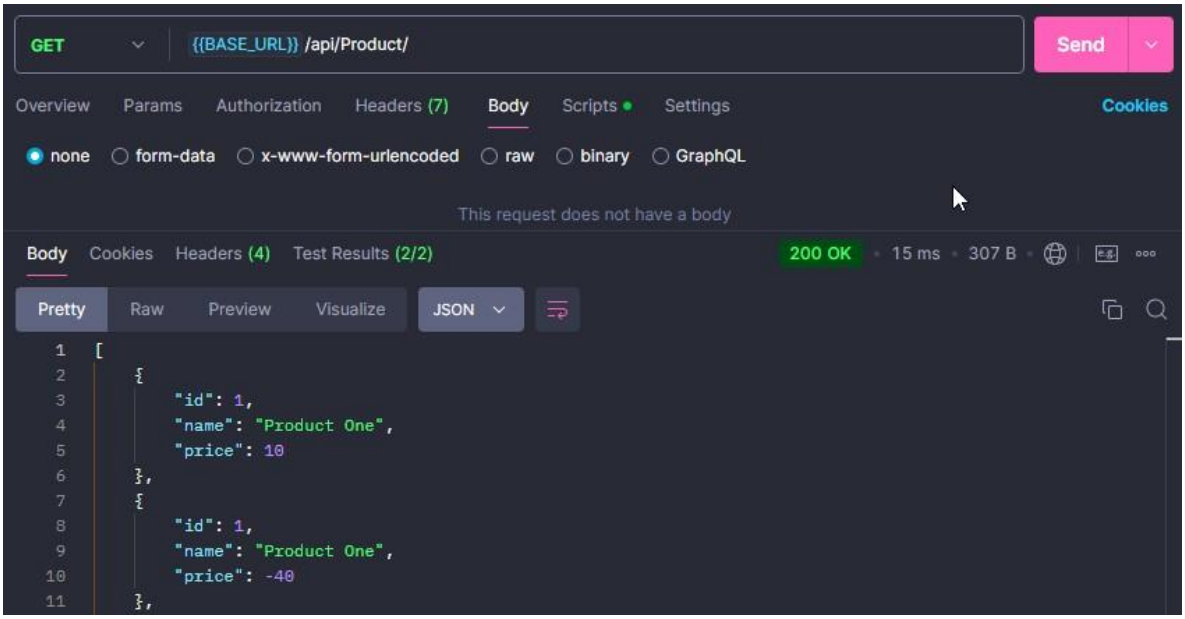


BACKEND (C# APIs): Product Management: (Reding Products - GET)

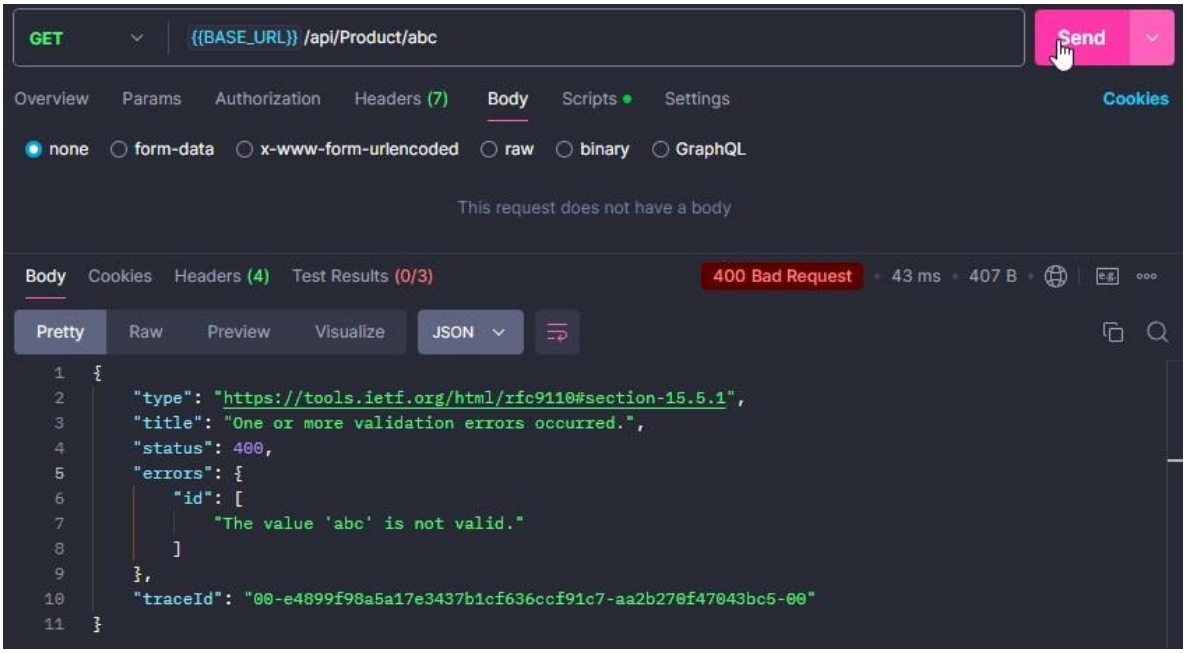
- **PC08- Get an existing product by ID**



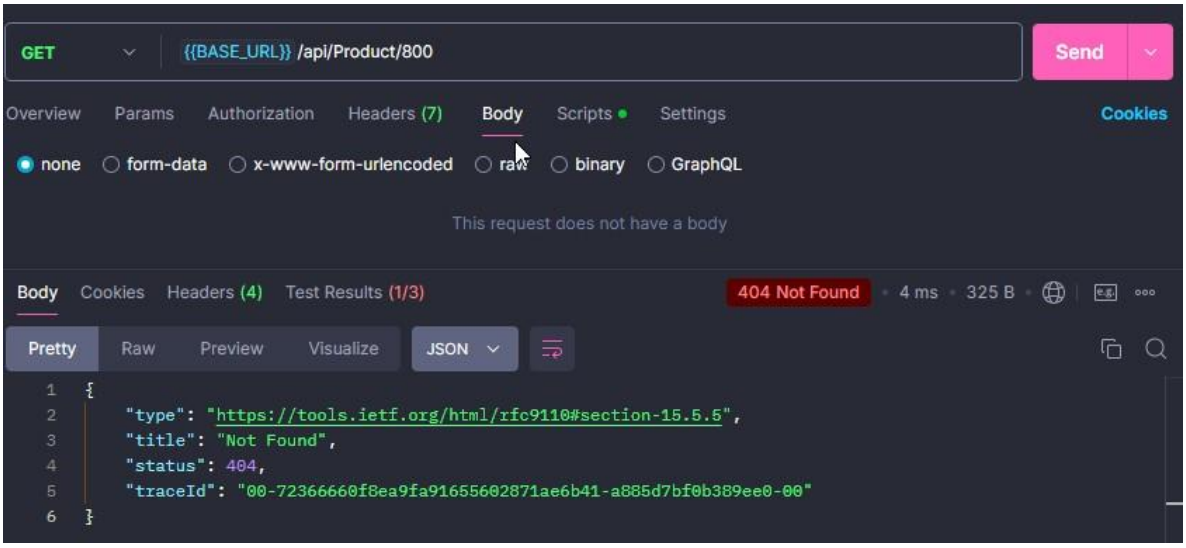
- **PC09- Get product list**



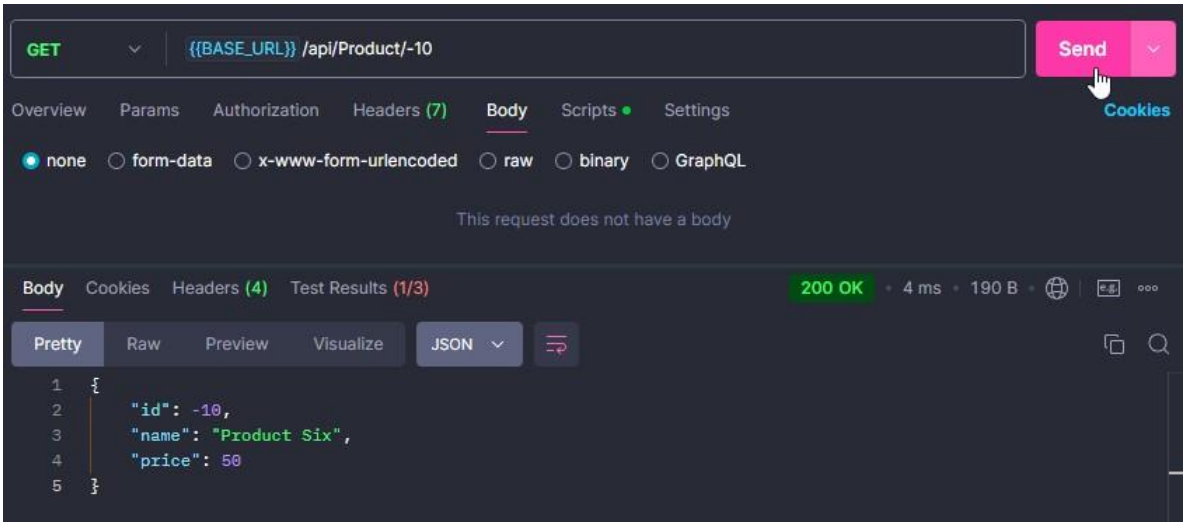
- **PC10- Get a product with invalid ID**



- **PC11- Get non-existing product**

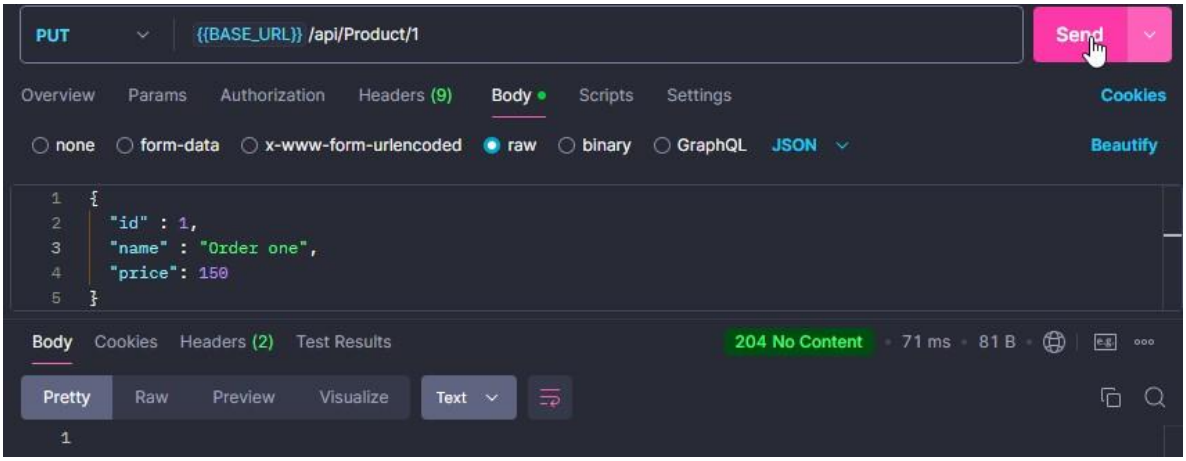


- **PC12- Get a product with negative id**

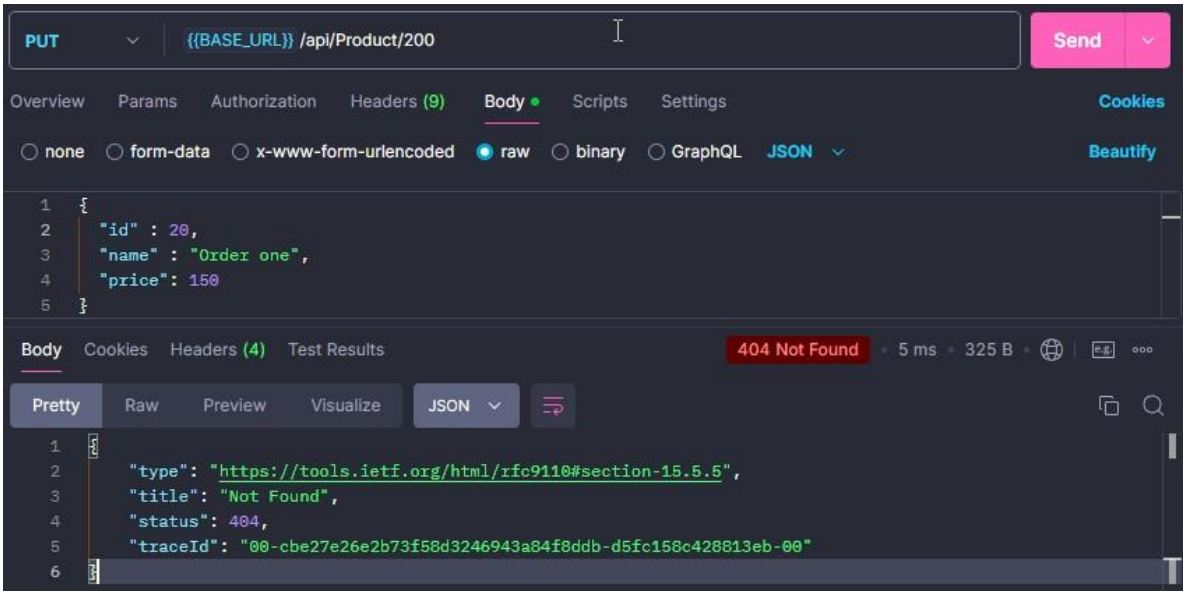


BACKEND (C# APIs): Product Management: (Update Products - PUT)

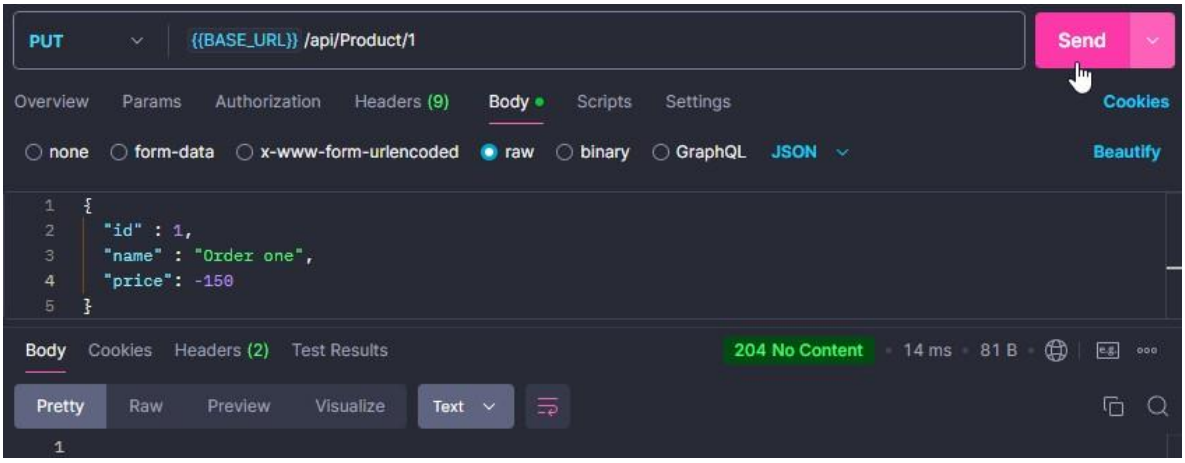
- **PC13- Update an existing product**



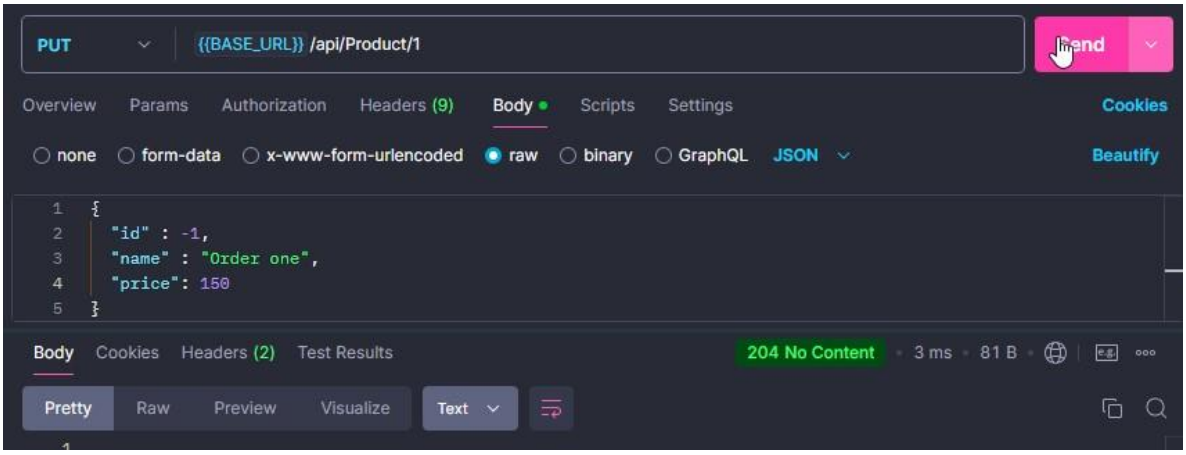
- **PC14- Update non-existent product**



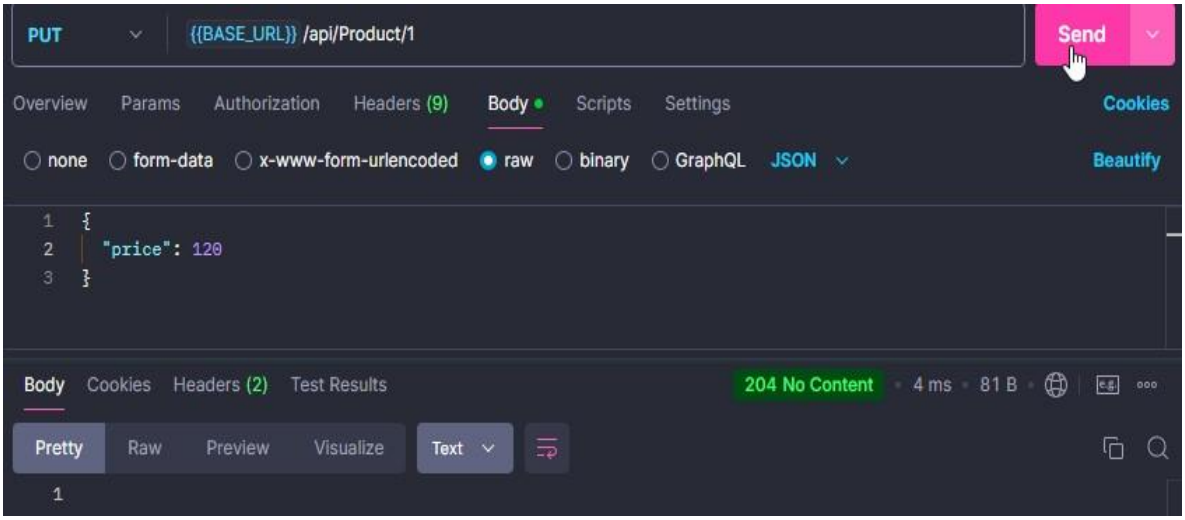
- **PC15- Update product with negative price**



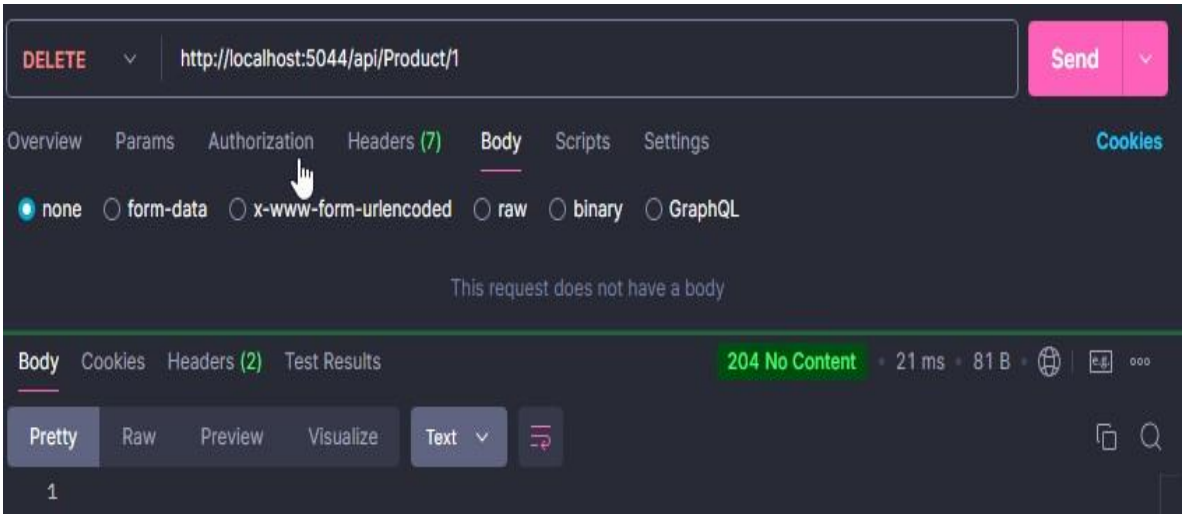
- **PC16- Update a product with negative ID**



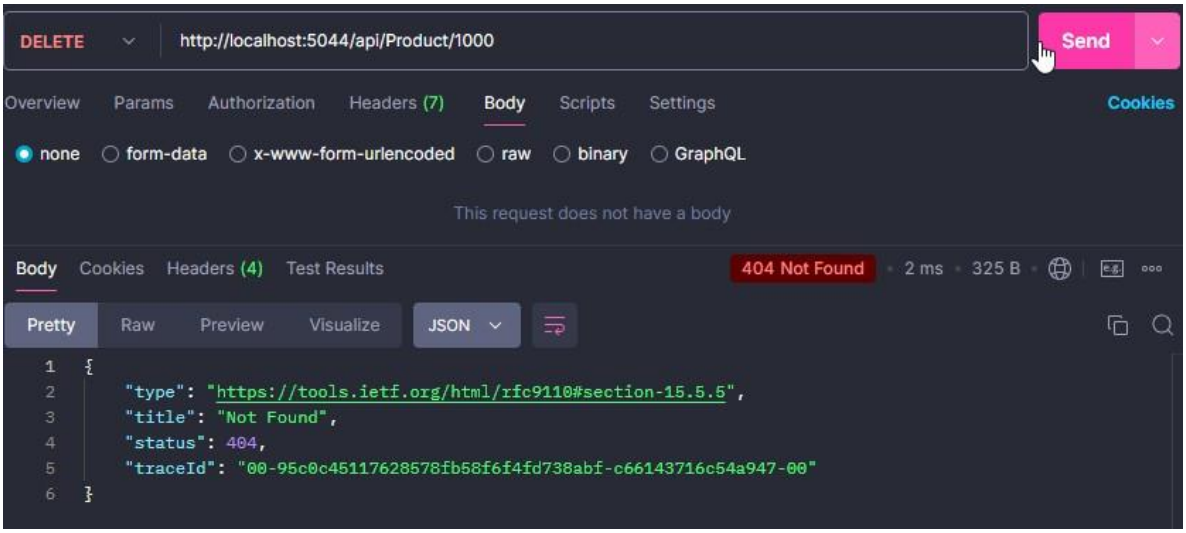
- **PC17- Update a product with missing fields**



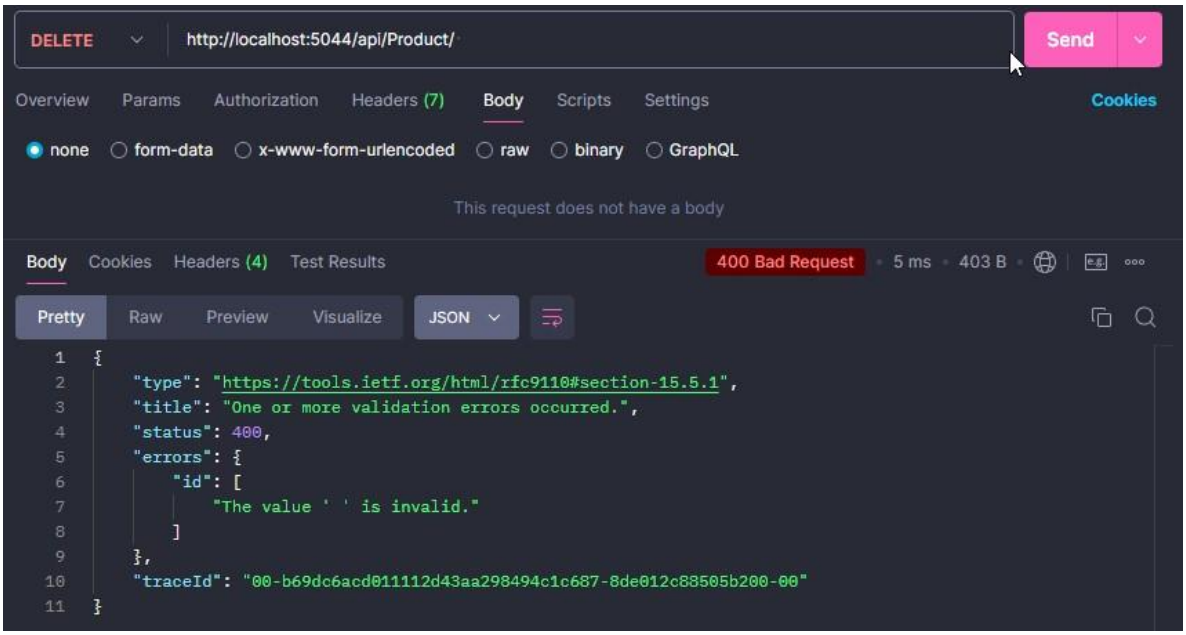
- **PC18- Delete an existing product**



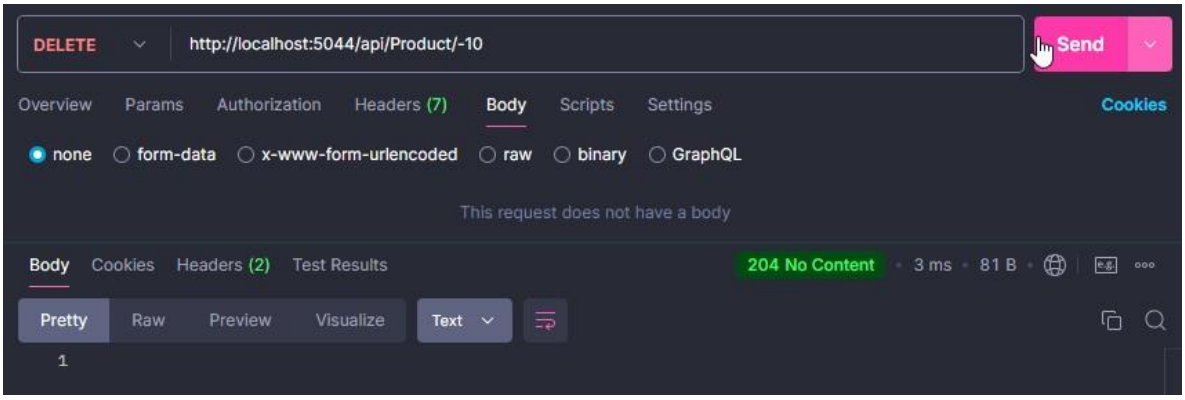
- **PC19- Delete a non-existing product**



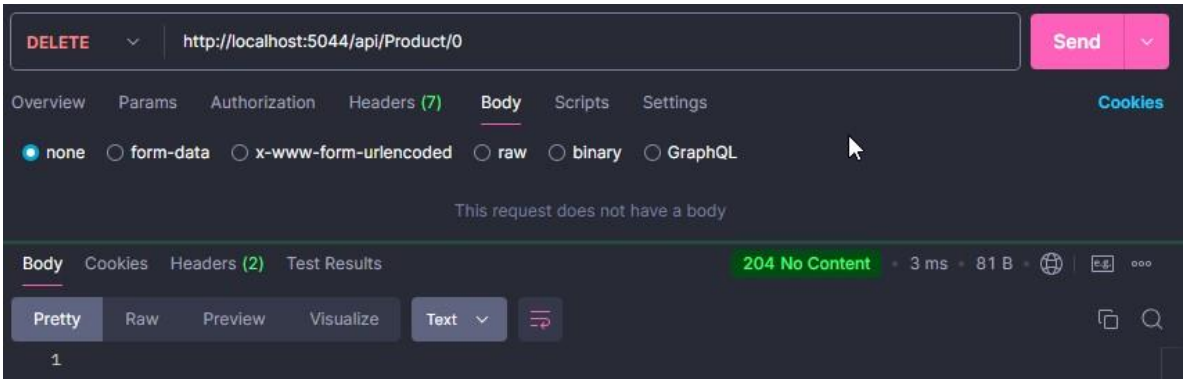
- **PC20- Delete a product without specifying an ID**



- **PC21- Delete a product with invalid negative ID**

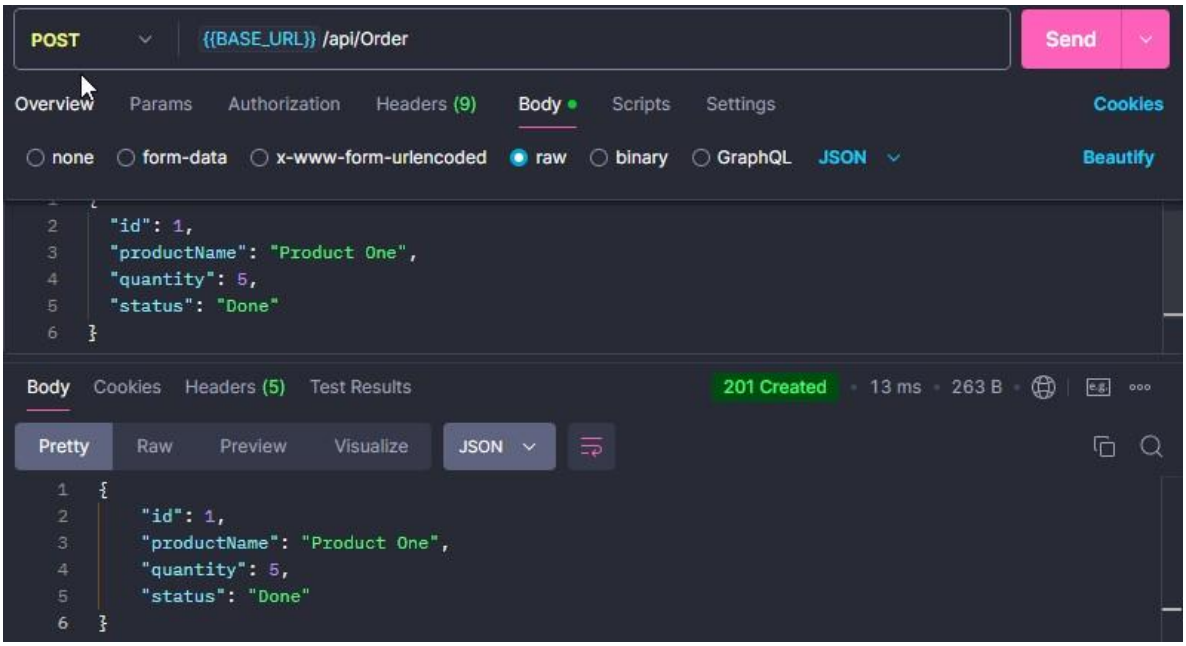


- **PC22- Delete a product with ID set to zero**

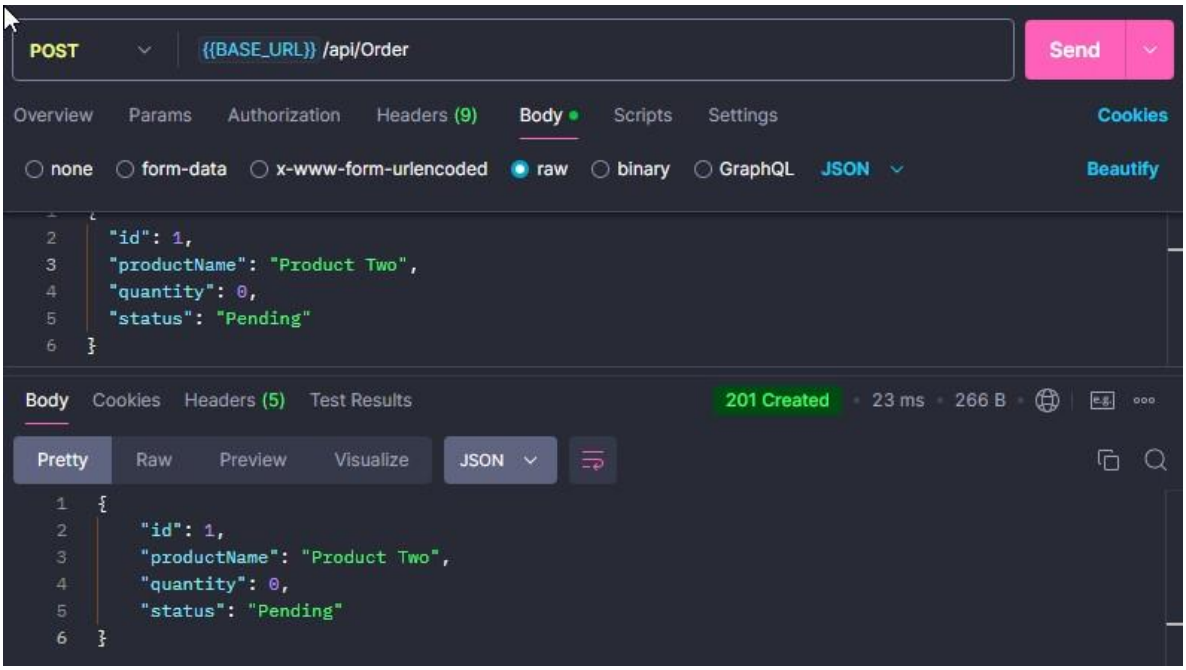


BACKEND (C# APIs): Order Processing: (Create Order- POST)

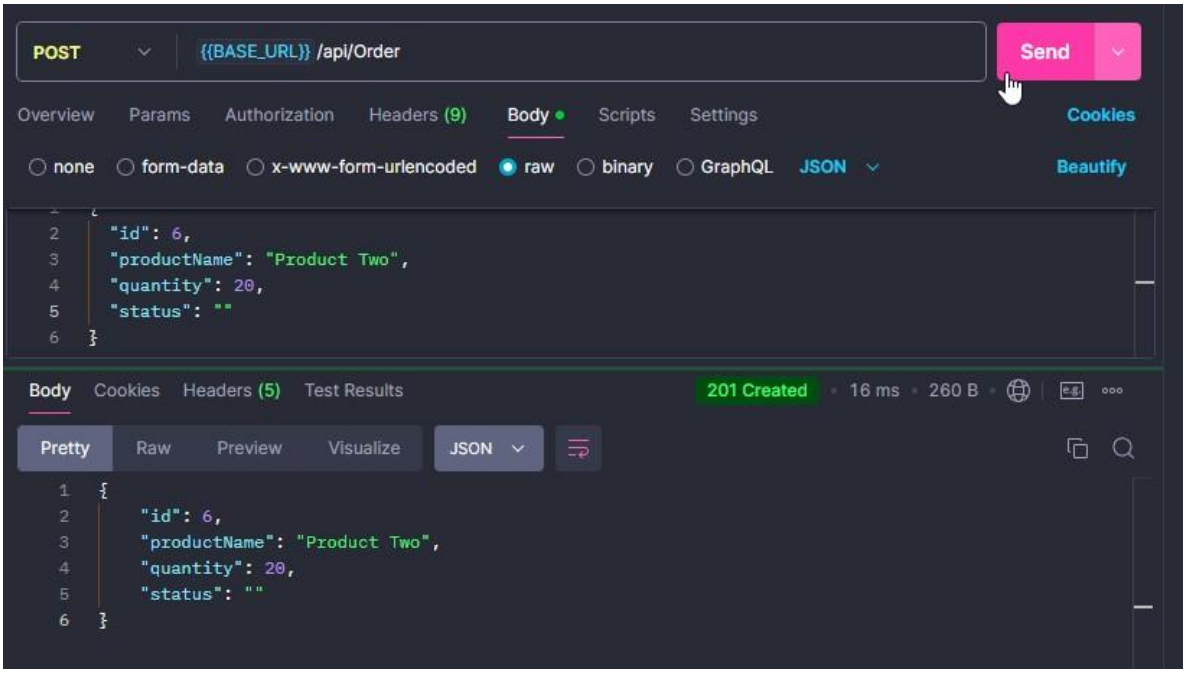
- **OC01- Create order with valid data**



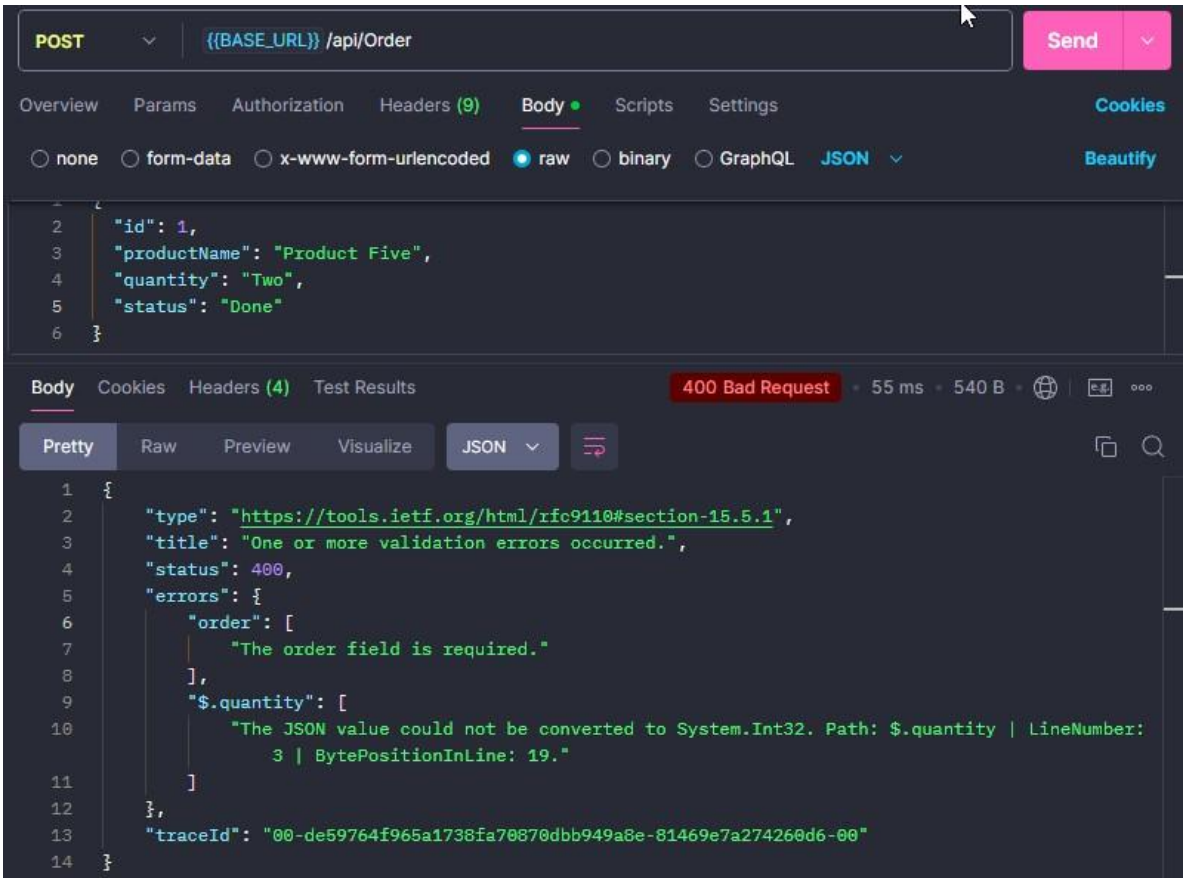
- **OC02- Create an order with zero quantity**



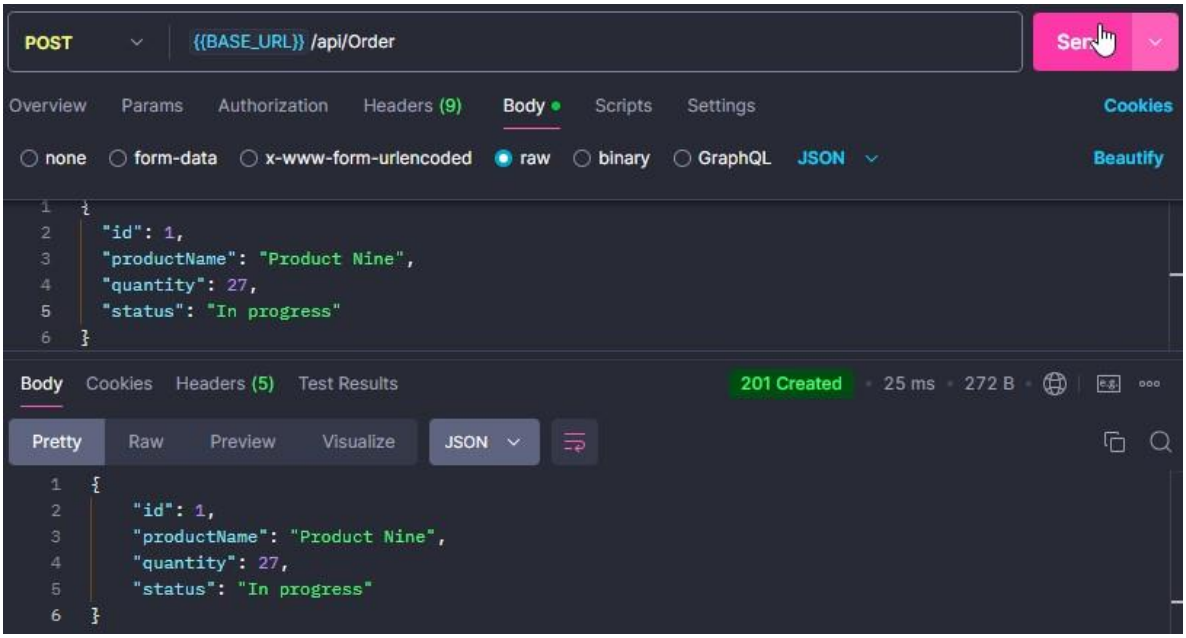
- **OC03- Create order with missing status**



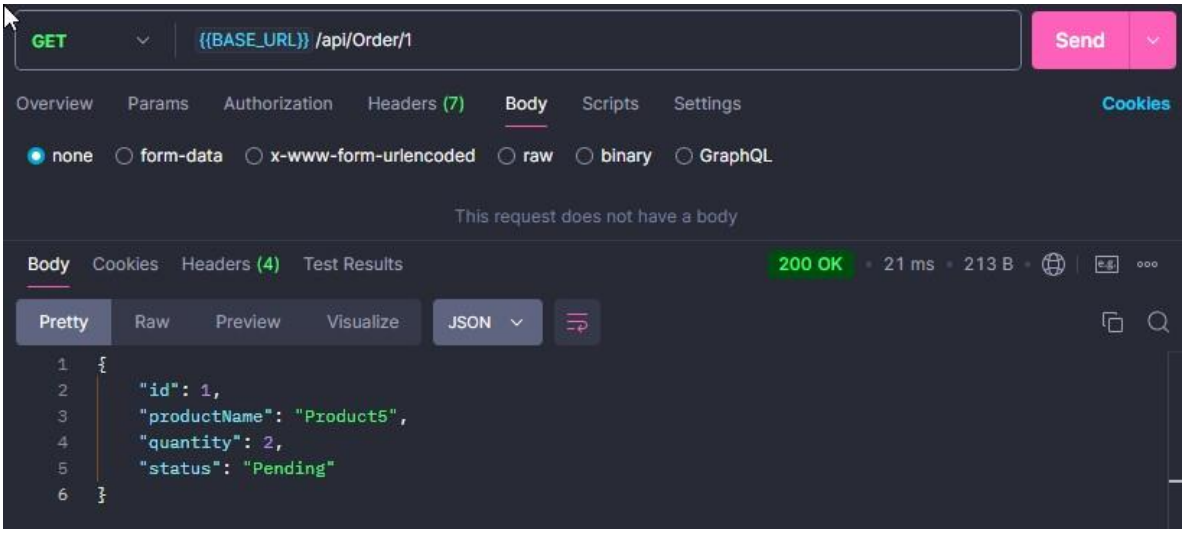
- **OC04-Create order with Invalid data types in the fields**



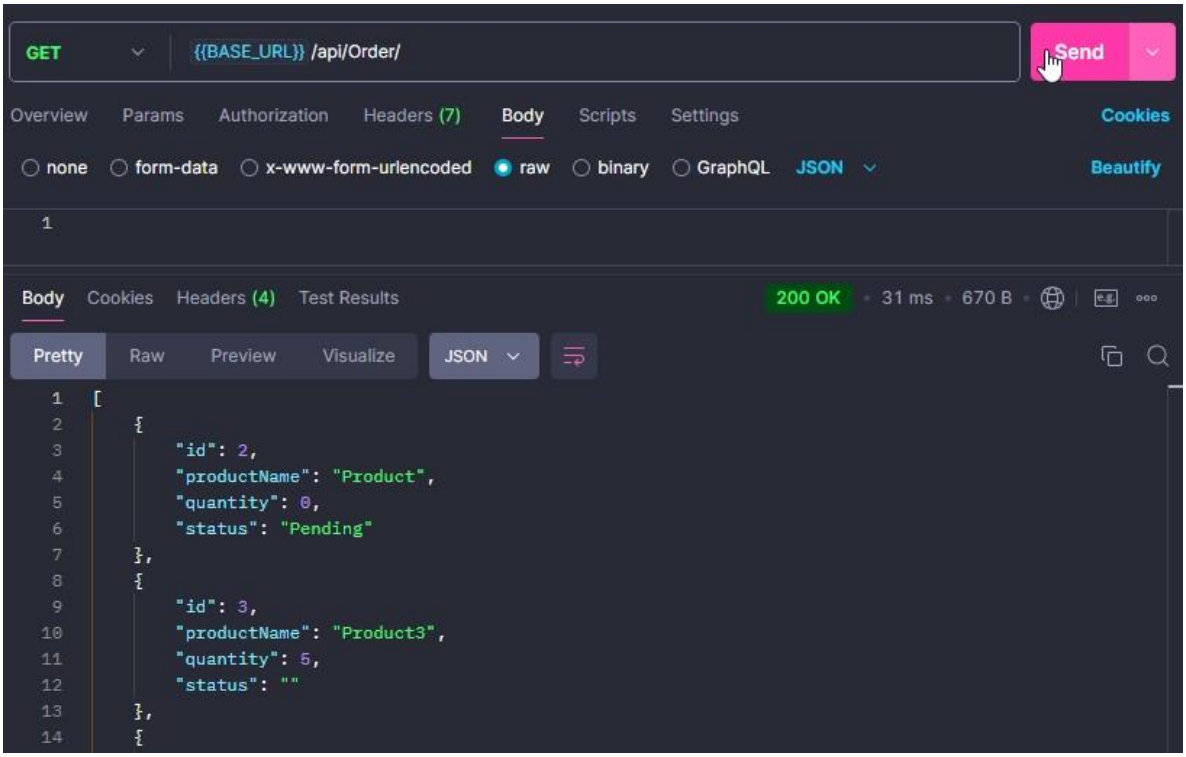
- **OC05- Create order with an existing order ID**



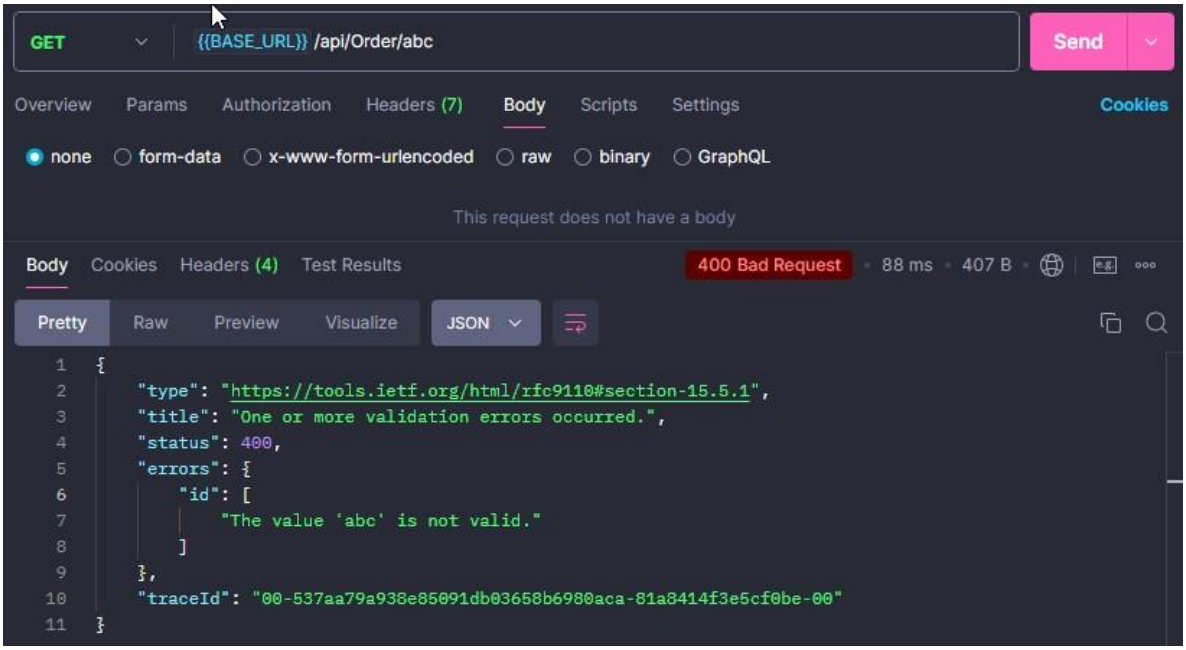
- **OC06- Get an existing Order by ID**



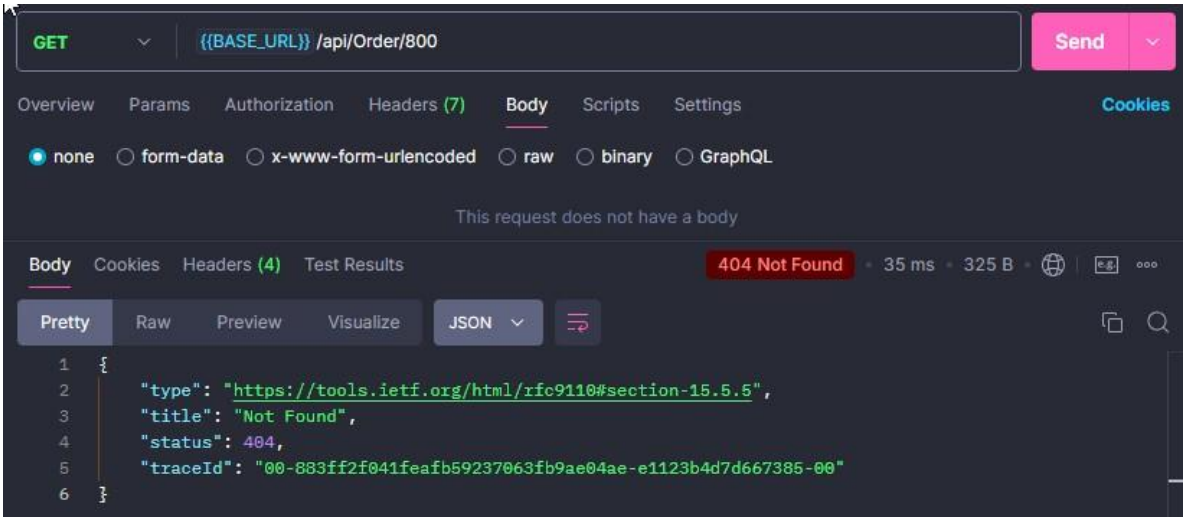
- **OC07- Get Order list**



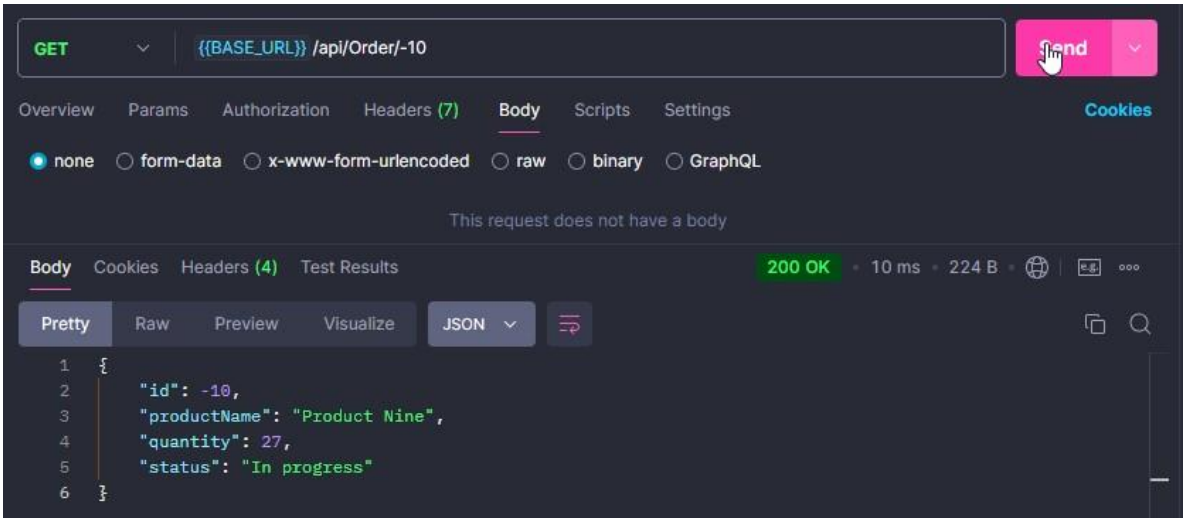
- **OC08- Get an order with invalid ID**



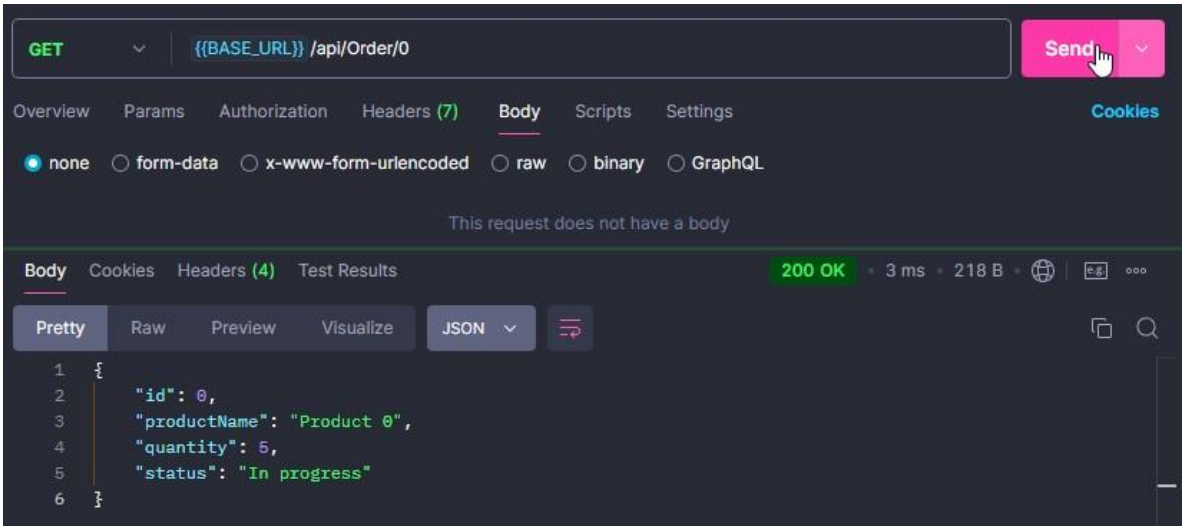
- **OC09- Get non-existing order**



- **OC10 - Get an order with negative id**

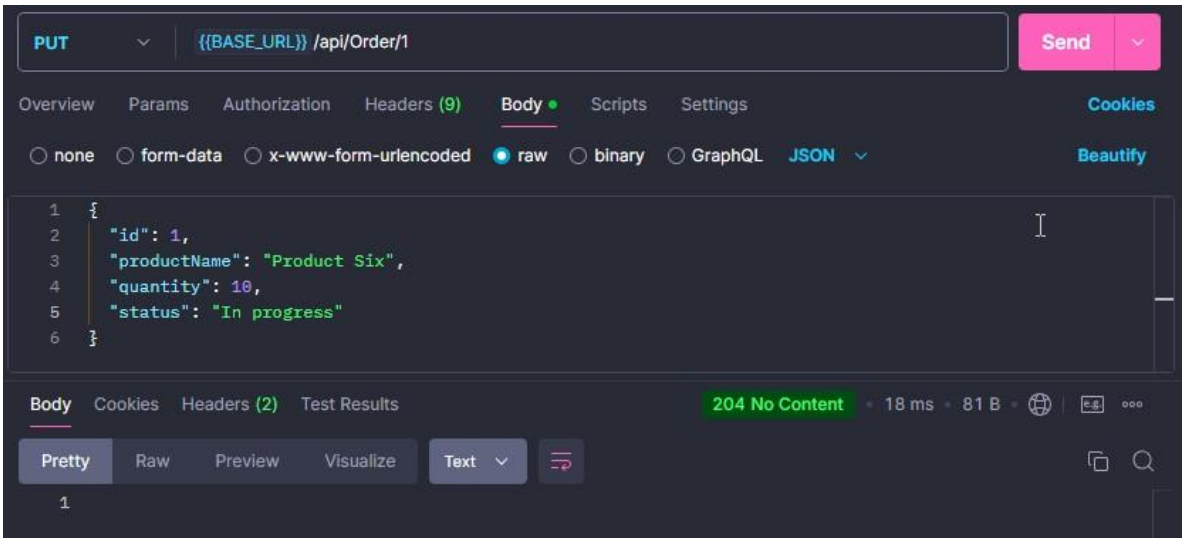


- **OC11- Get an order with id=0**

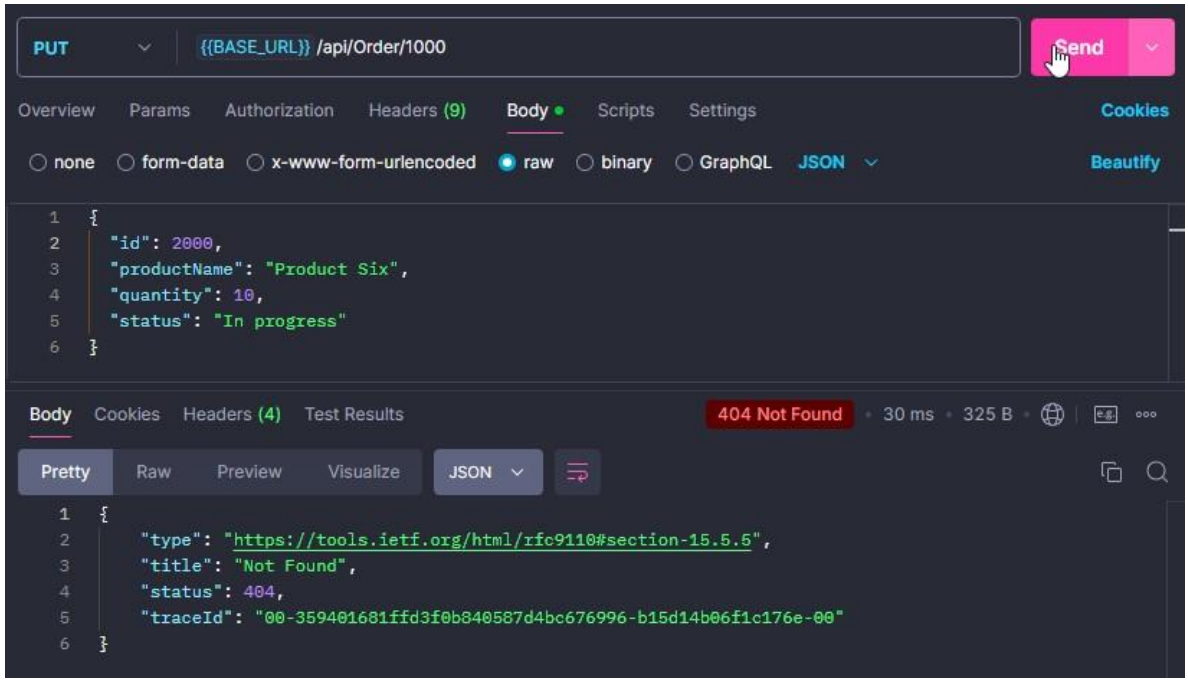


BACKEND (C# APIs): Order Processing: (Updated Order- PUT)

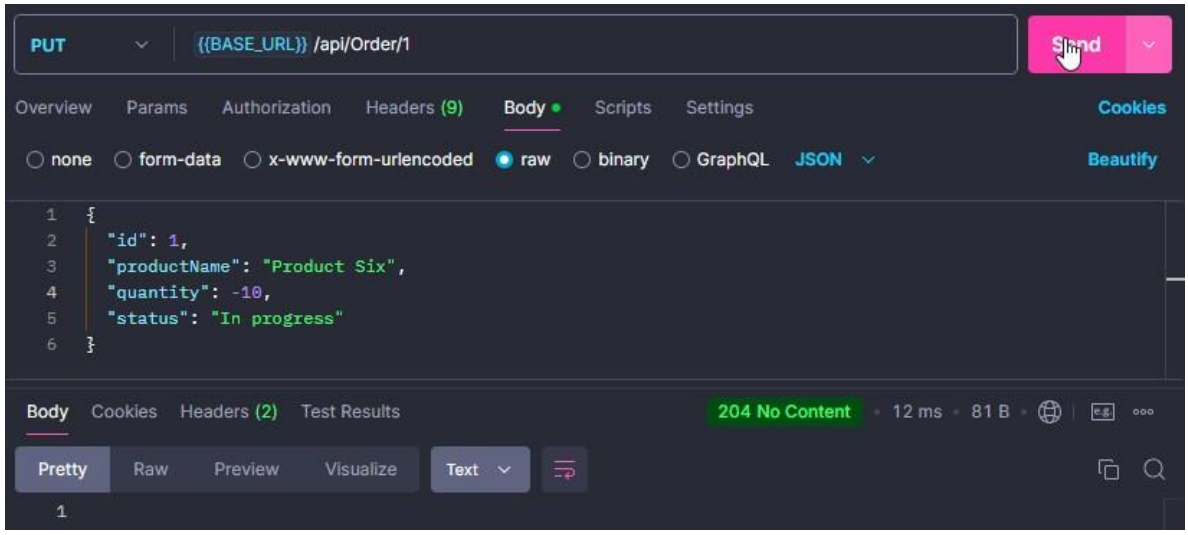
- **OC12- Update an existing order**



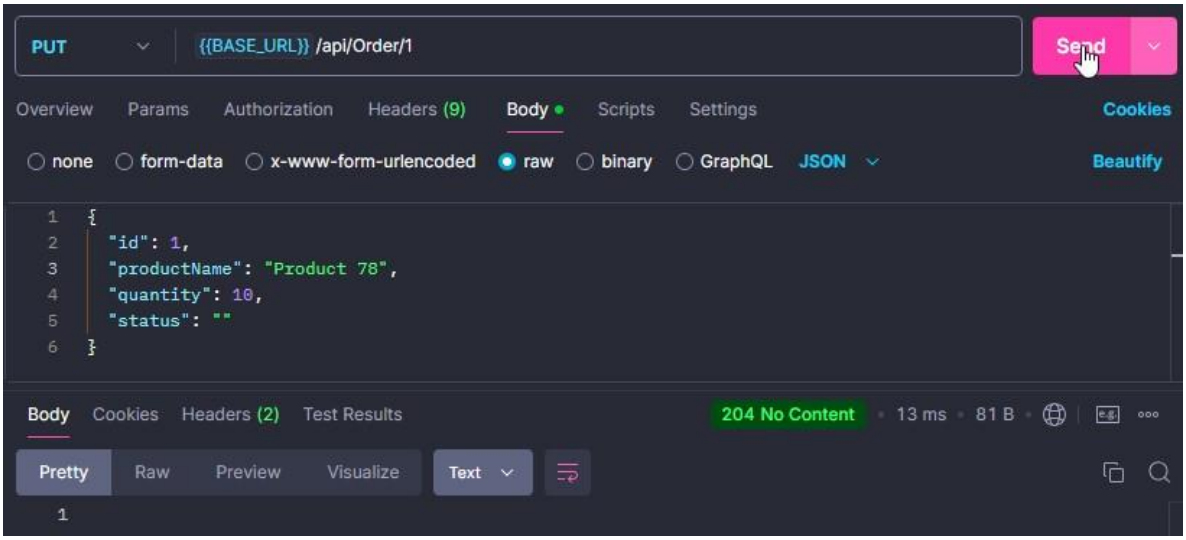
- **OC13- Update non-existent order**



- **OC14- Update an order with negative quantity**

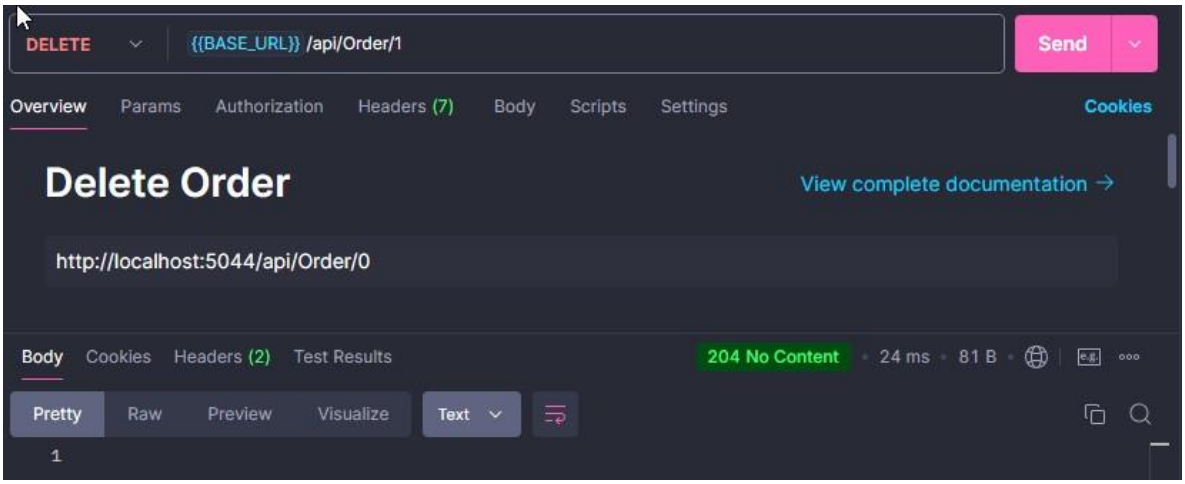


- **OC15- Update an order with missing fields**

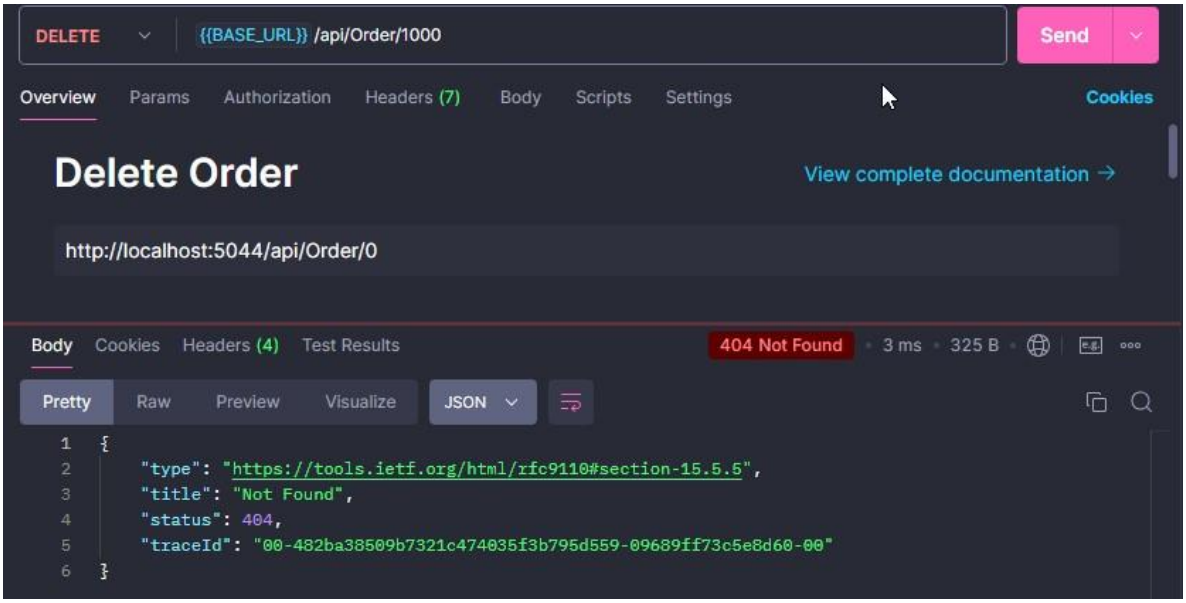


BACKEND (C# APIs): Order Processing: (Delete Order- DELETE)

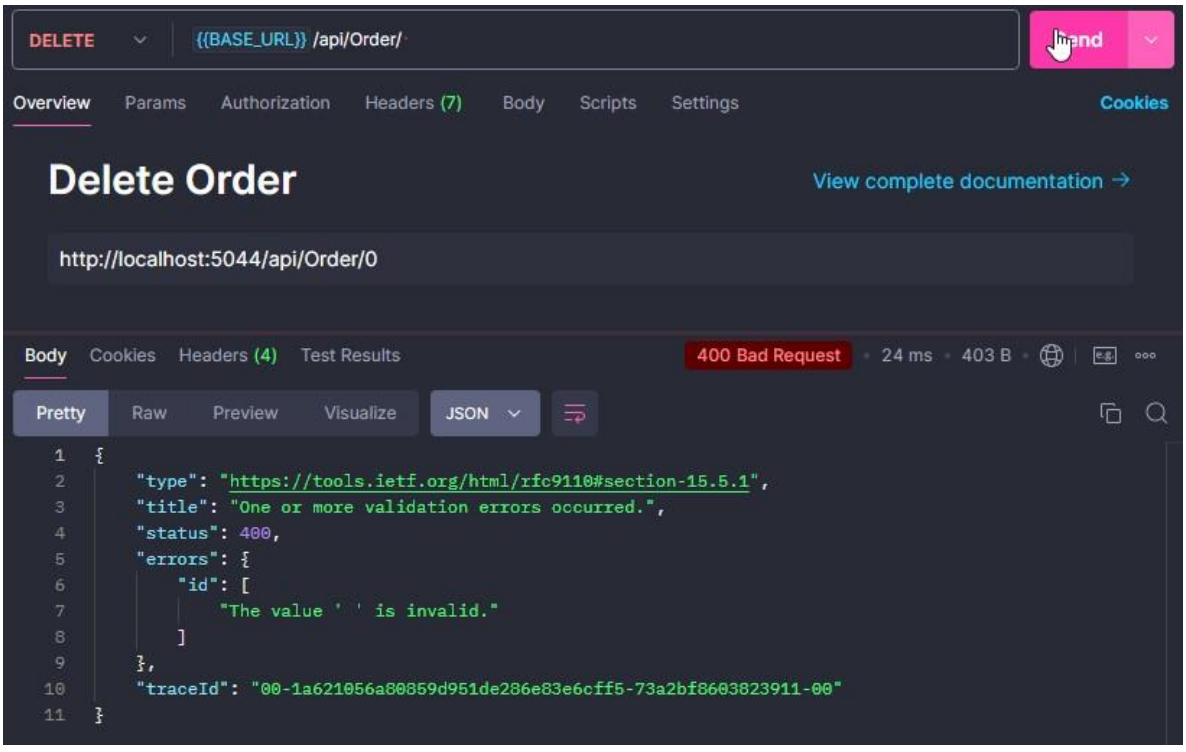
- **OC16- Delete an existing order**



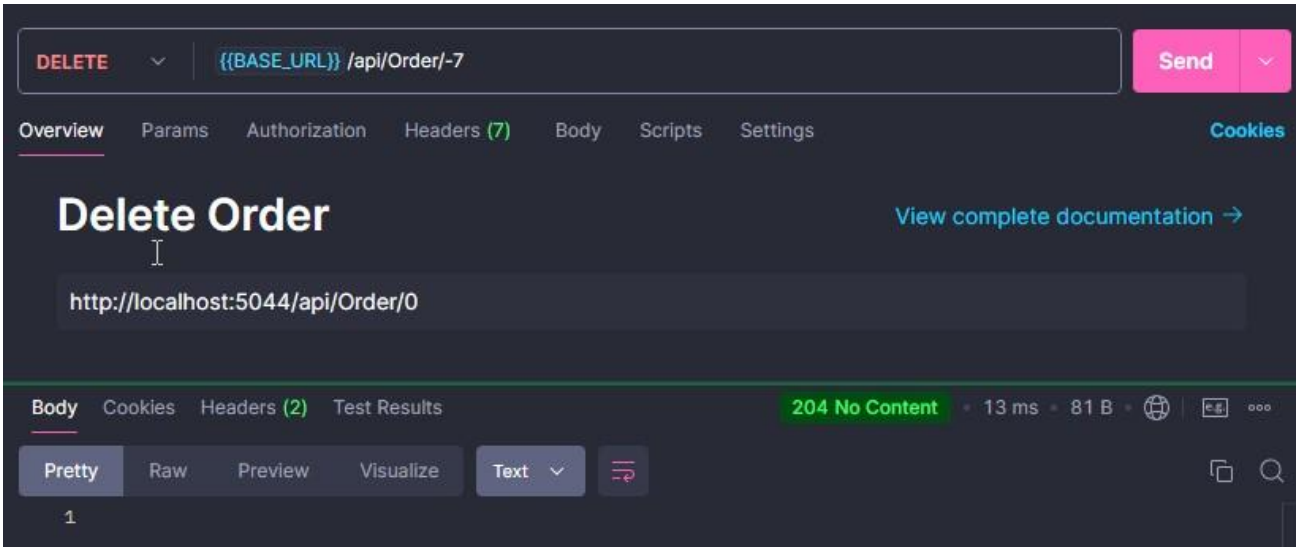
- **OC17- Delete a non-existing order**



- **OC18- Delete an order without specifying an ID**



- **OC19- Delete an order with invalid negative ID**



- **OC20- Delete an order with ID set to zero**