

TEST EXECUTION AND REPORTING

BACKEND (C# APIs):

U1 - Login with Valid Credentials

The screenshot shows a REST client interface with a POST request to `{{BASE_URL}}/api/User/login`. The request body is a JSON object: `{ "username": "testuser", "password": "password" }`. The response is a 200 OK status with a response time of 85 ms and a body size of 171 B. The response body is a JSON object: `{ "token": "sampletoken" }`.

```
POST {{BASE_URL}}/api/User/login
```

Overview Params Authorization Headers (8) Body Scripts Settings Cookies Beautify

☐ none ☐ form-data ☐ x-www-form-urlencoded ☒ raw ☐ binary ☐ GraphQL JSON

```
1 {
2   "username": "testuser",
3   "password": "password"
4 }
```

Body Cookies Headers (4) Test Results 200 OK • 85 ms • 171 B

Pretty Raw Preview Visualize JSON

```
1 {
2   "token": "sampletoken"
3 }
```

U2 - Test with incorrect username and password

The screenshot shows a REST client interface with a POST request to `{{BASE_URL}}/api/User/login`. The request body is a JSON object: `{ "username": "testuser2", "password": "password2" }`. The response is a 401 Unauthorized status with a response time of 59 ms and a body size of 331 B. The response body is a JSON object: `{ "type": "https://tools.ietf.org/html/rfc9110#section-15.5.2", "title": "Unauthorized", "status": 401, "traceId": "00-ff58012f4f0079045f4237739ceca520-5bbbce57db7f6bea-00" }`.

```
POST {{BASE_URL}}/api/User/login
```

Overview Params Authorization Headers (8) Body Scripts Settings Cookies Beautify

☐ none ☐ form-data ☐ x-www-form-urlencoded ☒ raw ☐ binary ☐ GraphQL JSON

```
1 {
2   "username": "testuser2",
3   "password": "password2"
4 }
```

Body Cookies Headers (4) Test Results 401 Unauthorized • 59 ms • 331 B

Pretty Raw Preview Visualize JSON

```
1 {
2   "type": "https://tools.ietf.org/html/rfc9110#section-15.5.2",
3   "title": "Unauthorized",
4   "status": 401,
5   "traceId": "00-ff58012f4f0079045f4237739ceca520-5bbbce57db7f6bea-00"
6 }
```

U3 - Login with Valid Username and Empty Password

The screenshot shows a REST client interface for a POST request to `{{BASE_URL}}/api/User/login`. The request body is a JSON object: `{ "username": "testuser", "password": "" }`. The response is a 401 Unauthorized status with a response time of 18 ms and a body size of 331 B. The response body is a JSON object: `{ "type": "https://tools.ietf.org/html/rfc9110#section-15.5.2", "title": "Unauthorized", "status": 401, "traceId": "00-b1d858a871d9be32cedb27f7e2258abd-c75ef1cefadea5cb-00" }`.

```
POST {{BASE_URL}}/api/User/login
```

Overview Params Authorization Headers (8) Body Scripts Settings Cookies

☐ none ☐ form-data ☐ x-www-form-urlencoded ☒ raw ☐ binary ☐ GraphQL JSON Beautify

```
1 {
2   "username": "testuser",
3   "password": ""
4 }
```

Body Cookies Headers (4) Test Results 401 Unauthorized • 18 ms • 331 B

Pretty Raw Preview Visualize JSON

```
1 {
2   "type": "https://tools.ietf.org/html/rfc9110#section-15.5.2",
3   "title": "Unauthorized",
4   "status": 401,
5   "traceId": "00-b1d858a871d9be32cedb27f7e2258abd-c75ef1cefadea5cb-00"
6 }
```

U4 - Login with Valid Password and Empty Username

The screenshot shows a REST client interface for a POST request to `{{BASE_URL}}/api/User/login`. The request body is a JSON object: `{ "username": "", "password": "password" }`. The response is a 401 Unauthorized status with a response time of 35 ms and a body size of 331 B. The response body is a JSON object: `{ "type": "https://tools.ietf.org/html/rfc9110#section-15.5.2", "title": "Unauthorized", "status": 401, "traceId": "00-0bc88c29c456298729a945af9ee382b6-eb901010c3427cae-00" }`.

```
POST {{BASE_URL}}/api/User/login
```

Overview Params Authorization Headers (8) Body Scripts Settings Cookies

☐ none ☐ form-data ☐ x-www-form-urlencoded ☒ raw ☐ binary ☐ GraphQL JSON Beautify

```
1 {
2   "username": "",
3   "password": "password"
4 }
```

Body Cookies Headers (4) Test Results 401 Unauthorized • 35 ms • 331 B

Pretty Raw Preview Visualize JSON

```
1 {
2   "type": "https://tools.ietf.org/html/rfc9110#section-15.5.2",
3   "title": "Unauthorized",
4   "status": 401,
5   "traceId": "00-0bc88c29c456298729a945af9ee382b6-eb901010c3427cae-00"
6 }
```

U5 - Login with Empty Fields

The screenshot shows a REST client interface. At the top, a POST request is configured to `{{BASE_URL}}/api/User/login`. The request body is a JSON object with empty fields: `{ "username": "", "password": "" }`. The response is a 401 Unauthorized status, with a response time of 19 ms and a body size of 331 B. The response body is displayed in JSON format:

```
1 {
2   "type": "https://tools.ietf.org/html/rfc9110#section-15.5.2",
3   "title": "Unauthorized",
4   "status": 401,
5   "traceId": "00-25d4165fcf3359d147924cce3303c543-958d39df98bf8a08-00"
6 }
```

U6 - Login with Missing Fields

The screenshot shows a REST client interface. At the top, a POST request is configured to `{{BASE_URL}}/api/User/login`. The request body is a JSON object with a missing password field: `{ "username": "testuser" }`. The response is a 400 Bad Request status, with a response time of 181 ms and a body size of 555 B. The response body is displayed in JSON format:

```
1 {
2   "type": "https://tools.ietf.org/html/rfc9110#section-15.5.1",
3   "title": "One or more validation errors occurred.",
4   "status": 400,
5   "errors": {
6     "$": [
7       "JSON deserialization for type 'SampleAPI.Controllers.LoginRequest' was missing required
7         properties, including the following: password"
8     ],
9     "request": [
10      "The request field is required."
11    ]
12  },
13   "traceId": "00-3c82ac66097f074323047c7ead338621-9d41358e505079a6-00"
14 }
```

U7 - Login with Invalid Payload

POST ▼ `{{BASE_URL}}/api/User/login` Send ▼

Overview Params Authorization Headers (8) **Body** ● Scripts Settings Cookies

☐ none ☐ form-data ☐ x-www-form-urlencoded ☒ raw ☐ binary ☐ GraphQL JSON ▼ Beautify

```
1 {
2   "username": "testuser",
3   "password": 123
4 }
```

Body Cookies Headers (4) Test Results **400 Bad Request** • 219 ms • 545 B •

Pretty Raw Preview Visualize JSON ▼

```
1 {
2   "type": "https://tools.ietf.org/html/rfc9110#section-15.5.1",
3   "title": "One or more validation errors occurred.",
4   "status": 400,
5   "errors": {
6     "request": [
7       "The request field is required."
8     ],
9     "$.password": [
10      "The JSON value could not be converted to System.String. Path: $.password | LineNumber: 2 | BytePositionInLine: 20."
11    ]
12  },
13   "traceId": "00-3cb762e724e5a7fe9714c6b7a29f5654-9b438efceae17764-00"
14 }
```

P01 - Create a product with valid data

POST ▼ `{{BASE_URL}}/api/Product` Send ▼

Overview Params Authorization Headers (9) **Body** ● Scripts ● Settings Cookies

☐ none ☐ form-data ☐ x-www-form-urlencoded ☒ raw ☐ binary ☐ GraphQL JSON ▼ Beautify

```
1 {
2   "id": 1,
3   "name": "Product",
4   "price": "120"
5 }
```

Body Cookies Headers (5) Test Results (1/2) **201 Created** • 71 ms • 237 B •

Pretty Raw Preview Visualize JSON ▼

```
1 {
2   "id": 1,
3   "name": "Product",
4   "price": 120
5 }
```

P02 - Create a product with duplicate ID

POST ▼ | `{{BASE_URL}}/api/Product` Send ▼

Overview Params Authorization Headers (9) **Body** • Scripts • Settings Cookies

☐ none ☐ form-data ☐ x-www-form-urlencoded ☒ raw ☐ binary ☐ GraphQL JSON ▼ Beautify

```
1 {
2   "id": 1,
3   "name": "Product2",
4   "price": "30"
5 }
```

Body Cookies Headers (5) Test Results (1/2) 201 Created • 69 ms • 237 B •

Pretty Raw Preview Visualize JSON ▼

```
1 {
2   "id": 1,
3   "name": "Product2",
4   "price": 30
5 }
```

P03 - Create a product with id = 0

POST ▼ | `{{BASE_URL}}/api/Product` Send ▼

Overview Params Authorization Headers (9) **Body** • Scripts • Settings Cookies

☐ none ☐ form-data ☐ x-www-form-urlencoded ☒ raw ☐ binary ☐ GraphQL JSON ▼ Beautify

```
1 {
2   "id": 0,
3   "name": "Product",
4   "price": "120"
5 }
```

Body Cookies Headers (5) Test Results (1/2) 201 Created • 17 ms • 237 B •

Pretty Raw Preview Visualize JSON ▼

```
1 {
2   "id": 0,
3   "name": "Product",
4   "price": 120
5 }
```

P04 - Create a product with negative id

The screenshot shows a REST client interface with a POST request to `{{BASE_URL}}/api/Product`. The request body is a JSON object: `{ "id": -1, "name": "Product", "price": 120 }`. The response status is `201 Created` with a response time of `188 ms` and a body size of `239 B`. The response body is displayed in the 'Pretty' view, showing the same JSON object: `{ "id": -1, "name": "Product", "price": 120 }`.

```
POST {{BASE_URL}}/api/Product
```

Overview Params Authorization Headers (9) Body Scripts Settings Cookies

☐ none ☐ form-data ☐ x-www-form-urlencoded ☒ raw ☐ binary ☐ GraphQL JSON

```
1 {
2   "id": -1,
3   "name": "Product",
4   "price": 120
5 }
```

Body Cookies Headers (5) Test Results (1/2) 201 Created • 188 ms • 239 B • [Globe] [E-S] [More]

Pretty Raw Preview Visualize JSON [More]

```
1 {
2   "id": -1,
3   "name": "Product",
4   "price": 120
5 }
```

P05 - Create a product without a name field

The screenshot shows a REST client interface with a POST request to `{{BASE_URL}}/api/Product`. The request body is a JSON object: `{ "id": 2, "price": 120 }`. The response status is `201 Created` with a response time of `18 ms` and a body size of `232 B`. The response body is displayed in the 'Pretty' view, showing the JSON object: `{ "id": 2, "name": null, "price": 120 }`.

```
POST {{BASE_URL}}/api/Product
```

Overview Params Authorization Headers (9) Body Scripts Settings Cookies

☐ none ☐ form-data ☐ x-www-form-urlencoded ☒ raw ☐ binary ☐ GraphQL JSON

```
1 {
2   "id": 2,
3   "price": 120
4 }
```

Body Cookies Headers (5) Test Results (1/2) 201 Created • 18 ms • 232 B • [Globe] [E-S] [More]

Pretty Raw Preview Visualize JSON [More]

```
1 {
2   "id": 2,
3   "name": null,
4   "price": 120
5 }
```

P06 - Create a product with negative price

The screenshot shows a REST client interface with a POST request to `{{BASE_URL}}/api/Product`. The request body is a JSON object: `{ "id": 3, "name": "Product", "price": -120 }`. The response status is `201 Created` with a response time of 20 ms and a body size of 238 B. The response body is displayed in a pretty-printed JSON format: `{ "id": 3, "name": "Product", "price": -120 }`.

```
POST {{BASE_URL}}/api/Product
```

Overview Params Authorization Headers (9) Body Scripts Settings Cookies

☐ none ☐ form-data ☐ x-www-form-urlencoded ☒ raw ☐ binary ☐ GraphQL JSON

```
1 {
2   "id": 3,
3   "name": "Product",
4   "price": -120
5 }
```

Body Cookies Headers (5) Test Results (1/2) 201 Created • 20 ms • 238 B • [Globe] [Copy] [More]

Pretty Raw Preview Visualize JSON [Dropdown] [More]

```
1 {
2   "id": 3,
3   "name": "Product",
4   "price": -120
5 }
```

P07 - Create a product with price as string

The screenshot shows a REST client interface with a POST request to `{{BASE_URL}}/api/Product`. The request body is a JSON object: `{ "id": 4, "name": "Product", "price": "free" }`. The response status is `400 Bad Request` with a response time of 24 ms and a body size of 540 B. The response body is displayed in a pretty-printed JSON format, showing validation errors: `{ "type": "https://tools.ietf.org/html/rfc9110#section-15.5.1", "title": "One or more validation errors occurred.", "status": 400, "errors": { "product": ["The product field is required."], "$.price": ["The JSON value could not be converted to System.Decimal. Path: $.price | LineNumber: 3 | BytePositionInLine: 17."] }, "traceId": "00-cc8e52a503c532cefd09ee8c2ad66aa4-5286d3751a941efe-00" }`.

```
POST {{BASE_URL}}/api/Product
```

Overview Params Authorization Headers (9) Body Scripts Settings Cookies

☐ none ☐ form-data ☐ x-www-form-urlencoded ☒ raw ☐ binary ☐ GraphQL JSON

```
1 {
2   "id": 4,
3   "name": "Product",
4   "price": "free"
5 }
```

Body Cookies Headers (4) Test Results (0/2) 400 Bad Request • 24 ms • 540 B • [Globe] [Copy] [More]

Pretty Raw Preview Visualize JSON [Dropdown] [More]

```
1 {
2   "type": "https://tools.ietf.org/html/rfc9110#section-15.5.1",
3   "title": "One or more validation errors occurred.",
4   "status": 400,
5   "errors": {
6     "product": [
7       "The product field is required."
8     ],
9     "$.price": [
10      "The JSON value could not be converted to System.Decimal. Path: $.price | LineNumber: 3 |
11      BytePositionInLine: 17."
12    ]
13   },
14   "traceId": "00-cc8e52a503c532cefd09ee8c2ad66aa4-5286d3751a941efe-00"
15 }
```


P08 - Create a product with empty fields

The screenshot shows a REST client interface with a POST request to `{{BASE_URL}}/api/Product`. The request body is a JSON object with empty string values for `id`, `name`, and `price`.

```
1 {
2   "id": "",
3   "name": "",
4   "price": ""
5 }
```

The response is a **400 Bad Request** with a status of 400, a response time of 23 ms, and a body size of 532 B. The response body is a JSON object containing error details:

```
1 {
2   "type": "https://tools.ietf.org/html/rfc9110#section-15.5.1",
3   "title": "One or more validation errors occurred.",
4   "status": 400,
5   "errors": {
6     "product": [
7       "The product field is required."
8     ],
9     "$.id": [
10      "The JSON value could not be converted to System.Int32. Path: $.id | LineNumber: 1 | BytePositionInLine: 10."
11    ]
12  },
13   "traceId": "00-0a97f0d4f924dbd8ae15fd86a5f68c61-a5aa67f59131c2ea-00"
14 }
```

P09 - Create a product with duplicate id and name

The screenshot shows a REST client interface with a POST request to `{{BASE_URL}}/api/Product`. The request body is a JSON object with `id` and `name` set to 1 and "Product" respectively, and `price` set to 10.

```
1 {
2   "id": 1,
3   "name": "Product",
4   "price": 10
5 }
```

The response is a **201 Created** with a status of 201, a response time of 17 ms, and a body size of 236 B. The response body is a JSON object with the same data as the request:

```
1 {
2   "id": 1,
3   "name": "Product",
4   "price": 10
5 }
```


P010 - Get list of products

GET `{{BASE_URL}}/api/Product/` **Send**

Overview Params Authorization Headers (7) **Body** Scripts Settings Cookies

☒ none ☐ form-data ☐ x-www-form-urlencoded ☐ raw ☐ binary ☐ GraphQL

This request does not have a body

Body Cookies Headers (4) Test Results (2/2) **200 OK** • 29 ms • 676 B •

Pretty Raw Preview Visualize JSON

```
13 {
14   "id": -2,
15   "name": "Object2",
16   "price": 5
17 },
18 {
19   "id": -2,
```

P011 - Get an existing product by id

GET `{{BASE_URL}}/api/Product/1` **Send**

Overview Params Authorization Headers (7) **Body** Scripts Settings Cookies

☒ none ☐ form-data ☐ x-www-form-urlencoded ☐ raw ☐ binary ☐ GraphQL

This request does not have a body

Body Cookies Headers (4) Test Results (1/2) **200 OK** • 26 ms • 185 B •

Pretty Raw Preview Visualize JSON

```
1 {
2   "id": 1,
3   "name": "Product",
4   "price": 120
5 }
```

P012 - Get a product that does not exist

The screenshot shows a REST client interface with a GET request to `{{BASE_URL}}/api/Product/999`. The response is a 404 Not Found status with a response time of 40 ms and a body size of 325 B. The response body is displayed in JSON format:

```
1 {
2   "type": "https://tools.ietf.org/html/rfc9110#section-15.5.5",
3   "title": "Not Found",
4   "status": 404,
5   "traceId": "00-e8abf8f5222a4bd6bc02abd947792897-330557e4ed9ff608-00"
6 }
```

P013 - Get a product with negative id

The screenshot shows a REST client interface with a GET request to `{{BASE_URL}}/api/Product/-1`. The response is a 200 OK status with a response time of 38 ms and a body size of 186 B. The response body is displayed in JSON format:

```
1 {
2   "id": -1,
3   "name": "Product",
4   "price": 120
5 }
```

P014 - Get a product with non-numeric id

The screenshot shows a REST client interface with a GET request to `{{BASE_URL}}/api/Product/abc`. The response is a 400 Bad Request with the following JSON body:

```
1 {
2   "type": "https://tools.ietf.org/html/rfc9110#section-15.5.1",
3   "title": "One or more validation errors occurred.",
4   "status": 400,
5   "errors": {
6     "id": [
7       "The value 'abc' is not valid."
8     ]
9   },
10  "traceId": "00-3c8adc4602e70168105bad7bd2e8a223-b417bfb44eca32c8-00"
11 }
```

P015 - Get a product with id in 0

The screenshot shows a REST client interface with a GET request to `{{BASE_URL}}/api/Product/0`. The response is a 200 OK with the following JSON body:

```
1 {
2   "id": 0,
3   "name": "Product",
4   "price": 120
5 }
```

P016 - Update an existing product

The screenshot shows a REST client interface with a PUT request to `{{BASE_URL}}/api/Product/1`. The request body is a JSON object: `{ "id": 2, "name": "Object", "price": 150 }`. The response is `204 No Content` with a status of 22 ms and 81 B. The response body is empty.

```
PUT {{BASE_URL}}/api/Product/1
```

Overview Params Authorization Headers (9) Body Scripts Settings Cookies

☐ none ☐ form-data ☐ x-www-form-urlencoded ☒ raw ☐ binary ☐ GraphQL JSON Beautify

```
1 {
2   "id": 2,
3   "name": "Object",
4   "price": 150
5 }
```

Body Cookies Headers (2) Test Results 204 No Content • 22 ms • 81 B • [Globe] [E.g.] [More]

Pretty Raw Preview Visualize Text [Dropdown] [Copy]

```
1
```

P017 - Update a non-existing product

The screenshot shows a REST client interface with a PUT request to `{{BASE_URL}}/api/Product/999`. The request body is a JSON object: `{ "id": 2, "name": "Object", "price": 150 }`. The response is `404 Not Found` with a status of 73 ms and 325 B. The response body is a JSON object: `{ "type": "https://tools.ietf.org/html/rfc9110#section-15.5.5", "title": "Not Found", "status": 404, "traceId": "00-48f684ddc13ce3c631d7ca4b5c4b55b8-1e1ace2492b114c6-00" }`.

```
PUT {{BASE_URL}}/api/Product/999
```

Overview Params Authorization Headers (9) Body Scripts Settings Cookies

☐ none ☐ form-data ☐ x-www-form-urlencoded ☒ raw ☐ binary ☐ GraphQL JSON Beautify

```
1 {
2   "id": 2,
3   "name": "Object",
4   "price": 150
5 }
```

Body Cookies Headers (4) Test Results 404 Not Found • 73 ms • 325 B • [Globe] [E.g.] [More]

Pretty Raw Preview Visualize JSON [Dropdown] [Copy]

```
1 {
2   "type": "https://tools.ietf.org/html/rfc9110#section-15.5.5",
3   "title": "Not Found",
4   "status": 404,
5   "traceId": "00-48f684ddc13ce3c631d7ca4b5c4b55b8-1e1ace2492b114c6-00"
6 }
```

P018 - Update a product with duplicate id

The screenshot shows a REST client interface with a PUT request to the endpoint `{{BASE_URL}}/api/Product/2`. The request body is a JSON object: `{ "id": 1, "name": "Duplicate", "price": 150 }`. The response is `204 No Content` with a status of 20 ms and 81 B. The response body is empty.

PUT `{{BASE_URL}}/api/Product/2` Send

Overview Params Authorization Headers (9) Body Scripts Settings Cookies

☐ none ☐ form-data ☐ x-www-form-urlencoded ☒ raw ☐ binary ☐ GraphQL JSON Beautify

```
1 {
2   "id": 1,
3   "name": "Duplicate",
4   "price": 150
5 }
```

Body Cookies Headers (2) Test Results 204 No Content • 20 ms • 81 B • [Globe] [e.g.] [More]

Pretty Raw Preview Visualize Text [v] [Copy]

1

P019 - Update a product with negative id

The screenshot shows a REST client interface with a PUT request to the endpoint `{{BASE_URL}}/api/Product/1`. The request body is a JSON object: `{ "id": -1, "name": "Duplicate", "price": 150 }`. The response is `204 No Content` with a status of 27 ms and 81 B. The response body is empty.

PUT `{{BASE_URL}}/api/Product/1` Send

Overview Params Authorization Headers (9) Body Scripts Settings Cookies

☐ none ☐ form-data ☐ x-www-form-urlencoded ☒ raw ☐ binary ☐ GraphQL JSON Beautify

```
1 {
2   "id": -1,
3   "name": "Duplicate",
4   "price": 150
5 }
```

Body Cookies Headers (2) Test Results 204 No Content • 27 ms • 81 B • [Globe] [e.g.] [More]

Pretty Raw Preview Visualize Text [v] [Copy]

1

P020 - Update a product eliminating required fields

The screenshot shows a REST client interface with a PUT request to `{{BASE_URL}}/api/Product/1`. The request body is a JSON object: `{ "price": 150 }`. The response is `204 No Content` with a status of 16 ms and 81 B. The interface includes tabs for Overview, Params, Authorization, Headers (9), Body, Scripts, and Settings. The Body tab is active, showing the JSON payload. The response status is highlighted in green.

```
PUT {{BASE_URL}}/api/Product/1
```

```
{  
  "price": 150  
}
```

204 No Content • 16 ms • 81 B

P021 - Update with mismatched URL and payload ID

The screenshot shows a REST client interface with a PUT request to `{{BASE_URL}}/api/Product/2`. The request body is a JSON object: `{ "id": 3, "name": "Duplicate", "price": 150 }`. The response is `204 No Content` with a status of 22 ms and 81 B. The interface includes tabs for Overview, Params, Authorization, Headers (9), Body, Scripts, and Settings. The Body tab is active, showing the JSON payload. The response status is highlighted in green.

```
PUT {{BASE_URL}}/api/Product/2
```

```
{  
  "id": 3,  
  "name": "Duplicate",  
  "price": 150  
}
```

204 No Content • 22 ms • 81 B

P022 - Delete an existing product

The screenshot shows a REST client interface with a DELETE request to `http://localhost:5044/api/Product/1`. The response is `204 No Content` with a status of 15 ms and 81 B. The interface includes tabs for Overview, Params, Authorization, Headers (7), Body, Scripts, and Settings. The Body tab is active, showing the message "This request does not have a body". The response status is highlighted in green.

```
DELETE http://localhost:5044/api/Product/1
```

This request does not have a body

204 No Content • 15 ms • 81 B

P023 - Delete a non-existing product

DELETE ▼ http://localhost:5044/api/Product/999 Send ▼

Overview Params Authorization Headers (7) Body Scripts Settings Cookies

☒ none ☐ form-data ☐ x-www-form-urlencoded ☐ raw ☐ binary ☐ GraphQL

This request does not have a body

Body Cookies Headers (4) Test Results **404 Not Found** • 4 ms • 325 B •

Pretty Raw Preview Visualize JSON ▼

```
1 {
2   "type": "https://tools.ietf.org/html/rfc9110#section-15.5.5",
3   "title": "Not Found",
4   "status": 404,
5   "traceId": "00-68bdacf7d4f4e2f84d3f0b845f92e48e-355bcf05e7ab4447-00"
6 }
```

P024 - Delete a product with negative id

DELETE ▼ http://localhost:5044/api/Product/-2 Send ▼

Overview Params Authorization Headers (7) Body Scripts Settings Cookies

☒ none ☐ form-data ☐ x-www-form-urlencoded ☐ raw ☐ binary ☐ GraphQL

This request does not have a body

Body Cookies Headers (2) Test Results **204 No Content** • 21 ms • 81 B •

Pretty Raw Preview Visualize Text ▼

```
1
```

P025 - Attempt to delete a product with id = 0

DELETE ▼ http://localhost:5044/api/Product/0 Send ▼

Overview Params Authorization Headers (7) Body Scripts Settings Cookies

☒ none ☐ form-data ☐ x-www-form-urlencoded ☐ raw ☐ binary ☐ GraphQL

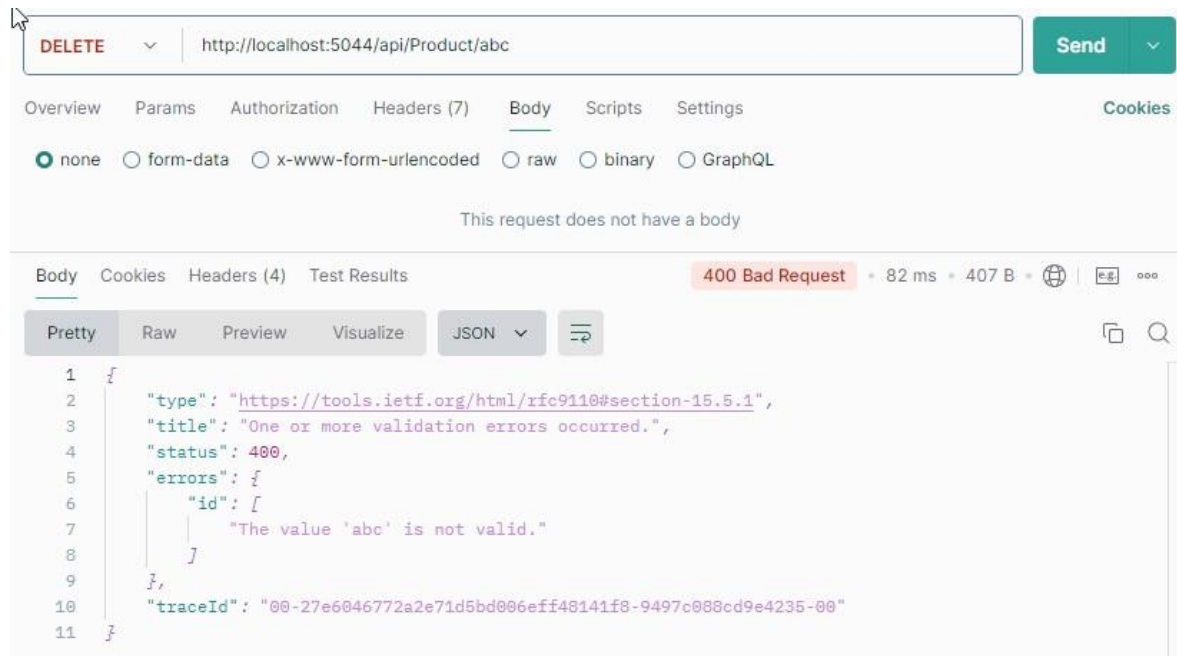
This request does not have a body

Body Cookies Headers (2) Test Results **204 No Content** • 25 ms • 81 B •

Pretty Raw Preview Visualize Text ▼

```
1
```


P026 - Delete a product with non-numeric id



DELETE ▼ http://localhost:5044/api/Product/abc Send ▼

Overview Params Authorization Headers (7) Body Scripts Settings Cookies

☒ none ☐ form-data ☐ x-www-form-urlencoded ☐ raw ☐ binary ☐ GraphQL

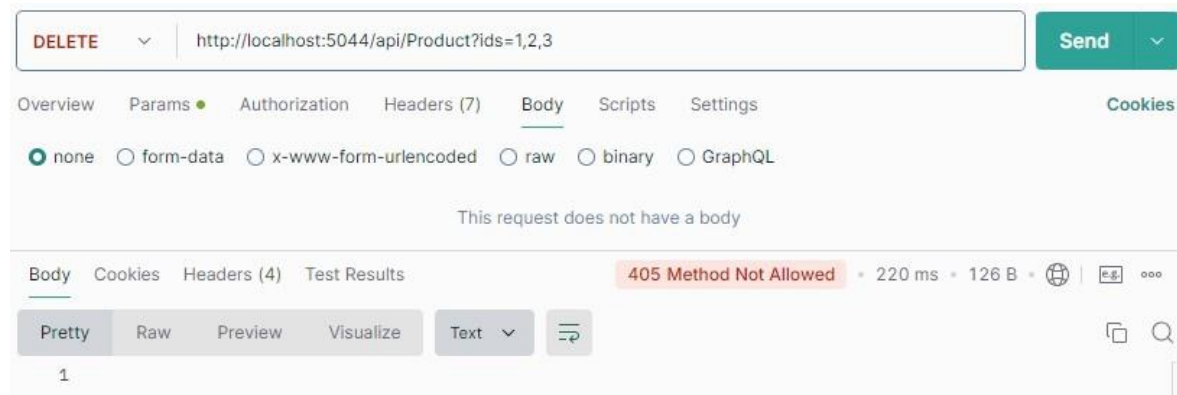
This request does not have a body

Body Cookies Headers (4) Test Results 400 Bad Request • 82 ms • 407 B •

Pretty Raw Preview Visualize JSON ▼

```
1 {
2   "type": "https://tools.ietf.org/html/rfc9110#section-15.5.1",
3   "title": "One or more validation errors occurred.",
4   "status": 400,
5   "errors": {
6     "id": [
7       "The value 'abc' is not valid."
8     ]
9   },
10  "traceId": "00-27e6046772a2e71d5bd006eff48141f8-9497c088cd9e4235-00"
11 }
```

P027 - Delete multiple products at once



DELETE ▼ http://localhost:5044/api/Product?ids=1,2,3 Send ▼

Overview Params ● Authorization Headers (7) Body Scripts Settings Cookies

☒ none ☐ form-data ☐ x-www-form-urlencoded ☐ raw ☐ binary ☐ GraphQL

This request does not have a body

Body Cookies Headers (4) Test Results 405 Method Not Allowed • 220 ms • 126 B •

Pretty Raw Preview Visualize Text ▼

```
1
```

O01 - Create an order with valid data

The screenshot shows a REST client interface with the following details:

- Method:** POST
- URL:** `{{BASE_URL}}/api/Order`
- Body Type:** raw (selected)
- Body Content:**

```
1 {  
2   "id": 1,  
3   "productName": "Product",  
4   "quantity": 3,  
5   "status": "Pending"  
6 }
```
- Response:** 201 Created, 99 ms, 262 B
- Response Body:**

```
1 {  
2   "id": 1,  
3   "productName": "Product",  
4   "quantity": 3,  
5   "status": "Pending"  
6 }
```

O02 - Create order with quantity = 0

The screenshot shows a REST client interface with the following details:

- Method:** POST
- URL:** `{{BASE_URL}}/api/Order`
- Body Type:** raw (selected)
- Body Content:**

```
1 {  
2   "id": 2,  
3   "productName": "Product",  
4   "quantity": 0,  
5   "status": "Pending"  
6 }
```
- Response:** 201 Created, 8 ms, 262 B
- Response Body:**

```
1 {  
2   "id": 2,  
3   "productName": "Product",  
4   "quantity": 0,  
5   "status": "Pending"  
6 }
```

O03 - Create order with empty status

The screenshot shows a REST client interface with a POST request to `{{BASE_URL}}/api/Order`. The request body is a JSON object: `{ "id": 3, "productName": "Product3", "quantity": 5, "status": "" }`. The response is a 201 Created status with a 7 ms response time and 256 B body size. The response body is displayed in the 'Pretty' view, showing the same JSON object.

```
POST {{BASE_URL}}/api/Order
```

Overview Params Authorization Headers (9) Body Scripts Settings Cookies

☐ none ☐ form-data ☐ x-www-form-urlencoded ☒ raw ☐ binary ☐ GraphQL JSON

```
1 {
2   "id": 3,
3   "productName": "Product3",
4   "quantity": 5,
5   "status": ""
6 }
```

Body Cookies Headers (5) Test Results 201 Created • 7 ms • 256 B • [Globe] [Copy] [More]

Pretty Raw Preview Visualize JSON [Dropdown] [Copy] [Search]

```
1 {
2   "id": 3,
3   "productName": "Product3",
4   "quantity": 5,
5   "status": ""
6 }
```

O04 - Create order with empty product name

The screenshot shows a REST client interface with a POST request to `{{BASE_URL}}/api/Order`. The request body is a JSON object: `{ "id": 4, "productName": "", "quantity": 2, "status": "Pending" }`. The response is a 201 Created status with a 7 ms response time and 255 B body size. The response body is displayed in the 'Pretty' view, showing the same JSON object.

```
POST {{BASE_URL}}/api/Order
```

Overview Params Authorization Headers (9) Body Scripts Settings Cookies

☐ none ☐ form-data ☐ x-www-form-urlencoded ☒ raw ☐ binary ☐ GraphQL JSON

```
1 {
2   "id": 4,
3   "productName": "",
4   "quantity": 2,
5   "status": "Pending"
6 }
```

Body Cookies Headers (5) Test Results 201 Created • 7 ms • 255 B • [Globe] [Copy] [More]

Pretty Raw Preview Visualize JSON [Dropdown] [Copy] [Search]

```
1 {
2   "id": 4,
3   "productName": "",
4   "quantity": 2,
5   "status": "Pending"
6 }
```

O05 - Create an order with duplicate ID

The screenshot shows a REST client interface with a POST request to `{{BASE_URL}}/api/Order`. The request body is a JSON object with the following fields:

```
1 {
2   "id": 1,
3   "productName": "Product5",
4   "quantity": 2,
5   "status": "Pending"
6 }
```

The response status is **201 Created** with a response time of 7 ms and a body size of 263 B. The response body is displayed in the 'Body' tab, showing the same JSON object as the request.

O06 - Get the list of all orders.

The screenshot shows a REST client interface with a GET request to `{{BASE_URL}}/api/Order/`. The response status is **200 OK** with a response time of 7 ms and a body size of 462 B. The response body is displayed in the 'Body' tab, showing a JSON array of three order objects:

```
1 [
2   {
3     "id": 1,
4     "productName": "Product",
5     "quantity": 3,
6     "status": "Pending"
7   },
8   {
9     "id": 2,
10    "productName": "Product",
11    "quantity": 0,
12    "status": "Pending"
13  },
14  {
15    "id": 3,
```

O07 - Get list when no orders exist.

The screenshot shows a REST client interface. The request is a GET to `{{BASE_URL}}/api/Order/`. The response is a 200 OK with a status of 200 OK, a time of 7 ms, and a size of 150 B. The response body is empty, indicated by the text "This request does not have a body".

Overview Params Authorization Headers (7) Body Scripts Settings Cookies

☒ none ☐ form-data ☐ x-www-form-urlencoded ☐ raw ☐ binary ☐ GraphQL

This request does not have a body

Body Cookies Headers (4) Test Results 200 OK • 7 ms • 150 B • [Globe Icon] [Copy Icon] [More Icon]

Pretty Raw Preview Visualize JSON [Dropdown] [Copy Icon]

```
1 [ ]
```

O08 - Get an order by an existing ID

The screenshot shows a REST client interface. The request is a GET to `{{BASE_URL}}/api/Order/1`. The response is a 200 OK with a status of 200 OK, a time of 28 ms, and a size of 212 B. The response body is a JSON object representing an order.

Overview Params Authorization Headers (7) Body Scripts Settings Cookies

☒ none ☐ form-data ☐ x-www-form-urlencoded ☐ raw ☐ binary ☐ GraphQL

This request does not have a body

Body Cookies Headers (4) Test Results 200 OK • 28 ms • 212 B • [Globe Icon] [Copy Icon] [More Icon]

Pretty Raw Preview Visualize JSON [Dropdown] [Copy Icon]

```
1 {
2   "id": 1,
3   "productName": "Product",
4   "quantity": 3,
5   "status": "Pending"
6 }
```

O09 - Get an order with non-existing ID

The screenshot shows a REST client interface. The request is a GET to `{{BASE_URL}}/api/Order/999`. The response is a 404 Not Found with a status of 404 Not Found, a time of 42 ms, and a size of 325 B. The response body is a JSON object representing an error.

Overview Params Authorization Headers (7) Body Scripts Settings Cookies

☒ none ☐ form-data ☐ x-www-form-urlencoded ☐ raw ☐ binary ☐ GraphQL

This request does not have a body

Body Cookies Headers (4) Test Results 404 Not Found • 42 ms • 325 B • [Globe Icon] [Copy Icon] [More Icon]

Pretty Raw Preview Visualize JSON [Dropdown] [Copy Icon]

```
1 {
2   "type": "https://tools.ietf.org/html/rfc9110#section-15.6.5",
3   "title": "Not Found",
4   "status": 404,
5   "traceId": "00-a93bf1dd96d0fa3efe5bb580f31f0026-8592b6ca69935094-00"
6 }
```

O010 - Get an order with negative ID

The screenshot shows a REST client interface with the following details:

- Method:** GET
- URL:** `{{BASE_URL}}/api/Order/-1`
- Buttons:** Overview, Params, Authorization, Headers (7), **Body**, Scripts, Settings, Cookies
- Body Type:** none (selected), form-data, x-www-form-urlencoded, raw, binary, GraphQL
- Status:** 200 OK, 3 ms, 214 B
- Body Content (JSON):**

```
1 {
2   "id": -1,
3   "productName": "Product5",
4   "quantity": 2,
5   "status": "Pending"
6 }
```

O011 Get an order with non-numeric ID

The screenshot shows a REST client interface with the following details:

- Method:** GET
- URL:** `{{BASE_URL}}/api/Order/abc`
- Buttons:** Overview, Params, Authorization, Headers (7), **Body**, Scripts, Settings, Cookies
- Body Type:** none (selected), form-data, x-www-form-urlencoded, raw, binary, GraphQL
- Status:** 400 Bad Request, 68 ms, 407 B
- Body Content (JSON):**

```
1 {
2   "type": "https://tools.ietf.org/html/rfc9110#section-15.5.1",
3   "title": "One or more validation errors occurred.",
4   "status": 400,
5   "errors": {
6     "id": [
7       "The value 'abc' is not valid."
8     ]
9   },
10  "traceId": "00-179a1107474ab0da6313da9bf71d2610-31f93cdf71bc67c2-00"
11 }
```

O012 Get an order with id = 0

GET `{{BASE_URL}}/api/Order/0` **Send**

Overview Params Authorization Headers (7) **Body** Scripts Settings Cookies

☒ none ☐ form-data ☐ x-www-form-urlencoded ☐ raw ☐ binary ☐ GraphQL

This request does not have a body

Body Cookies Headers (4) Test Results **200 OK** • 5 ms • 213 B •

Pretty Raw Preview Visualize JSON

```
1 {  
2   "id": 0,  
3   "productName": "Product7",  
4   "quantity": 2,  
5   "status": "Pending"  
6 }
```

O013 Update an existing order with valid data.

PUT `{{BASE_URL}}/api/Order/1` **Send**

Overview Params Authorization Headers (9) **Body** Scripts Settings Cookies

☐ none ☐ form-data ☐ x-www-form-urlencoded ☒ raw ☐ binary ☐ GraphQL JSON Beautify

```
1 {  
2   "id": 1,  
3   "productName": "aasa",  
4   "quantity": 2,  
5   "status": "Shipped"  
6 }
```

Body Cookies Headers (2) Test Results **204 No Content** • 56 ms • 81 B •

Pretty Raw Preview Visualize Text

1

O014 Update order with quantity = 0.

The screenshot shows a REST client interface with a PUT request to `{{BASE_URL}}/api/Order/1`. The request body is a JSON object: `{ "id": 1, "productName": "Product", "quantity": 0, "status": "Shipped" }`. The response is `204 No Content` with a status of 18 ms and 81 B. The response body is empty.

PUT `{{BASE_URL}}/api/Order/1` Send

Overview Params Authorization Headers (9) Body Scripts Settings Cookies

☐ none ☐ form-data ☐ x-www-form-urlencoded ☒ raw ☐ binary ☐ GraphQL JSON Beautify

```
1 {
2   "id": 1,
3   "productName": "Product",
4   "quantity": 0,
5   "status": "Shipped"
6 }
```

Body Cookies Headers (2) Test Results 204 No Content • 18 ms • 81 B • [Globe] [Copy] [More]

Pretty Raw Preview Visualize Text [Dropdown] [More]

1

O015 Update order with empty status

The screenshot shows a REST client interface with a PUT request to `{{BASE_URL}}/api/Order/1`. The request body is a JSON object: `{ "id": 1, "productName": "Product", "quantity": 1, "status": "" }`. The response is `204 No Content` with a status of 21 ms and 81 B. The response body is empty.

PUT `{{BASE_URL}}/api/Order/1` Send

Overview Params Authorization Headers (9) Body Scripts Settings Cookies

☐ none ☐ form-data ☐ x-www-form-urlencoded ☒ raw ☐ binary ☐ GraphQL JSON Beautify

```
1 {
2   "id": 1,
3   "productName": "Product",
4   "quantity": 1,
5   "status": ""
6 }
```

Body Cookies Headers (2) Test Results 204 No Content • 21 ms • 81 B • [Globe] [Copy] [More]

Pretty Raw Preview Visualize Text [Dropdown] [More]

1

O016 Update non-existent order.

The screenshot shows a REST client interface with a PUT request to `{{BASE_URL}}/api/Order/999`. The request body is a JSON object: `{ "id": 1, "productName": "Product", "quantity": 1, "status": "" }`. The response is a `404 Not Found` status with a response time of 17 ms and a body size of 325 B. The response body is shown in JSON format: `{ "type": "https://tools.ietf.org/html/rfc9110#section-15.5.5", "title": "Not Found", "status": 404, "traceId": "00-afb8e6957b6cdf42704af0e610b28b45-edc30dbd88fd7e7d-00" }`.

```
PUT {{BASE_URL}}/api/Order/999
```

Overview Params Authorization Headers (9) Body Scripts Settings Cookies

☐ none ☐ form-data ☐ x-www-form-urlencoded ☒ raw ☐ binary ☐ GraphQL JSON

```
1 {
2   "id": 1,
3   "productName": "Product",
4   "quantity": 1,
5   "status": ""
6 }
```

Body Cookies Headers (4) Test Results 404 Not Found • 17 ms • 325 B • [Globe] [Copy] [More]

Pretty Raw Preview Visualize JSON [Copy]

```
1 {
2   "type": "https://tools.ietf.org/html/rfc9110#section-15.5.5",
3   "title": "Not Found",
4   "status": 404,
5   "traceId": "00-afb8e6957b6cdf42704af0e610b28b45-edc30dbd88fd7e7d-00"
6 }
```

O017 Delete an existing order

The screenshot shows a REST client interface with a DELETE request to `{{BASE_URL}}/api/Order/1`. The response is a `204 No Content` status with a response time of 17 ms and a body size of 81 B. The response body is empty.

```
DELETE {{BASE_URL}}/api/Order/1
```

Overview Params Authorization Headers (7) Body Scripts Settings Cookies

Delete Order

[View complete documentation →](#)

`http://localhost:5044/api/Order/0`

Add request description...

Body Cookies Headers (2) Test Results 204 No Content • 17 ms • 81 B • [Globe] [Copy] [More]

Pretty Raw Preview Visualize Text [Copy]

```
1
```

O018 Delete a non-existent order

The screenshot shows a REST client interface with a DELETE request to `{{BASE_URL}}/api/Order/999`. The response is a **404 Not Found** status with a response time of 4 ms and a body size of 325 B. The response body is displayed in JSON format:

```
1 {
2   "type": "https://tools.ietf.org/html/rfc9110#section-15.5.5",
3   "title": "Not Found",
4   "status": 404,
5   "traceId": "00-d571688c27eac24ad47b06b6cba696b0-d74ce188002d1e35-00"
6 }
```

O019 Delete an order with negative ID

The screenshot shows a REST client interface with a DELETE request to `{{BASE_URL}}/api/Order/-1`. The response is a **204 No Content** status with a response time of 4 ms and a body size of 81 B. The response body is empty.

O020 Delete an order with non-numeric ID

The screenshot shows a REST client interface with a DELETE request to `{{BASE_URL}}/api/Order/abc`. The response is a 400 Bad Request with the following JSON body:

```
1 {
2   "type": "https://tools.ietf.org/html/rfc9110#section-15.5.1",
3   "title": "One or more validation errors occurred.",
4   "status": 400,
5   "errors": {
6     "id": [
7       "The value 'abc' is not valid."
8     ]
9   },
10  "traceId": "00-c8bcf88f5a15353d1962b74ed586ea1d-5a9d7a67d10ac502-00"
11 }
```

O021 Delete an order with id = 0

The screenshot shows a REST client interface with a DELETE request to `{{BASE_URL}}/api/Order/0`. The response is a 204 No Content.

