Session Notes 1 – ECSE 429 (Todos) Laurent

Charter

- Identify capabilities and areas of potential instability of the “REST API todo list manager.”
- Identify documented and undocumented capabilities for Todos.
- Create and test scripts for each capability.
- Exercise each capability with data typical to intended use.

Build

- runTodoManagerRestAPI-1.5.5.jar

Area

- TODOS core flows + relationships (categories, projects) per provided Swagger.

Test Environment

- Windows 10
- Postman v11 (desktop client)
- API running locally via java -jar runTodoManagerRestAPI-1.5.5.jar

Date & Time

- October 1st, 2025 – Start time: 7:00 PM
- Session Duration: 45 minutes

Tester

- Laurent Chiricota – 261113415 – laurent.chiricota@mail.mcgill.ca

Images of the Postman results for todos will be attached at the end of this document.

Session Log

7:02 PM
Performed initial GET /todos. Response: 200 OK with list of todos. All doneStatus values set to "false". Noted that todos are not returned in sorted order by ID.

7:04 PM

Attempted PUT /todos and DELETE /todos directly. Both returned 405 Method Not Allowed, as expected per Swagger.

7:05 PM

POST /todos with valid JSON body including title, doneStatus (Boolean), and description. Todo created successfully with ID 4. Returned object matched expected schema.

7:07 PM

Performed GET /todos again. New todo with ID 4 present. The todos are displayed in random order based on their id.

7:09 PM

Attempted POST /todos with only a title field. Request succeeded, and API defaulted doneStatus to "false". This behavior is undocumented but useful.

7:12 PM

POST /todos with invalid value "maybe" for doneStatus. Response: 400 Bad Request. Observation: API correctly enforces boolean validation for doneStatus.

7:15 PM

GET /todos/2 (existing). Response: 200 OK, correct details returned.

7:16 PM

HEAD /todos/2 (existing). Response: 200 OK, headers only.
HEAD /todos/99 (non-existent). Response: 404 Not Found.

7:17 PM

GET /todos/99 (non-existent). Response: 404 Not Found with error message: "Could not find an instance with todos/99". Status code is correct, though phrasing is awkward**.**

7:20 PM

PUT /todos/2 – updated title and description. Worked as expected, fields updated. Matches Swagger “amend with fields to amend.”

7:23 PM

POST /todos/2 (partial amend: change description only). Response: 200 OK. Behavior same as PUT (Swagger says POST and PUT on /todos/:id both amend). Works.

7:25 PM

HEAD /todos – Response: 200 OK, empty body (headers only).

7:27 PM
DELETE /todos/4 – Response: 200 OK. Body empty {}. Swagger says “Deleted a specific todo” (no explicit schema). #ISSUE: confirmation message/body inconsistent with wording.

7:30 PM
DELETE /todos/99 (non-existent). Response: 404 Not Found. Proper error handling.

7:33 PM – Relationship (categories, smoke check)

- GET /todos/2/categories → 200 OK, empty list.
- HEAD /todos/2/categories → 200 OK, headers only.
- POST /todos/2/categories with body { "id": "1" } → 201 Created, relationship established.
- DELETE /todos/2/categories/1 → 404 Not Found if relationship does not exist (tested case). Aligns with Swagger “relationship not found.”
  Note: Only smoke-checking relationships here; full interoperability testing will be done in a separate session.

7:36 PM – Relationship (tasksof, smoke check)

- GET /todos/2/tasksof → 200 OK, empty list.
- HEAD /todos/2/tasksof → 200 OK, headers only.
- POST /todos/2/tasksof with body { "id": "1" } (after project creation) → 201 Created, relationship established.
- DELETE /todos/2/tasksof/1 → 404 Not Found if no relationship exists. Correct per Swagger.

7:40 PM
Explored HEAD /todos again. Response: 200 OK, consistent behavior.

7:44 PM
Final test: OPTIONS /todos. Response: 200 OK, headers indicate supported verbs and CORS.

Additional Observations

- IDs are not reused after deletions, creating permanent gaps between todos.
- Swagger does not explicitly describe behavior for HEAD/OPTIONS, but they behave correctly.

- Error messages functional but phrasing is awkward e.g. "Could not find any instances with todos/66"
- Data not persisted across server restarts (IDs reset).

Session Summary Findings

Key Issues

- Inconsistent ID management (gaps not reused).
- DELETE response body inconsistent with Swagger wording.
- Error messages awkwardly phrased (“Could not find an instance with todos/ID”).

Capabilities Identified

- CRUD operations (GET, POST, PUT, DELETE) work reliably.
- Boolean validation enforced on doneStatus.
- Default values applied when doneStatus missing.
- HEAD and OPTIONS supported on todos and individual todos.
- Relationship endpoints (categories and tasksof) reachable and working in smoke tests; full interop to be tested separately in another session.

New Test Ideas

- Test extremely long strings for title and description.
- Stress test with hundreds of todos created/deleted in sequence.
- Check interoperability in depth: multiple categories/projects, bidirectional updates.
- Test XML payloads for todos.

Areas of Potential Risk

- Clients may misinterpret ID gaps after deletions.
- Minimal or missing confirmation bodies on DELETE may confuse integrators.
- Error message phrasing unclear for developers relying on parsing.

Decisions Made

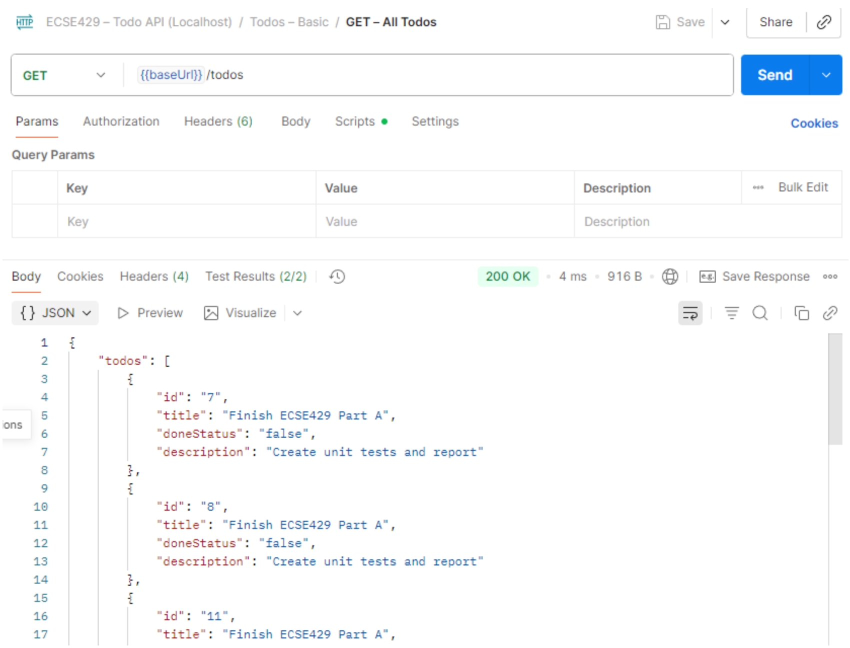
- Log issue for DELETE response body inconsistency.

- Note awkward error messages for potential bug report.
- Leave interoperability deep dive to dedicated session.

Files Referenced

- Screenshots from Postman showing valid and invalid responses.

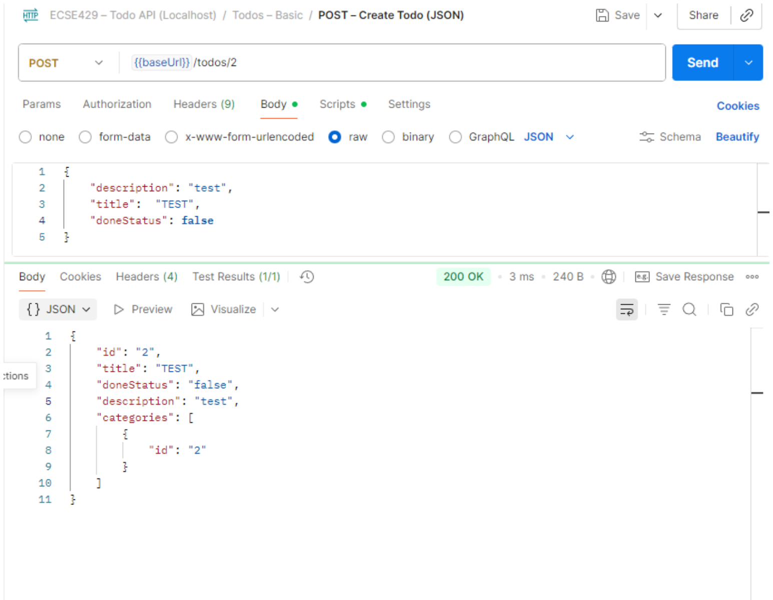
TODOS



GET {{baseUrl}} /todos

200 OK · 4 ms · 916 B

```
{
    "todos": [
        {
            "id": "7",
            "title": "Finish ECSE429 Part A",
            "doneStatus": "false",
            "description": "Create unit tests and report"
        },
        {
            "id": "8",
            "title": "Finish ECSE429 Part A",
            "doneStatus": "false",
            "description": "Create unit tests and report"
        },
        {
            "id": "11",
            "title": "Finish ECSE429 Part A",
```



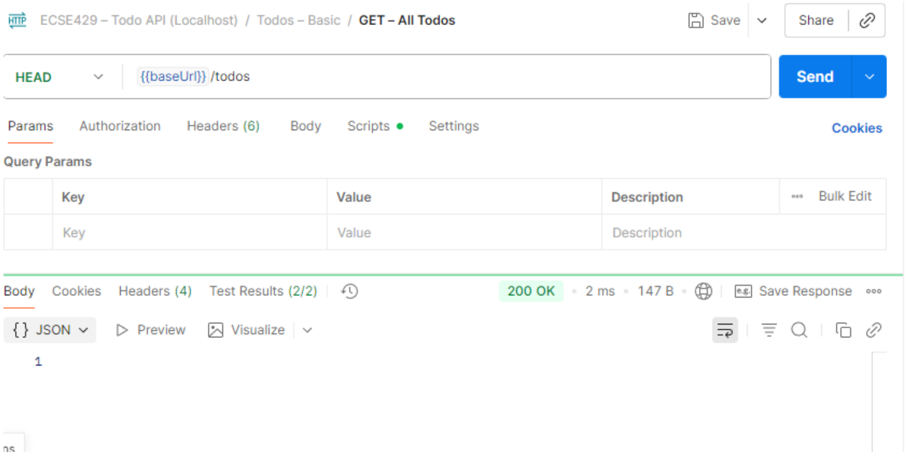
ECSE429 – Todo API (Localhost) / Todos – Basic / **POST – Create Todo (JSON)**

POST {{baseUrl}} /todos/2

```
{
    "description": "test",
    "title":  "TEST",
    "doneStatus": false
}
```

200 OK · 3 ms · 240 B

```
{
    "id": "2",
    "title": "TEST",
    "doneStatus": "false",
    "description": "test",
    "categories": [
        {
            "id": "2"
        }
    ]
}
```



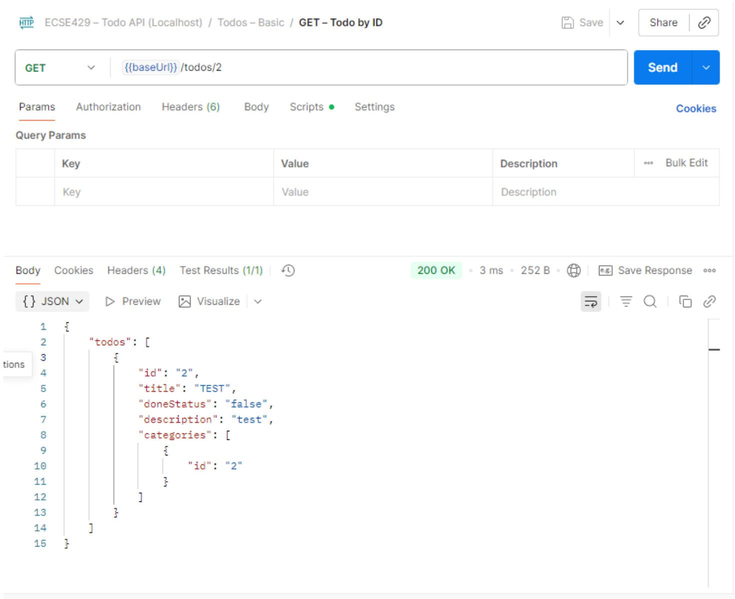
ECSE429 – Todo API (Localhost) / Todos – Basic / **GET – All Todos**

HEAD {{baseUrl}} /todos

Query Params

Key	Value	Description
Key	Value	Description

200 OK · 2 ms · 147 B

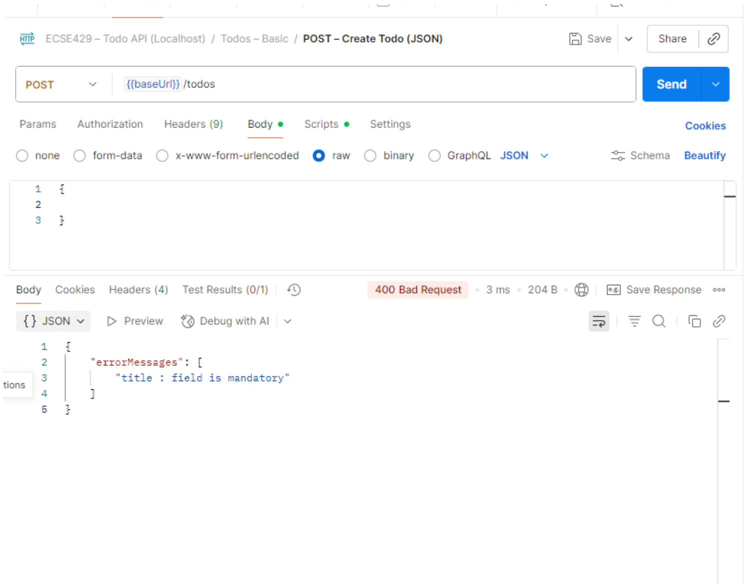


ECSE429 – Todo API (Localhost) / Todos – Basic / GET – Todo by ID

GET {{baseUrl}} /todos/2

200 OK · 3 ms · 252 B

```
{
    "todos": [
        {
            "id": "2",
            "title": "TEST",
            "doneStatus": "false",
            "description": "test",
            "categories": [
                {
                    "id": "2"
                }
            ]
        }
    ]
}
```



ECSE429 – Todo API (Localhost) / Todos – Basic / POST – Create Todo (JSON)

POST {{baseUrl}} /todos

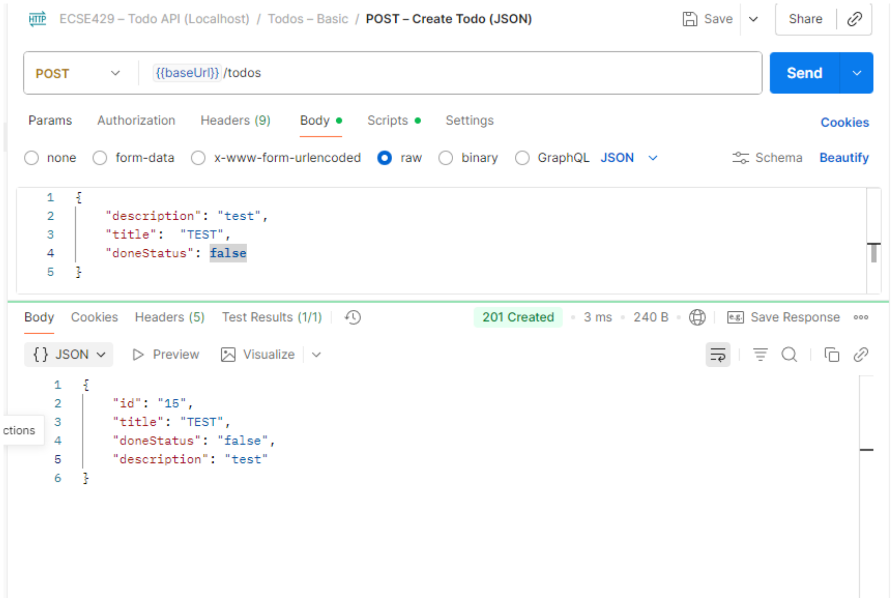
○ none ○ form-data ○ x-www-form-urlencoded ● raw ○ binary ○ GraphQL JSON

```
{

}
```

400 Bad Request · 3 ms · 204 B

```
{
    "errorMessages": [
        "title : field is mandatory"
    ]
}
```



ECSE429 – Todo API (Localhost) / Todos – Basic / **POST – Create Todo (JSON)**

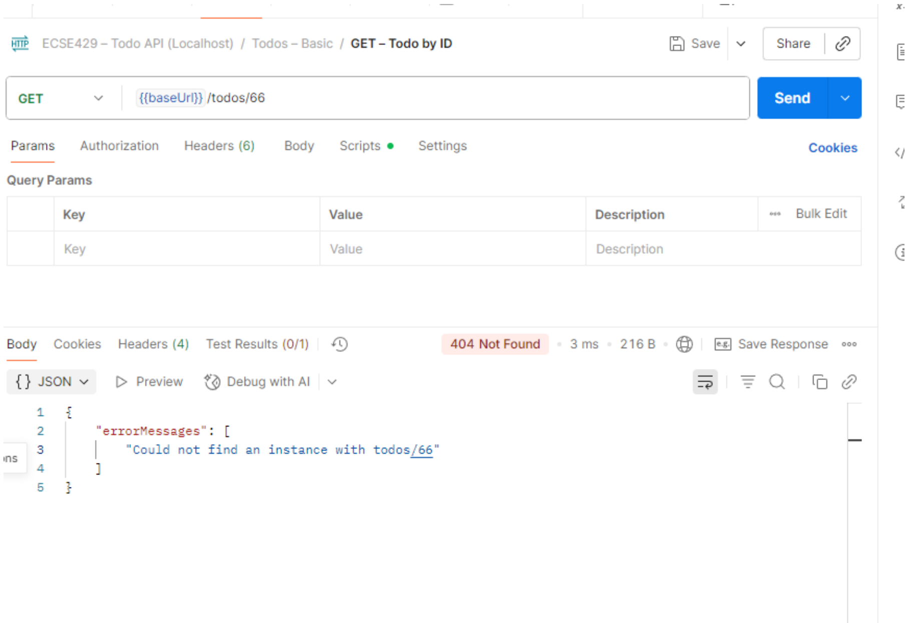
POST {{baseUrl}} /todos

○ none ○ form-data ○ x-www-form-urlencoded ● raw ○ binary ○ GraphQL JSON

```
{
    "description": "test",
    "title":  "TEST",
    "doneStatus": false
}
```

201 Created • 3 ms • 240 B

```
{
    "id": "15",
    "title": "TEST",
    "doneStatus": "false",
    "description": "test"
}
```



ECSE429 – Todo API (Localhost) / Todos – Basic / **GET – Todo by ID**

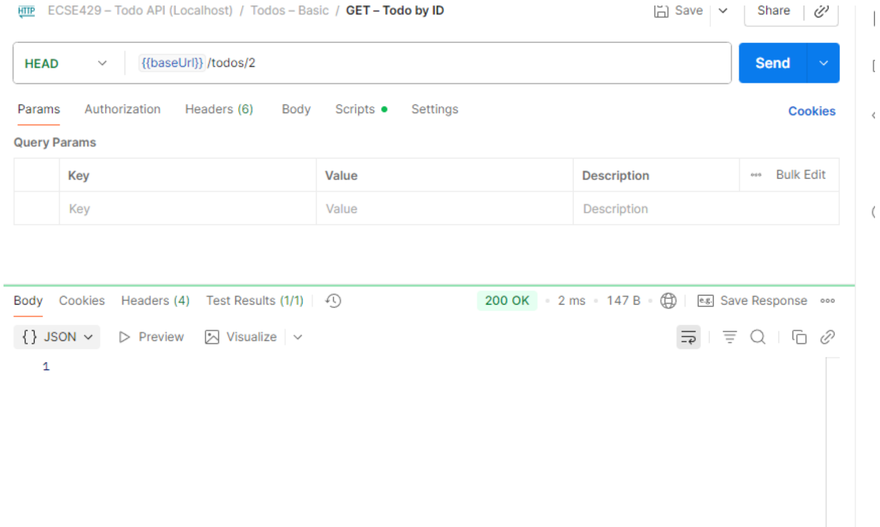
GET {{baseUrl}} /todos/66

Query Params

	Key	Value	Description
	Key	Value	Description

404 Not Found · 3 ms · 216 B

```
{
    "errorMessages": [
        "Could not find an instance with todos/66"
    ]
}
```



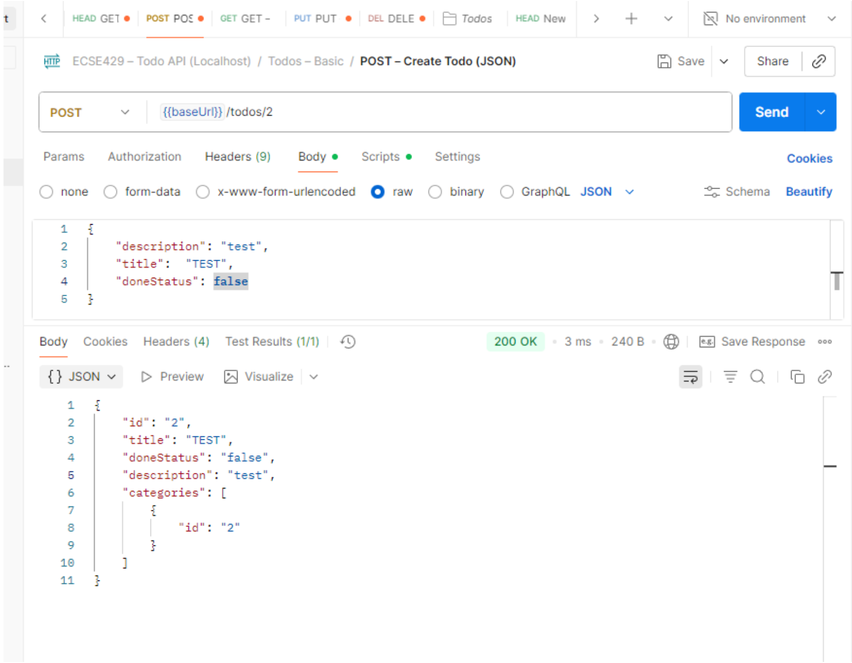
ECSE429 – Todo API (Localhost) / Todos – Basic / **GET – Todo by ID**

HEAD {{baseUrl}} /todos/2

Query Params

	Key	Value	Description
	Key	Value	Description

200 OK · 2 ms · 147 B



ECSE429 – Todo API (Localhost) / Todos – Basic / **POST – Create Todo (JSON)**

POST {{baseUrl}} /todos/2

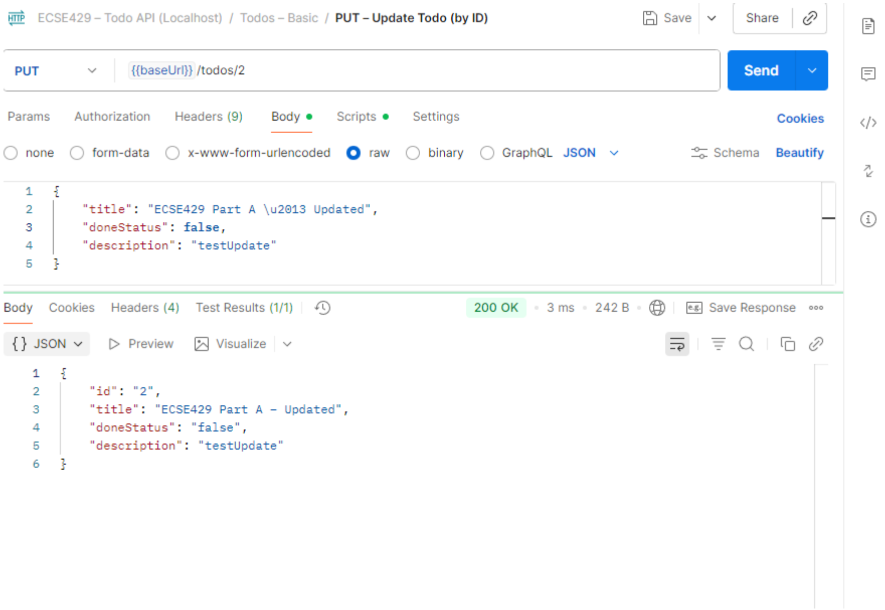
Body: ○ none ○ form-data ○ x-www-form-urlencoded ● raw ○ binary ○ GraphQL JSON

```
{
    "description": "test",
    "title":  "TEST",
    "doneStatus": false
}
```

Body Cookies Headers (4) Test Results (1/1) — 200 OK • 3 ms • 240 B

{ } JSON

```
{
    "id": "2",
    "title": "TEST",
    "doneStatus": "false",
    "description": "test",
    "categories": [
        {
            "id": "2"
        }
    ]
}
```



ECSE429 – Todo API (Localhost) / Todos – Basic / **PUT – Update Todo (by ID)**

PUT {{baseUrl}} /todos/2

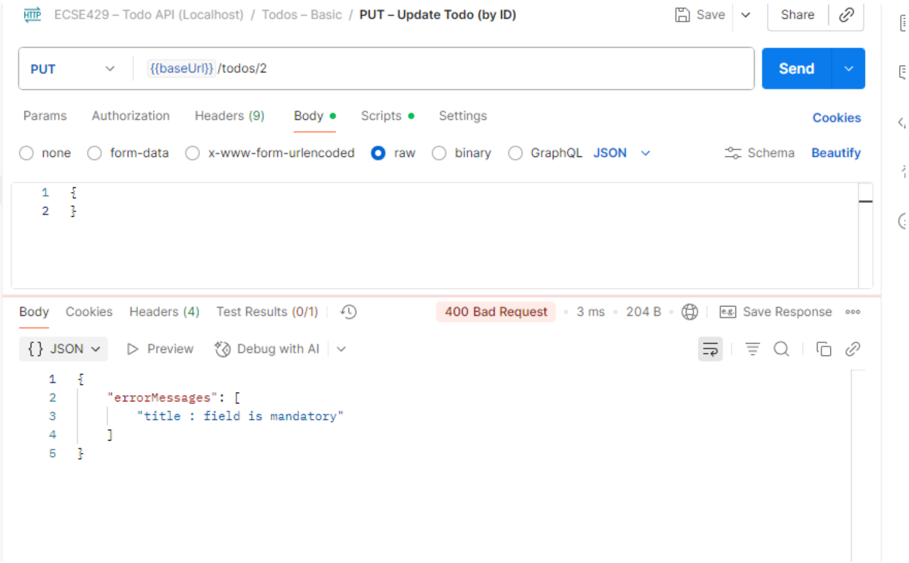
Body: ○ none ○ form-data ○ x-www-form-urlencoded ● raw ○ binary ○ GraphQL JSON

```
{
    "title": "ECSE429 Part A \u2013 Updated",
    "doneStatus": false,
    "description": "testUpdate"
}
```

Body Cookies Headers (4) Test Results (1/1) — 200 OK • 3 ms • 242 B

{ } JSON

```
{
    "id": "2",
    "title": "ECSE429 Part A – Updated",
    "doneStatus": "false",
    "description": "testUpdate"
}
```



ECSE429 – Todo API (Localhost) / Todos – Basic / **PUT – Update Todo (by ID)**

PUT {{baseUrl}} /todos/2

Params | Authorization | Headers (9) | Body ● | Scripts ● | Settings

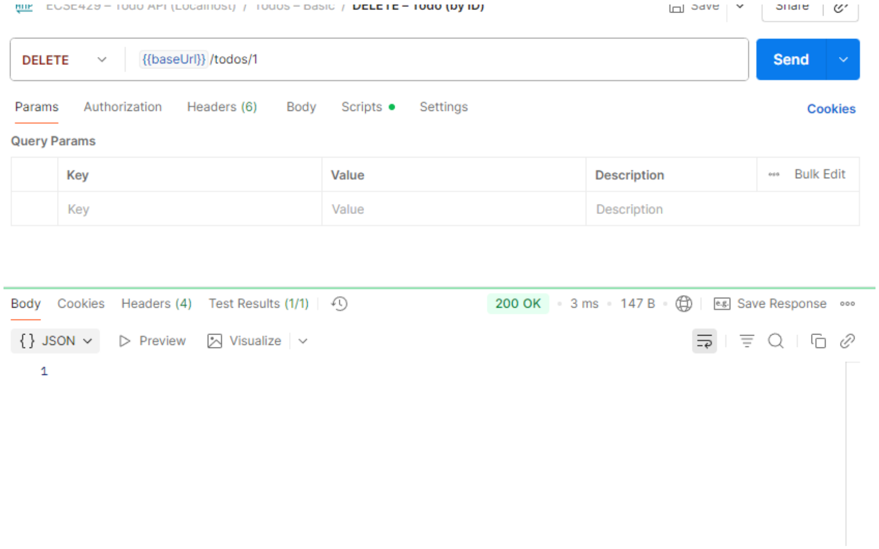
○ none ○ form-data ○ x-www-form-urlencoded ● raw ○ binary ○ GraphQL JSON

```
{
}
```

Body | Cookies | Headers (4) | Test Results (0/1) — 400 Bad Request • 3 ms • 204 B

{} JSON

```
{
    "errorMessages": [
        "title : field is mandatory"
    ]
}
```



ECSE429 – Todo API (Localhost) / Todos – Basic / **DELETE – Todo (by ID)**

DELETE {{baseUrl}} /todos/1

Params | Authorization | Headers (6) | Body | Scripts ● | Settings

Query Params

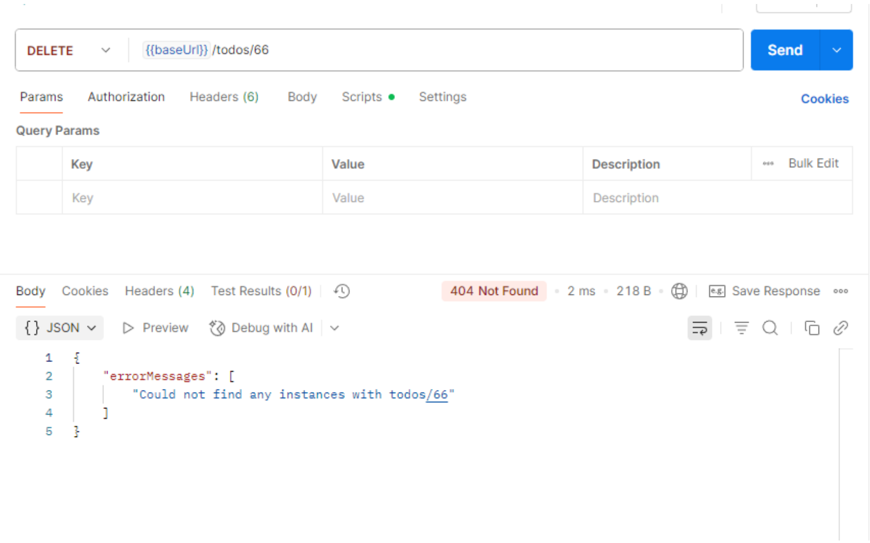
Key	Value	Description
Key	Value	Description

Body | Cookies | Headers (4) | Test Results (1/1) — 200 OK • 3 ms • 147 B

{} JSON

```

```



DELETE {{baseUrl}} /todos/66

Params | Authorization | Headers (6) | Body | Scripts ● | Settings

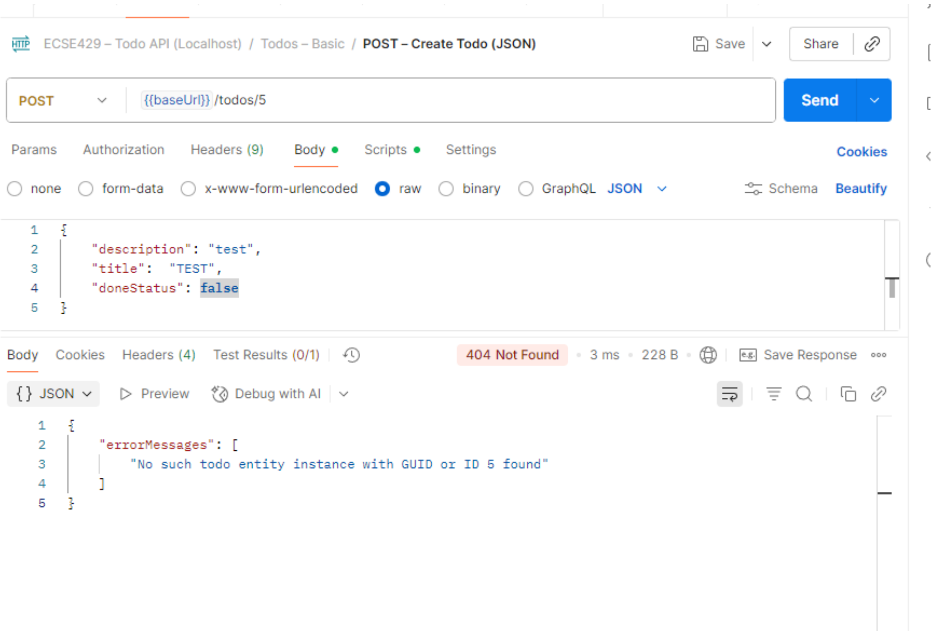
Query Params

Key	Value	Description
Key	Value	Description

Body | Cookies | Headers (4) | Test Results (0/1) — 404 Not Found • 2 ms • 218 B

{} JSON

```
{
    "errorMessages": [
        "Could not find any instances with todos/66"
    ]
}
```



ECSE429 – Todo API (Localhost) / Todos – Basic / POST – Create Todo (JSON)

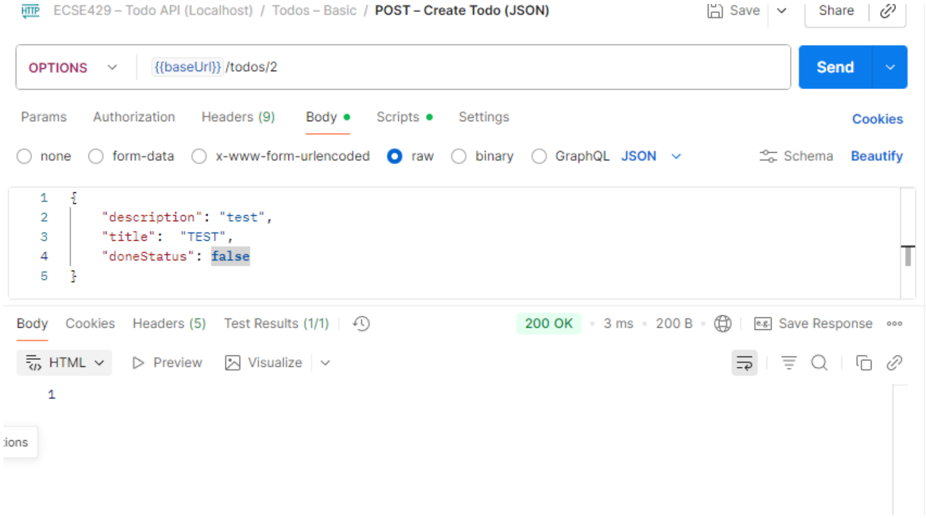
POST {{baseUrl}} /todos/5

Body: ● raw JSON

```
{
    "description": "test",
    "title":  "TEST",
    "doneStatus": false
}
```

Body Cookies Headers (4) Test Results (0/1) 404 Not Found • 3 ms • 228 B

```
{
    "errorMessages": [
        "No such todo entity instance with GUID or ID 5 found"
    ]
}
```



ECSE429 – Todo API (Localhost) / Todos – Basic / POST – Create Todo (JSON)

OPTIONS {{baseUrl}} /todos/2

Body: ● raw JSON

```
{
    "description": "test",
    "title":  "TEST",
    "doneStatus": false
}
```

Body Cookies Headers (5) Test Results (1/1) 200 OK • 3 ms • 200 B