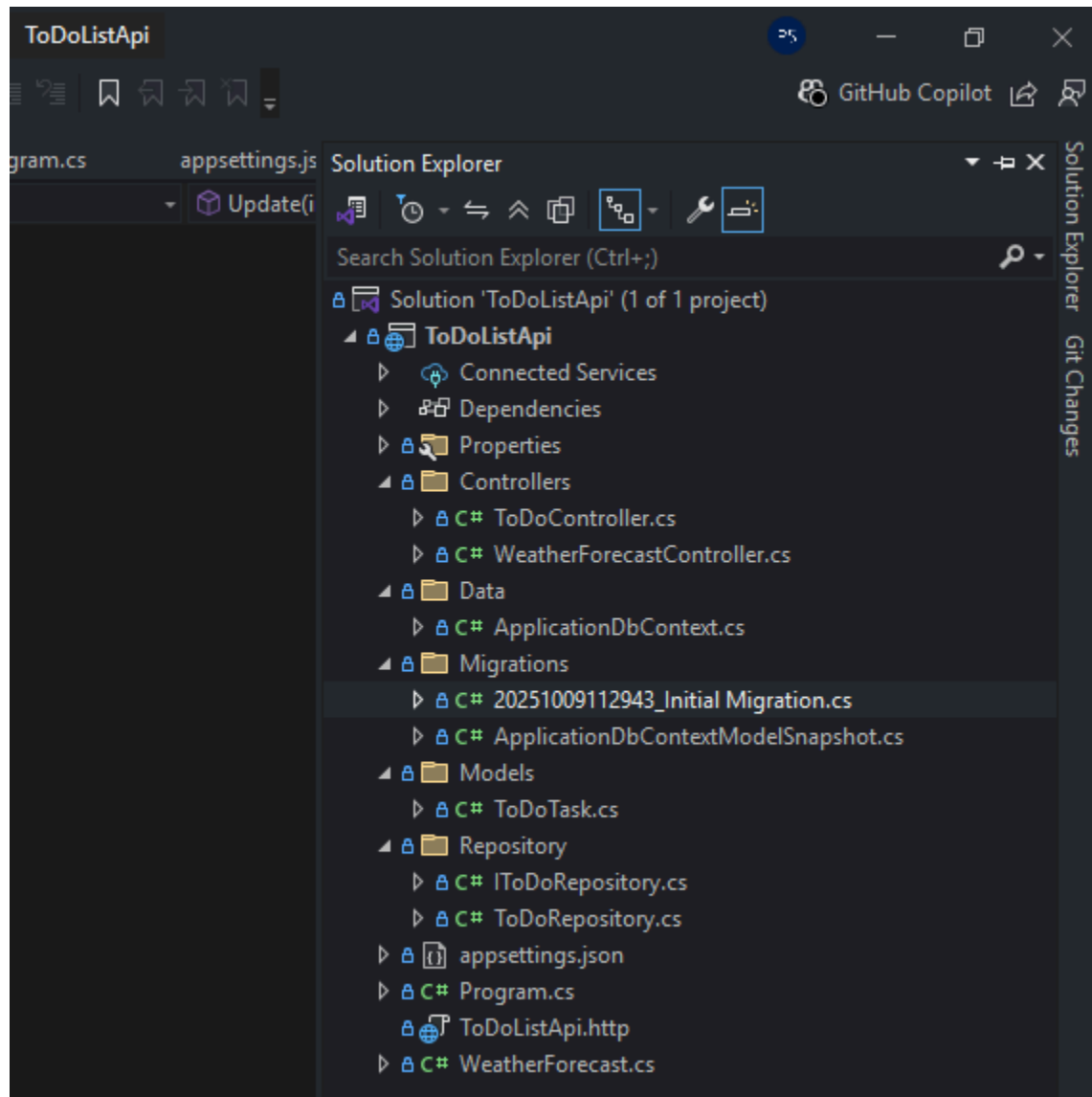


To Do List Assignment 3

Project Structure



Default Values in Table

18 | `select * from ToDoTasks`

	TaskId	Description	Priority	StartDate	EndDate	Status
1	1	Complete ASP.NET Core assignment	High	2025-10-09 00:00:00.0000000	2025-10-12 00:00:00.0000000	Pending
2	2	Prepare project documentation	Normal	2025-10-10 00:00:00.0000000	2025-10-13 00:00:00.0000000	InProgress
3	3	Team meeting and review	Low	2025-10-11 00:00:00.0000000	2025-10-11 00:00:00.0000000	Completed
4	4	Update GitHub repository with latest code	High	2025-10-12 00:00:00.0000000	2025-10-13 00:00:00.0000000	Pending
5	5	Test all API endpoints in Postman	Normal	2025-10-09 00:00:00.0000000	2025-10-10 00:00:00.0000000	Completed
6	6	Plan new feature for next sprint	Low	2025-10-14 00:00:00.0000000	2025-10-15 00:00:00.0000000	Pending
7	8	string	string	2025-10-09 00:00:00.0000000	2025-10-09 00:00:00.0000000	string

Getting All Tasks:

HTTP <https://localhost:7218/api/ToDo> Save

GET <https://localhost:7218/api/ToDo> Send

Params Authorization Headers (6) Body Pre-request Script Tests Settings Cookies

Body Cookies Headers (4) Test Results Status: 200 OK Time: 9 ms Size: 1.23 KB Save Response

Pretty Raw Preview Visualize JSON

```
1  {
2    {
3      "taskId": 1,
4      "description": "Complete ASP.NET Core assignment",
5      "priority": "High",
6      "startDate": "2025-10-09T00:00:00",
7      "endDate": "2025-10-12T00:00:00",
8      "status": "Pending"
9    },
10   {
11     "taskId": 2,
12     "description": "string",
13     "priority": "string",
14     "startDate": "2025-10-09T00:00:00",
15     "endDate": "2025-10-09T00:00:00",
16     "status": "string"
17   },
18   {
19     "taskId": 3,
20     "description": "Team meeting and review",
21     "priority": "Low",
```

Getting Task By ID:

GET

▼

https://localhost:7218/api/ToDo/5

ParamsAuthorizationHeaders (6)BodyPre-request ScriptTestsSettings

BodyCookiesHeaders (4)Test Results

PrettyRawPreviewVisualizeJSON ▼

1

2

3

4

5

6

7

8

```
{
  "taskId": 5,
  "description": "Test all API endpoints in Postman",
  "priority": "Normal",
  "startDate": "2025-10-09T00:00:00",
  "endDate": "2025-10-10T00:00:00",
  "status": "Completed"
}
```

Creating Task using Post request:

The screenshot displays a REST client interface for a POST request. The URL bar shows the endpoint `https://localhost:7218/api/ToDo/`. The 'Body' tab is selected, and the request body is formatted as JSON. The JSON object contains the following fields: `description` (a string), `priority` (a string), `startDate` (an ISO 8601 timestamp), `endDate` (an ISO 8601 timestamp), and `status` (a string). Below the request body, the 'Test Results' tab is active, showing the response body in a 'Pretty' format. The response is a JSON object that includes a `taskId` (a number) and the same task details as the request body. The status bar at the bottom right indicates a status of 200.

POST ▼ `https://localhost:7218/api/ToDo/`

Params Authorization Headers (8) **Body** ● Pre-request Script Tests Settings

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

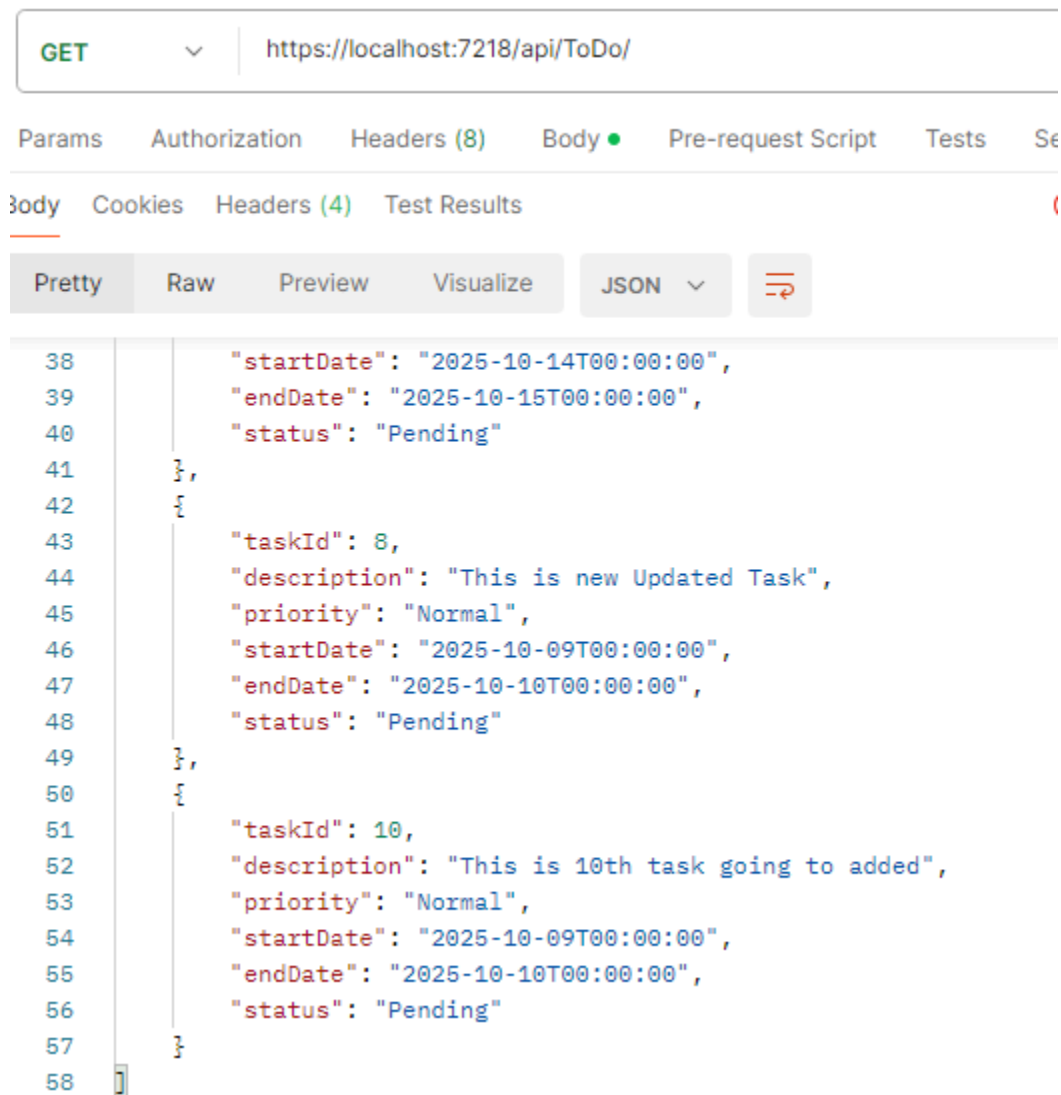
```
1 {
2   "description": "This is 10th task going to added",
3   "priority": "Normal",
4   "startDate": "2025-10-09T00:00:00",
5   "endDate": "2025-10-10T00:00:00",
6   "status": "Pending"
7 }
```

Body Cookies Headers (5) Test Results Status: 2

Pretty Raw Preview Visualize **JSON** ▼

```
1 {
2   "taskId": 10,
3   "description": "This is 10th task going to added",
4   "priority": "Normal",
5   "startDate": "2025-10-09T00:00:00",
6   "endDate": "2025-10-10T00:00:00",
7   "status": "Pending"
8 }
```

The Task is successfully created in DB:



See the above image, we have the 10th task.

Updating Task using Put request:

The screenshot displays a REST client interface with a PUT request to `https://localhost:7218/api/ToDo/8`. The 'Body' tab is selected, showing a JSON payload for updating a task. Below the request, the 'Response' tab is active, showing the server's response in a pretty-printed JSON format.

Request:

Method: PUT
URL: `https://localhost:7218/api/ToDo/8`

Body (JSON):

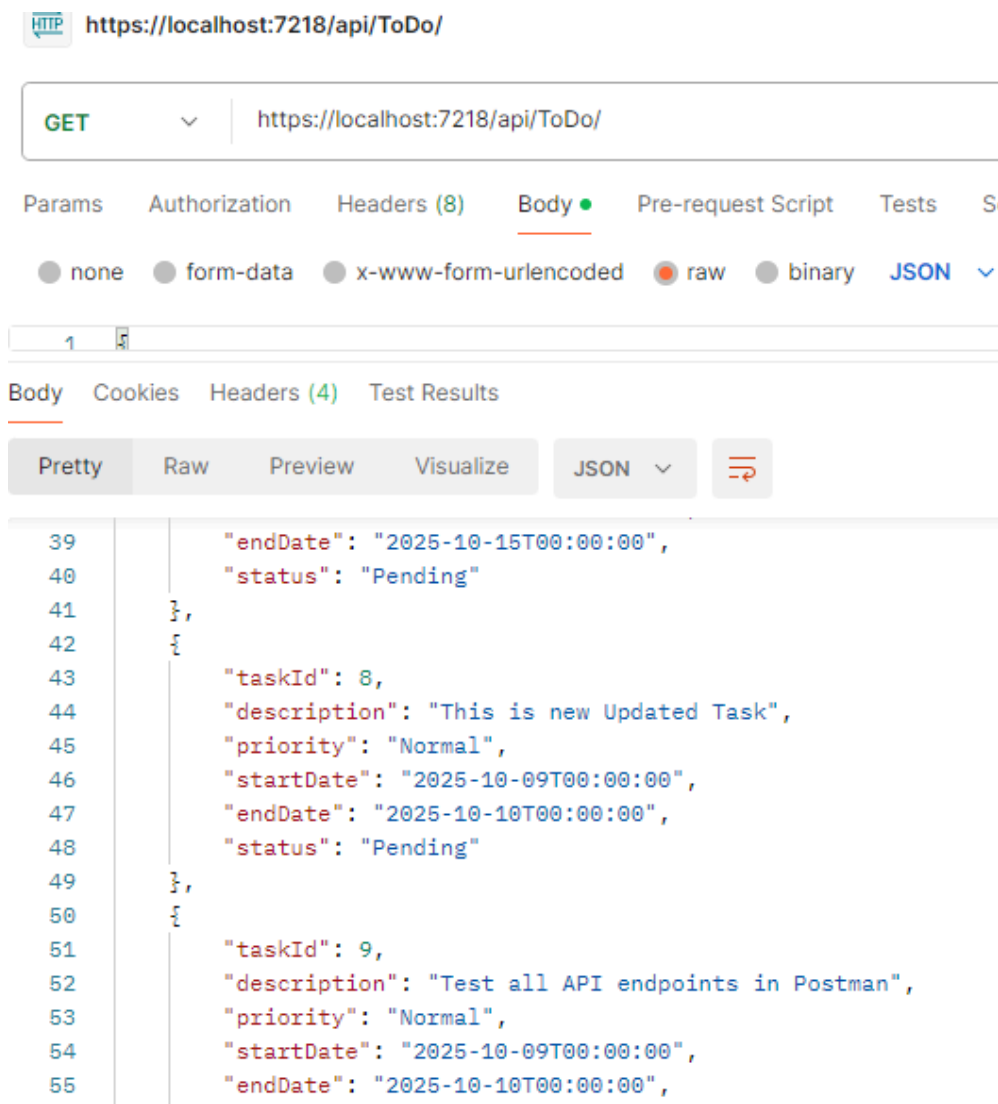
```
1 {  
2   "description": "This is new Updated Task",  
3   "priority": "Normal",  
4   "startDate": "2025-10-09T00:00:00",  
5   "endDate": "2025-10-10T00:00:00",  
6   "status": "Pending"  
7 }
```

Response:

Body (JSON):

```
1 {  
2   "taskId": 8,  
3   "description": "This is new Updated Task",  
4   "priority": "Normal",  
5   "startDate": "2025-10-09T00:00:00",  
6   "endDate": "2025-10-10T00:00:00",  
7   "status": "Pending"  
8 }
```

After updating a task, we get a updated task list:



See the above image, we have the 8th task has updated.

Deleting the Task:

DELETE

https://localhost:7218/api/ToDo/9

Params

Authorization

Headers (8)

Body ●

Pre-request Script

Tests

Settings

☐ none

☐ form-data

☐ x-www-form-urlencoded

☒ raw

☐ binary

JSON

▼

1

5

Body

Cookies

Headers (2)

Test Results

▼

Status: 204 No Content

T

Pretty

Raw

Preview

Visualize

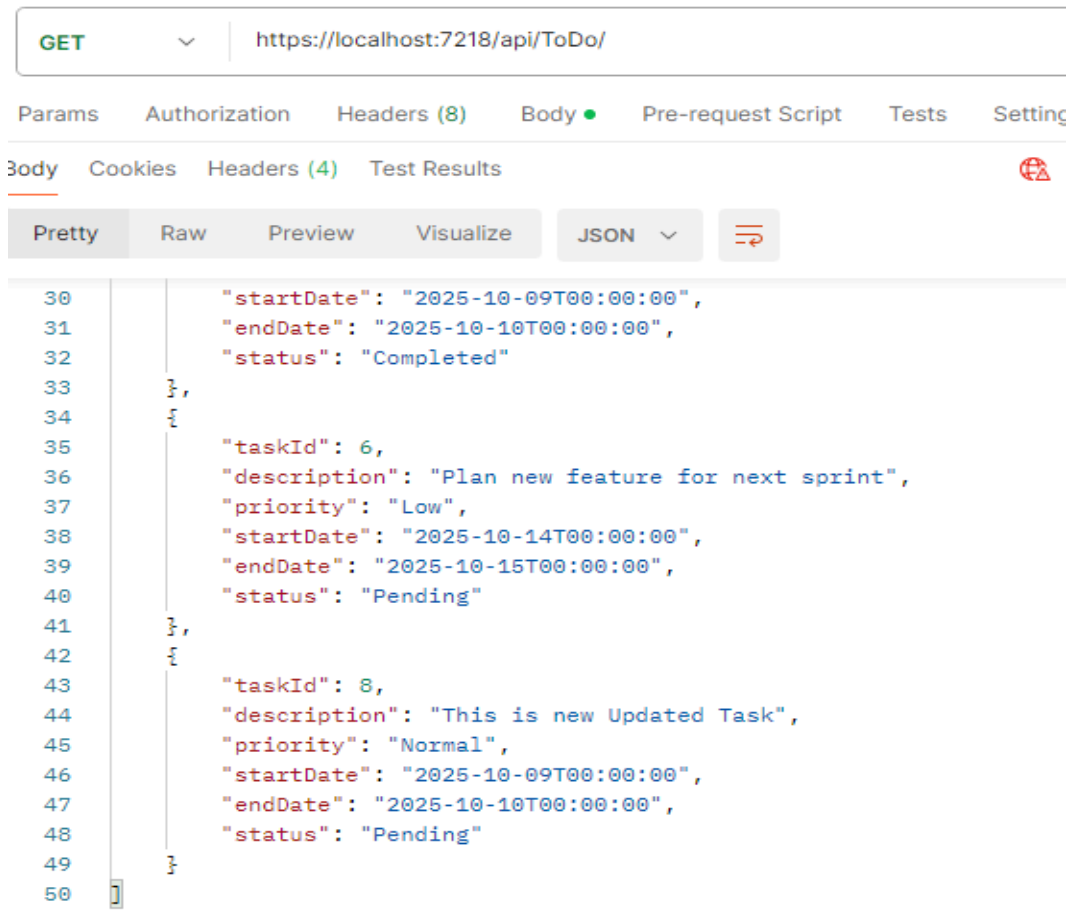
Text

▼

↺

1

After Deleting the Task:



```
GET https://localhost:7218/api/ToDo/

Params  Authorization  Headers (8)  Body •  Pre-request Script  Tests  Settings

Body  Cookies  Headers (4)  Test Results

Pretty  Raw  Preview  Visualize  JSON  ⌵  ⌵

[
  {
    "startDate": "2025-10-09T00:00:00",
    "endDate": "2025-10-10T00:00:00",
    "status": "Completed"
  },
  {
    "taskId": 6,
    "description": "Plan new feature for next sprint",
    "priority": "Low",
    "startDate": "2025-10-14T00:00:00",
    "endDate": "2025-10-15T00:00:00",
    "status": "Pending"
  },
  {
    "taskId": 8,
    "description": "This is new Updated Task",
    "priority": "Normal",
    "startDate": "2025-10-09T00:00:00",
    "endDate": "2025-10-10T00:00:00",
    "status": "Pending"
  }
]
```

See, we get only 8 tasks, 9th Task is deleted in the above image.

Github Link:

<https://github.com/praveen-dotnet-chn/ToDoListApi>