Postman Project Summary: CRUD Operations

This document summarizes the HTTP operations performed in the Postman project so far, including test scripts and use of environments and variables.

# 1. Environment Setup

Environment Name: Development  
  
Variables:  
- base\_url: https://jsonplaceholder.typicode.com  
- todoId: 1

# 2. GET All Todos

Endpoint: GET {{base\_url}}/todos

Tests:

pm.test("Status code is 200", function () {  
 pm.response.to.have.status(200);  
});  
  
pm.test("Response contains array of todos", function () {  
 var jsonData = pm.response.json();  
 pm.expect(jsonData.length).to.be.above(0);  
});

# 3. GET Todo by ID

Endpoint: GET {{base\_url}}/todos/{{todoId}}

Tests:

pm.test("Status is 200", function () {  
 pm.response.to.have.status(200);  
});  
  
pm.test("Response has title", function () {  
 const jsonData = pm.response.json();  
 pm.expect(jsonData).to.have.property("title");  
});

# 4. POST New Todo

Endpoint: POST {{base\_url}}/todos

Body:

{  
 "userId": 1,  
 "title": "Learn Postman CRUD",  
 "completed": false  
}

Tests:

pm.test("Status code is 201", function () {  
 pm.response.to.have.status(201);  
});

# 5. PUT Update Todo

Endpoint: PUT {{base\_url}}/todos/{{todoId}}

Body:

{  
 "userId": 1,  
 "title": "Updated Title from Postman",  
 "completed": true  
}

Tests:

pm.test("Status code is 200", function () {  
 pm.response.to.have.status(200);  
});  
  
pm.test("Title is updated", function () {  
 var jsonData = pm.response.json();  
 pm.expect(jsonData.title).to.eql("Updated Title from Postman");  
});

# 6. DELETE Todo

Endpoint: DELETE {{base\_url}}/todos/{{todoId}}

Tests:

pm.test("Status code is 200 or 204", function () {  
 pm.expect(pm.response.code).to.be.oneOf([200, 204]);  
});  
  
pm.test("Response body is empty or minimal", function () {  
 const resText = pm.response.text();  
 pm.expect(resText === "" || resText === "{}").to.be.true;  
});