1. Endpoint : <https://jsonplaceholder.typicode.com/posts>

Method : GET

Deskripsi : Mendapatkan semua data post.

Script Test :

*const* jsonData = pm.response.json();

*let* resStatus = pm.response.code;

*let* resTime = pm.response.responseTime;

*let* exresponseTime = pm.environment.get("responseTime");

*console*.log(resTime);

pm.test(`Response Status Code is 200`,*function*(){

pm.response.to.have.status(resStatus);

});

pm.test(`Response Time Less Than ${exresponseTime}`, *function*(){

pm.expect(resTime).to.below(*Number*(exresponseTime));

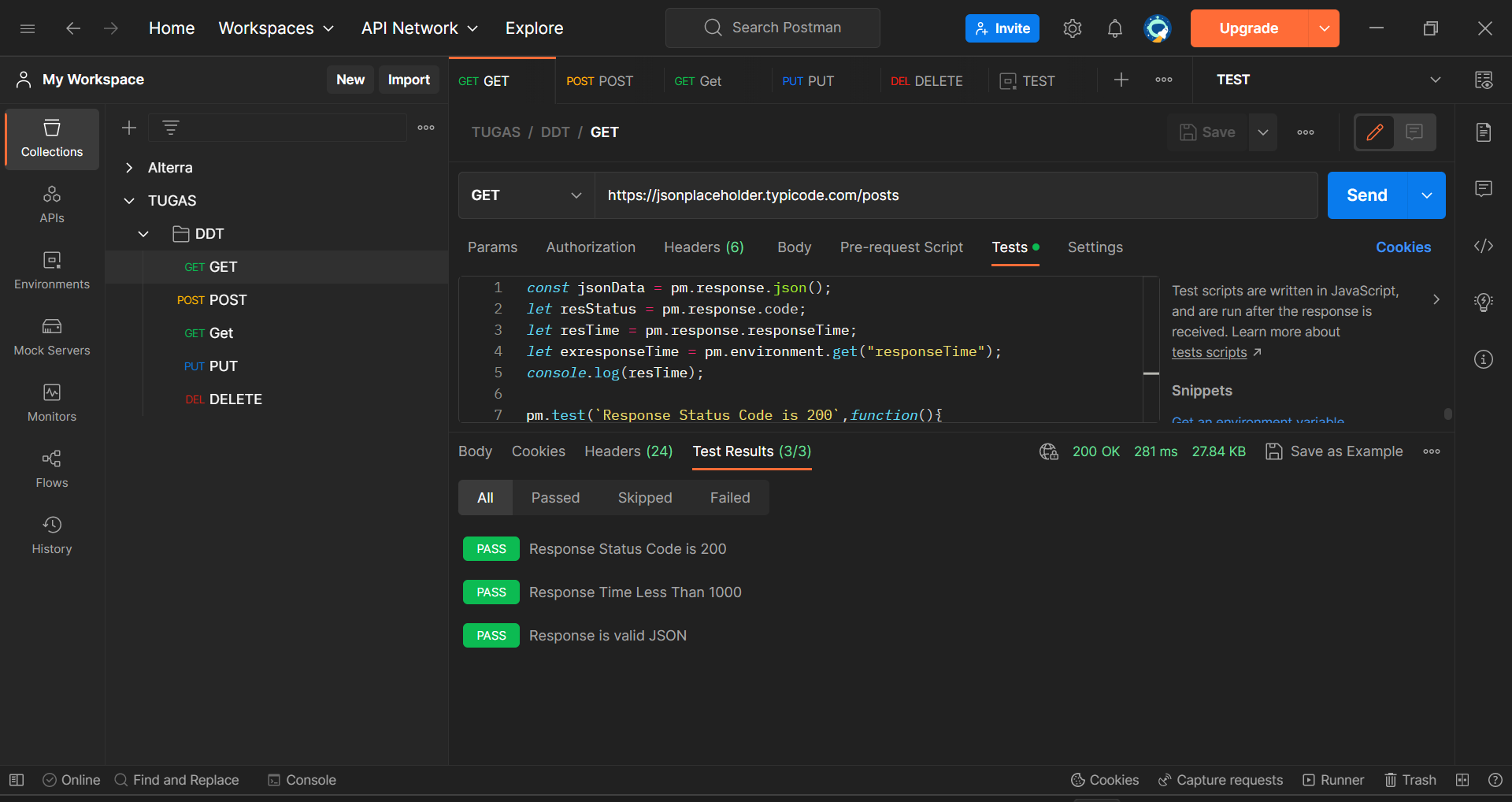
});

pm.test("Response is valid JSON", *function* () {

pm.response.to.be.json;

});

Test Result :



1. Endpoint : <https://jsonplaceholder.typicode.com/posts>

Method : POST

Deskripsi : Membuat data post baru.

Script Test :

*const* jsonData = pm.response.json();

*let* resStatusCode = pm.response.code;

*console*.log(resStatusCode);

if (resStatusCode === 201) {

*let* resTime = pm.response.responseTime;

*let* exresponseTime = pm.environment.get("responseTime");

*let* title = jsonData.title;

*let* body = jsonData.body;

pm.test("Status code is 201", *function* () {

pm.response.to.have.status(201);

});

pm.test(`Response time less than ${exresponseTime}ms`, *function*(){

pm.expect(resTime).to.below(*Number*(exresponseTime));

});

pm.test("Response is valid JSON", *function* () {

pm.response.to.be.json;

});

pm.test("Title is Created", *function* () {

pm.expect(jsonData.title).to.eql("Input title");

});

pm.test("Body is Created", *function* () {

pm.expect(jsonData.body).to.eql("Input body");

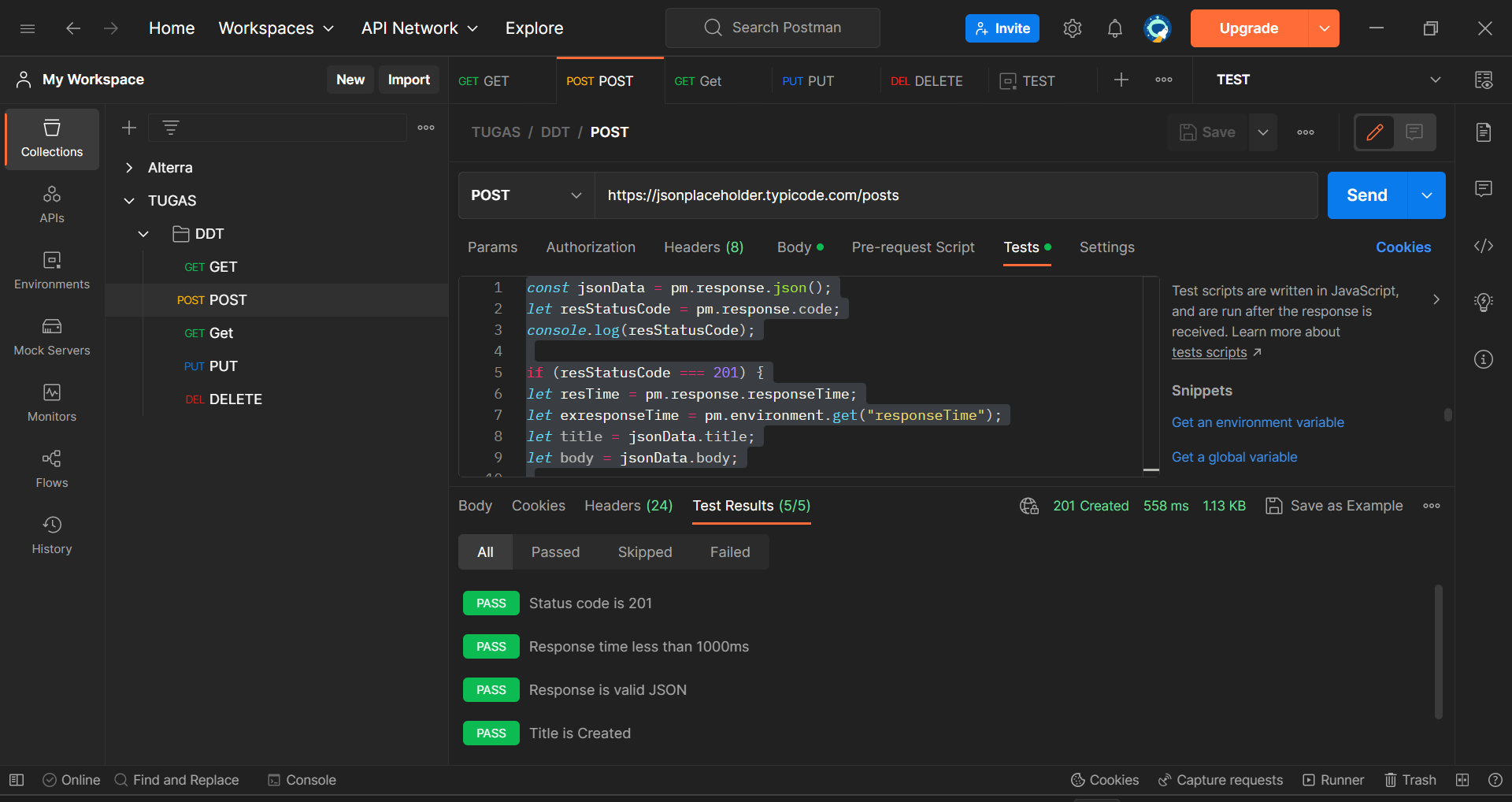
});

} else if (resStatusCode !== 201){

pm.test(`fail to create user`);

};

Test Result :



1. Endpoint : <https://jsonplaceholder.typicode.com/posts/1>

Method : GET

Deskripsi : Mendapatkan data post berdasarkan ID.

Script Test :

*const* jsonData = pm.response.json();

*const* Id = 1;

*let* resStatus = pm.response.code;

*let* resContent = pm.response.headers;

*let* resTime = pm.response.responseTime;

*let* exresponseTime = pm.environment.get("responseTime");

*console*.log(resTime);

pm.test(`Response Status Code is 200`,*function*(){

pm.response.to.have.status(resStatus);

});

pm.test(`Response Time Less Than ${exresponseTime}`, *function*(){

pm.expect(resTime).to.below(*Number*(exresponseTime));

});

pm.test("Response ID is valid", *function*() {

pm.expect(jsonData.id).to.eql(Id);

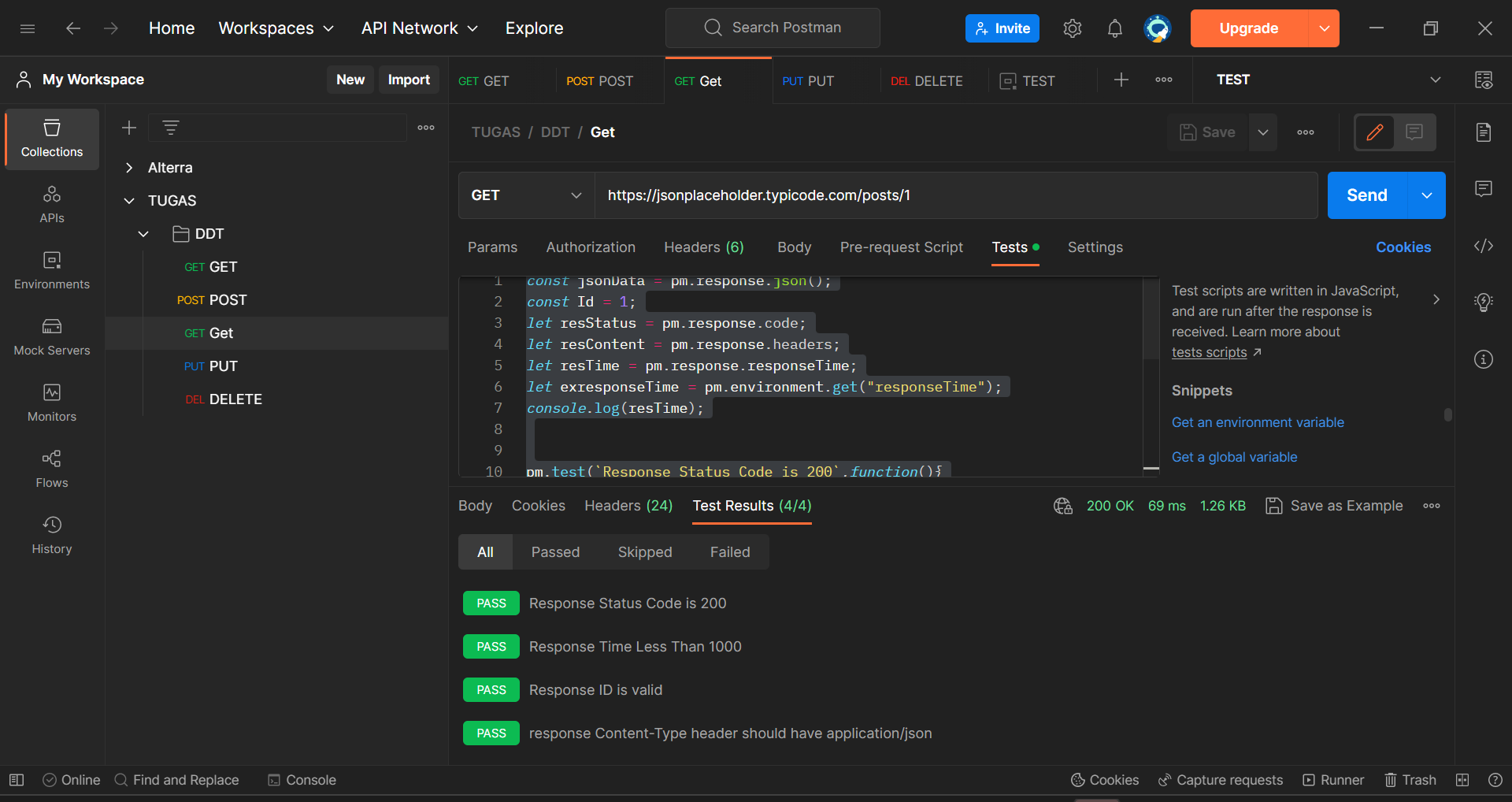
});

pm.test('response Content-Type header should have application/json', *function*(){

pm.expect(pm.response.headers.get('Content-Type')).to.equals('application/json; charset=utf-8')

});

Test Result :



1. Endpoint : <https://jsonplaceholder.typicode.com/posts/1>

Method : PUT

Deskripsi : Mengubah data post berdasarkan ID.

Script Test :

*const* jsonData = pm.response.json();

*let* resStatus = pm.response.code;

*let* title = jsonData.title;

*let* body = jsonData.body;

*let* resTime = pm.response.responseTime;

*let* exresTime = pm.environment.get("responseTime");

pm.test("Status code is 200", *function* () {

pm.response.to.have.status(200);

});

pm.test(`Response Time Less Than ${exresTime}`, *function*(){

pm.expect(resTime).to.below(*Number*(exresTime));

});

pm.test("Response is valid JSON", *function* () {

pm.response.to.be.json;

});

pm.test("Title is updated", *function* () {

pm.expect(jsonData.title).to.eql("Updated Title");

});

pm.test("Body is updated", *function* () {

pm.expect(jsonData.body).to.eql("Updated Body");

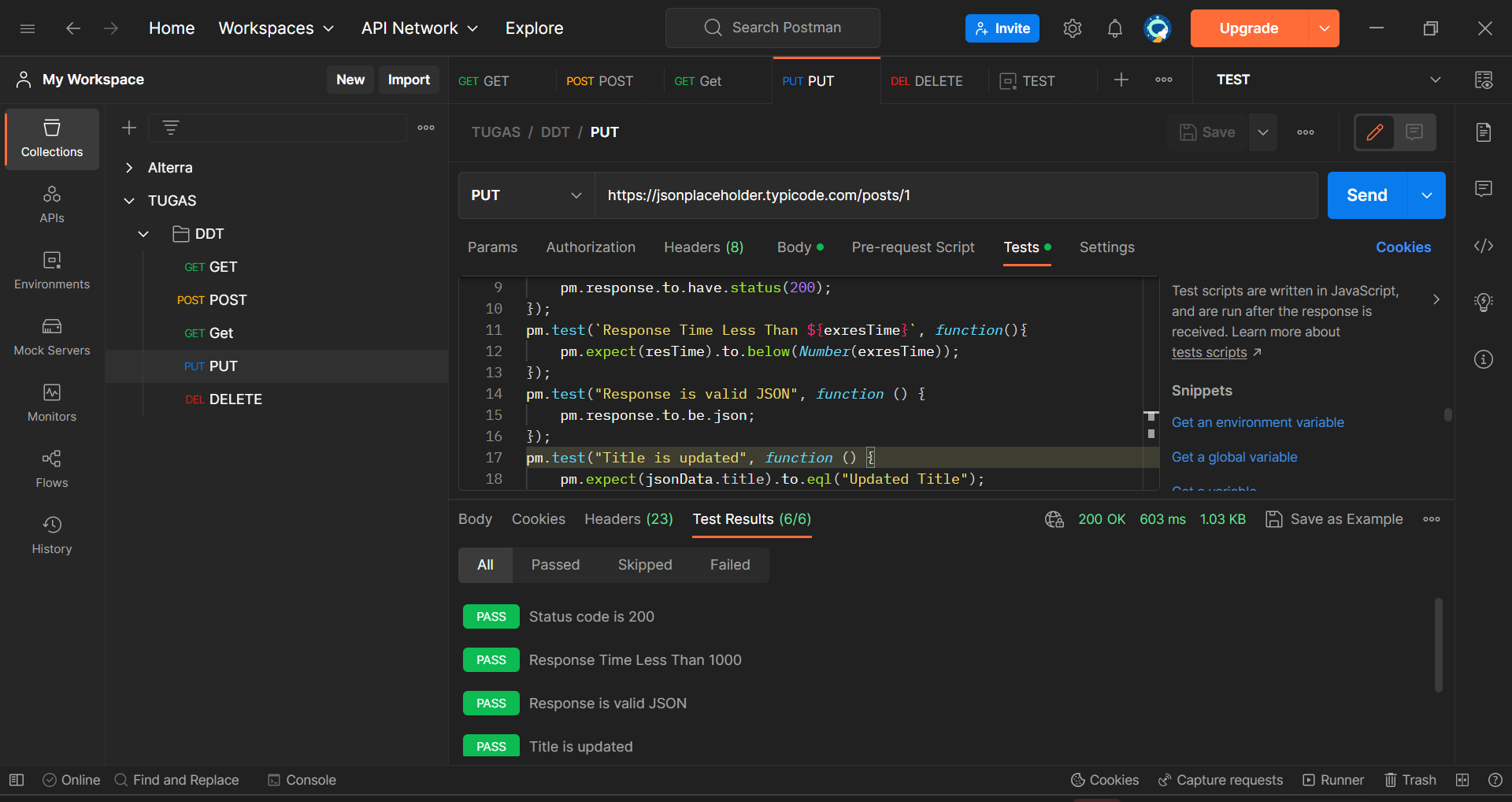
});

pm.test("User ID is 1", *function* () {

pm.expect(jsonData.userId).to.eql(1);

});

Test Result :



1. Endpoint : <https://jsonplaceholder.typicode.com/posts/1>

Method : DELETE

Deskripsi : Menghapus data post berdasarkan ID.

Script Test :

pm.test("Response code is 200", *function* () {

pm.response.to.have.status(200);

});

pm.test("Response has valid data", *function* () {

*var* jsonData = pm.response.json();

pm.expect(jsonData).to.eql({});

});

pm.sendRequest({

url: 'https://jsonplaceholder.typicode.com/posts/1',

method: 'DELETE',

headers: {

'Content-Type': 'application/json'

}

}, *function* (*err*, *res*) {

if (err) {

*console*.log(err);

}

else {

*console*.log(res.json());

}

});

Test Result :

