Week 3 (Automation Testing)

→ Tools Used

- **Automation Tool:** Selenium WebDriver
- **Language:** C#
- **Test Framework:** NUnit
- **API Tool (Optional):** Postman
- **Test Application:** SauceDemo (https://www.saucedemo.com)

→ Automation Testing Overview

Automation testing is used to execute test cases automatically using scripts instead of manual effort. It helps save time, improve accuracy, and support regression testing.

In this project, Selenium WebDriver is used to automate browser actions such as login, navigation, and feature validation.

→Automated Test Cases

Test Case 1: Login Validation

Objective:
Verify that a user can successfully log in using valid credentials.

Steps Automated:

1. Open SauceDemo website
2. Enter valid username
3. Enter valid password
4. Click Login button
5. Verify user is redirected to Products page

Expected Result:
User should log in successfully and see the Products page.

Status: Passed

Test Case 2: Feature Functionality Add Product to Cart

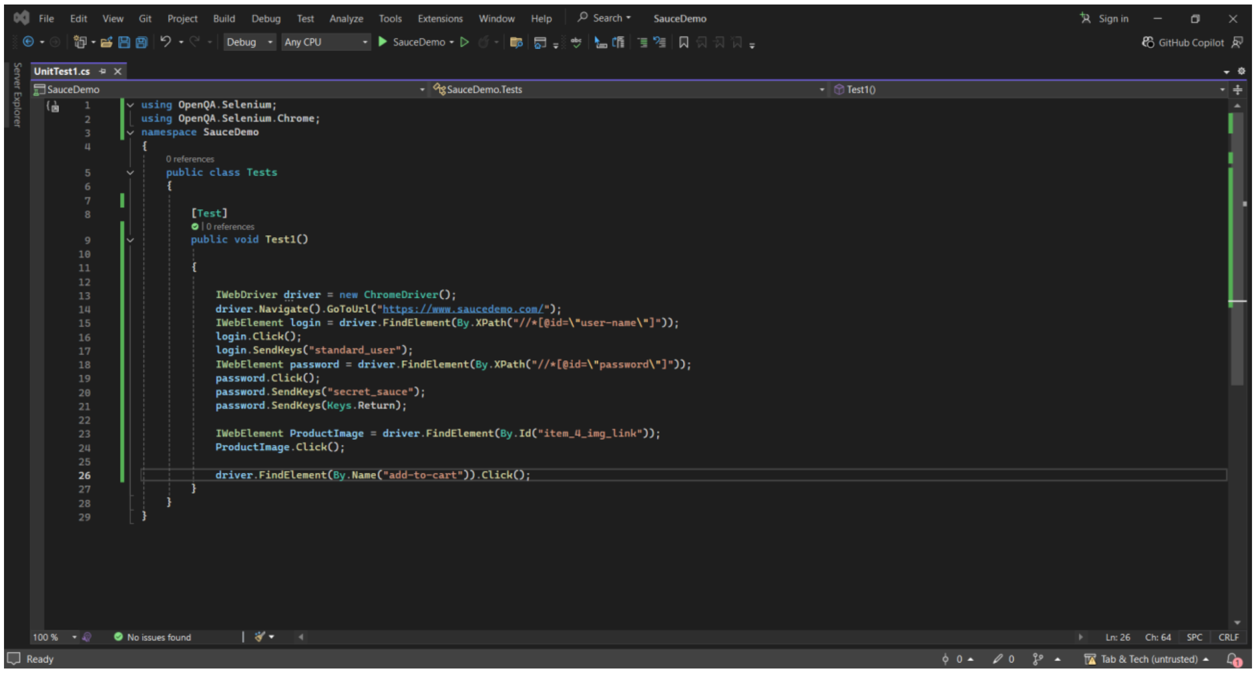
Objective:
Verify that a product can be added to the cart.

Steps Automated:

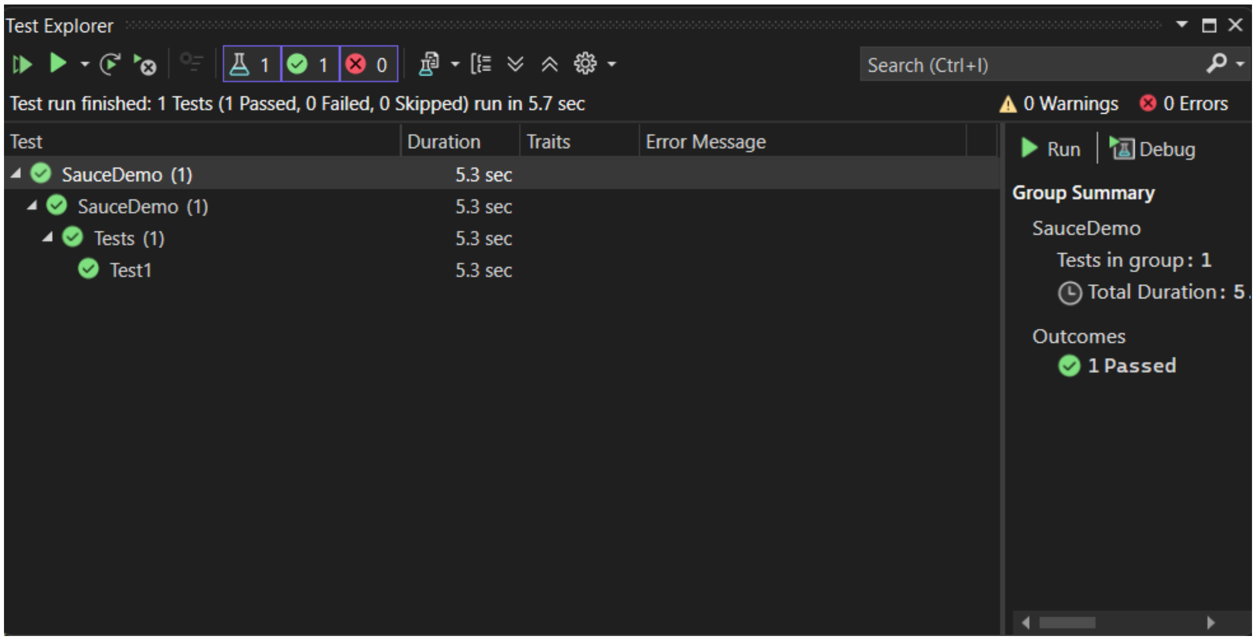
1. Login to the application
2. Click on a product
3. Click “Add to Cart” button
4. Verify cart icon shows item count

Expected Result:
Product should be added to cart successfully.

Status: Passed



```csharp
using OpenQA.Selenium;
using OpenQA.Selenium.Chrome;
namespace SauceDemo
{
    0 references
    public class Tests
    {

        [Test]
        | 0 references
        public void Test1()

        {

            IWebDriver driver = new ChromeDriver();
            driver.Navigate().GoToUrl("https://www.saucedemo.com/");
            IWebElement login = driver.FindElement(By.XPath("//*[@id=\"user-name\"]"));
            login.Click();
            login.SendKeys("standard_user");
            IWebElement password = driver.FindElement(By.XPath("//*[@id=\"password\"]"));
            password.Click();
            password.SendKeys("secret_sauce");
            password.SendKeys(Keys.Return);

            IWebElement ProductImage = driver.FindElement(By.Id("item_4_img_link"));
            ProductImage.Click();

            driver.FindElement(By.Name("add-to-cart")).Click();
        }
    }
}
```


Test Explorer

Test run finished: 1 Tests (1 Passed, 0 Failed, 0 Skipped) run in 5.7 sec

0 Warnings 0 Errors

Test	Duration	Traits	Error Message
SauceDemo (1)	5.3 sec		
SauceDemo (1)	5.3 sec		
Tests (1)	5.3 sec		
Test1	5.3 sec		

Group Summary
SauceDemo
Tests in group: 1
Total Duration: 5

Outcomes
1 Passed

➔ **SCRIPT:**

```
using OpenQA.Selenium;
using OpenQA.Selenium.Chrome;
namespace SauceDemo
{
    public class Tests
    {
        [Test]
        public void Test1()
        {
            IWebDriver driver = new ChromeDriver();
            driver.Navigate().GoToUrl("https://www.saucedemo.com/");
            IWebElement login = driver.FindElement(By.XPath("//*[@id=\"user-name\"]"));
            login.Click();
            login.SendKeys("standard_user");
            IWebElement password = driver.FindElement(By.XPath("//*[@id=\"password\"]"));
            password.Click();
            password.SendKeys("secret_sauce");
            password.SendKeys(Keys.Return);

            IWebElement ProductImage = driver.FindElement(By.Id("item_4_img_link"));
            ProductImage.Click();

            driver.FindElement(By.Name("add-to-cart")).Click();
        }
    }}
```

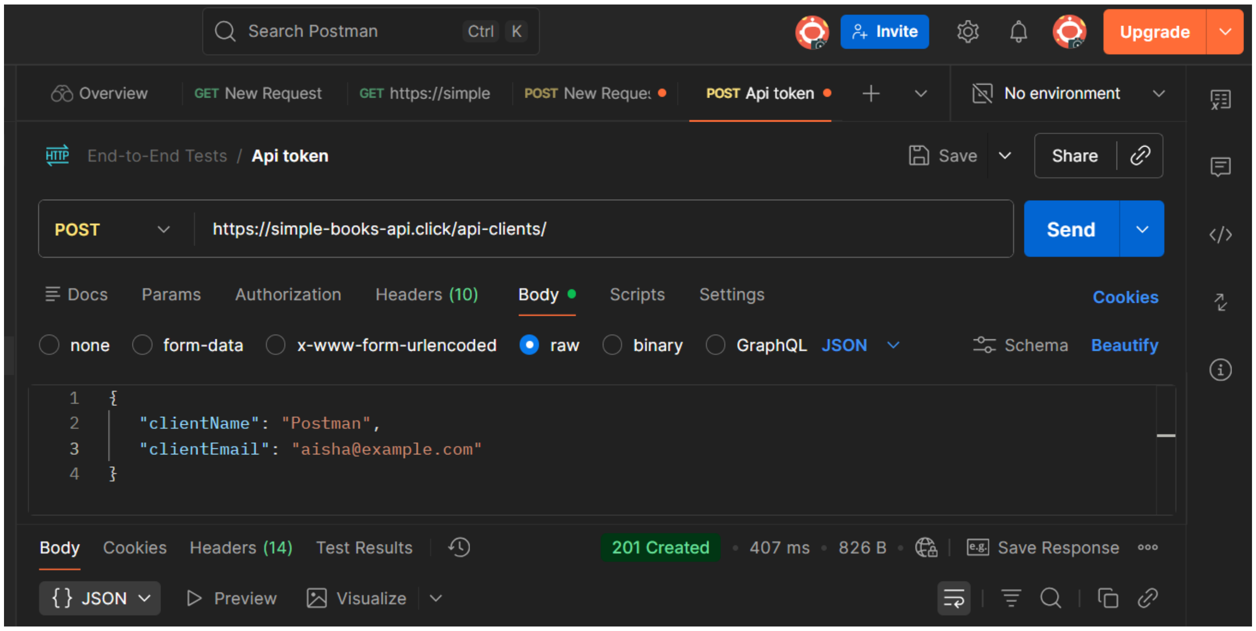
→API Testing Using Postman (Simple Books API)

API Reference Used

For API testing, the **Simple Books API** was used.
This API provides endpoints for practicing common HTTP methods such as **GET, POST, PUT, and DELETE**.

API Documentation Source:
Simple Books API , Introduction to Postman Course

➔ Post (Token Generation):

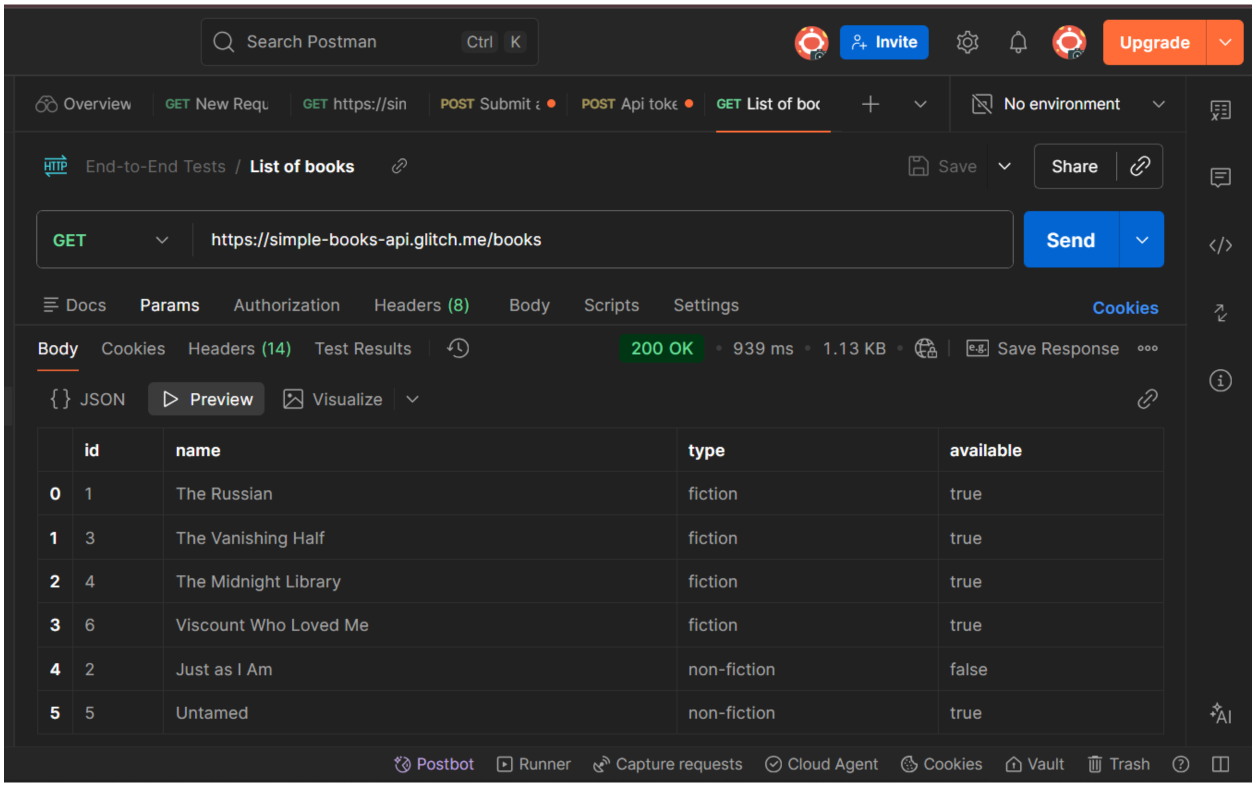


POST https://simple-books-api.click/api-clients/

```
{
  "clientName": "Postman",
  "clientEmail": "aisha@example.com"
}
```

201 Created · 407 ms · 826 B

➔ GET (List Of Books):

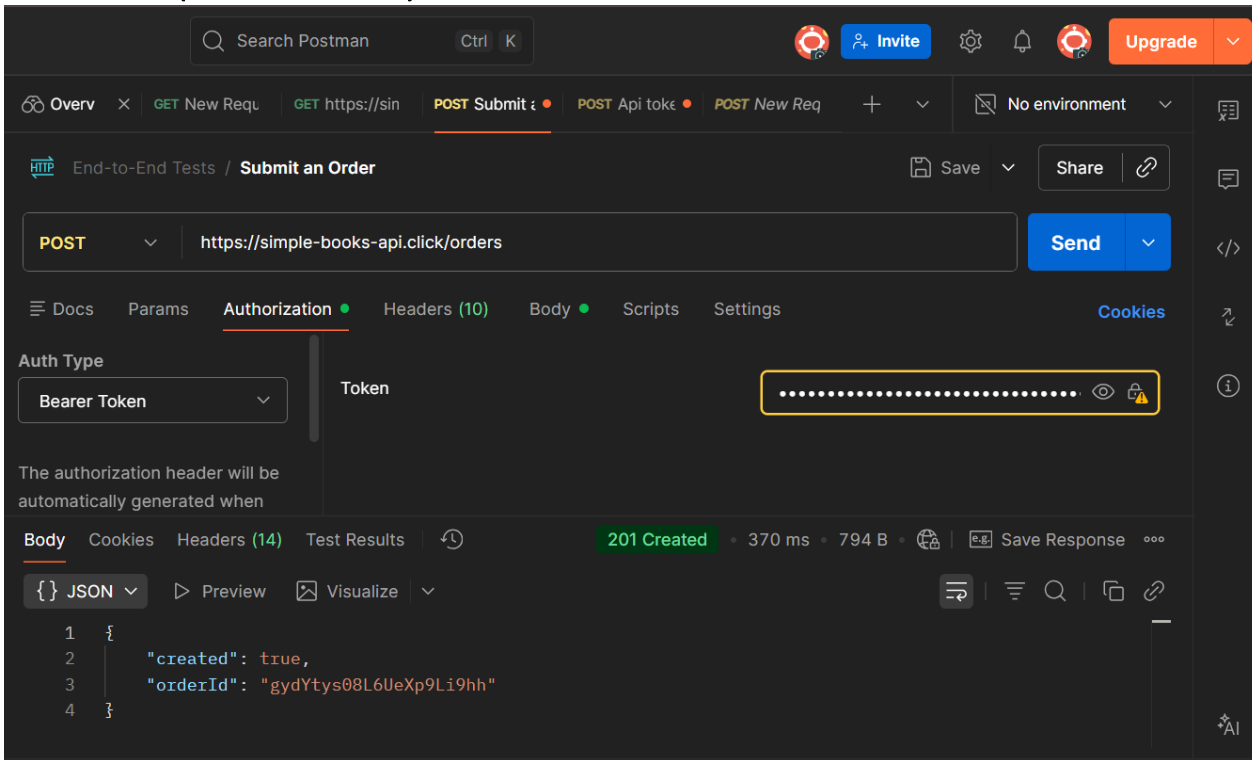


GET https://simple-books-api.glitch.me/books

200 OK • 939 ms • 1.13 KB

	id	name	type	available
0	1	The Russian	fiction	true
1	3	The Vanishing Half	fiction	true
2	4	The Midnight Library	fiction	true
3	6	Viscount Who Loved Me	fiction	true
4	2	Just as I Am	non-fiction	false
5	5	Untamed	non-fiction	true

➔ **POST (Submit an Order):**



POST https://simple-books-api.click/orders

Auth Type: Bearer Token

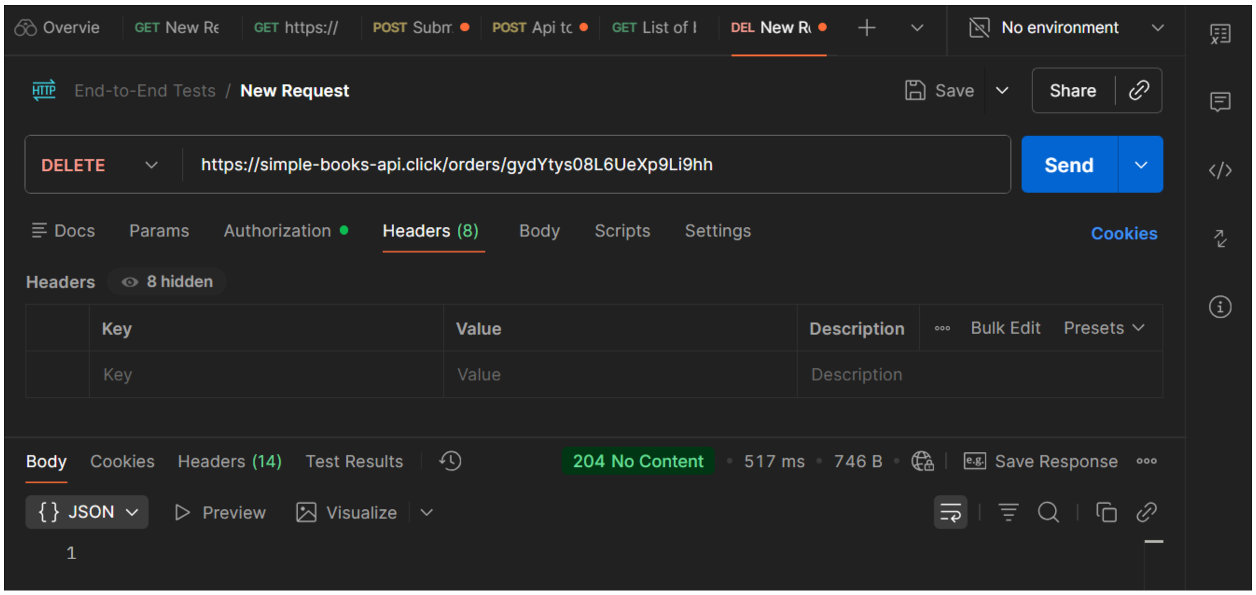
Token: ••••••••••••••••••••••••••••••••••

The authorization header will be automatically generated when

201 Created • 370 ms • 794 B

```
{
    "created": true,
    "orderId": "gydYtys08L6UeXp9Li9hh"
}
```

➔ **DELETE (Delete an order):**



End-to-End Tests / New Request

DELETE https://simple-books-api.click/orders/gydYtys08L6UeXp9Li9hh

Headers

Key	Value	Description
Key	Value	Description

204 No Content • 517 ms • 746 B