

# Hooks in MOCHA

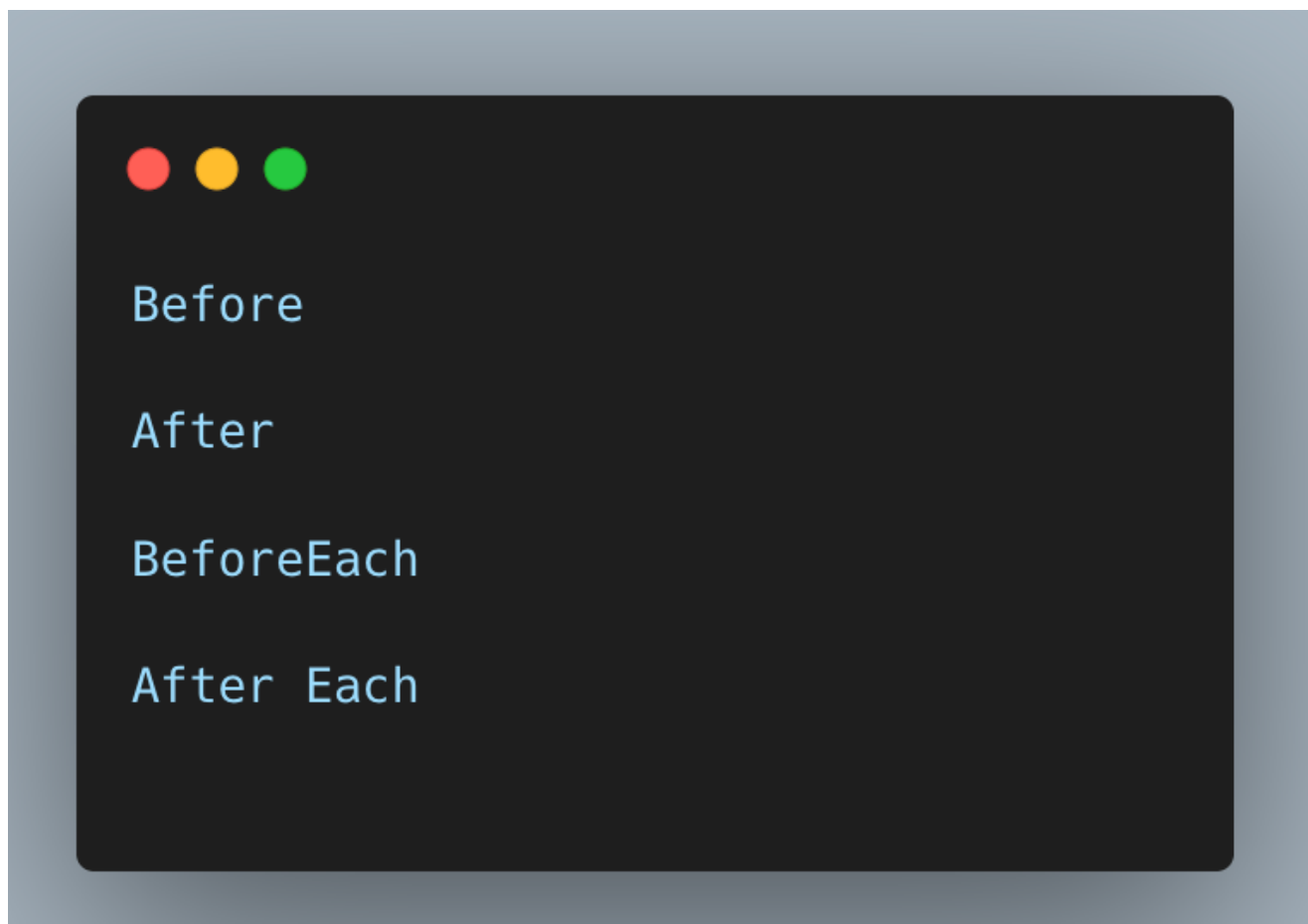


# ***Introduction***

Mocha is a popular JavaScript test framework that provides a flexible and powerful testing environment for your code, using which you can write tests clearly and concisely using the BDD (behavior-driven development) style.

In Mocha, Hooks are the functions using which you can define actions that run before or after specific test cases or test suites and provides a convenient way to set up the test environment and perform cleanup tasks.

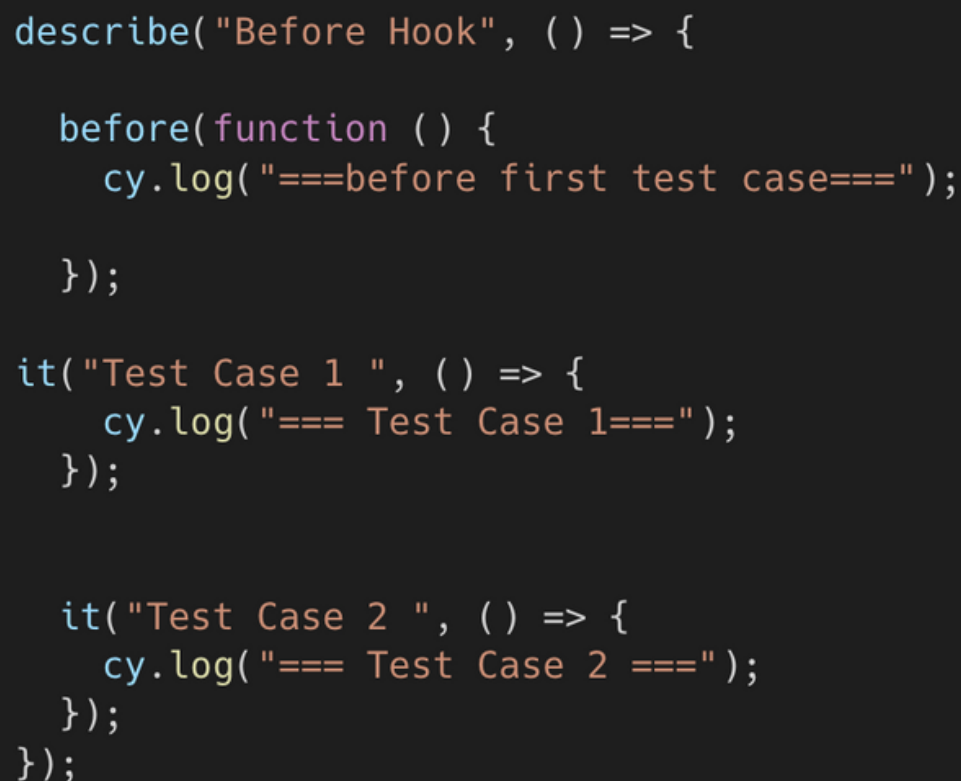
Below are the four types of MOCHA Hooks:



# *before*

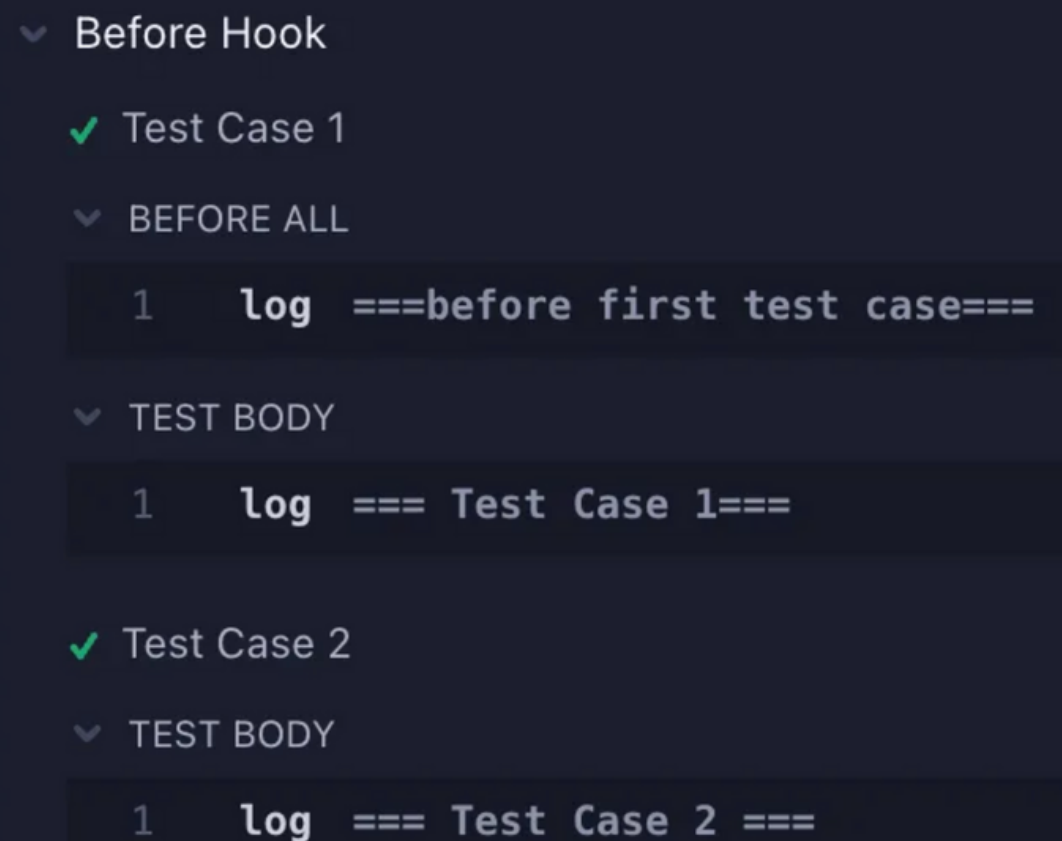
This hook runs once before the first tests in a test suite. It's useful for setting up common conditions that are required for all tests

## Example :



```
describe("Before Hook", () => {  
  before(function () {  
    cy.log("===before first test case===");  
  });  
  
  it("Test Case 1 ", () => {  
    cy.log("=== Test Case 1===");  
  });  
  
  it("Test Case 2 ", () => {  
    cy.log("=== Test Case 2 ===");  
  });  
});
```

## Code



```
▼ Before Hook  
✓ Test Case 1  
▼ BEFORE ALL  
1 log ===before first test case===  
▼ TEST BODY  
1 log === Test Case 1===  
✓ Test Case 2  
▼ TEST BODY  
1 log === Test Case 2 ===
```

## Execution

# beforeEach

This hook runs before each test in a test suite. It's useful for setting up conditions that are required for individual tests.

## Example :

```
describe("Before Each", () =>
{
  beforeEach(function ()
  {
    cy.log("===before each test case===");
  });

  it("Test Case 1 ", () => {
    cy.log("=== Test Case 1===");
  });

  it("Test Case 2 ", () => {
    cy.log("=== Test Case 2 ===");
  });
});
```

## Code

```
▼ Before Each
✓ Test Case 1
  ▼ BEFORE EACH
    1    log    ===before each test case===

  ▼ TEST BODY
    1    log    === Test Case 1===

✓ Test Case 2
  ▼ BEFORE EACH
    1    log    ===before each test case===

  ▼ TEST BODY
    1    log    === Test Case 2 ===
```

## Execution

Anshita Bhasin

# after

This hook runs once after the last tests in a test suite. It's useful for cleaning up after all tests have run.

## Example :

```
describe("After Hook", () =>
{
  after(function () {
    cy.log("=== After last test case===");
  });

  it("Test Case 1 ", () => {
    cy.log("=== Test Case 1===");
  });

  it("Test Case 2 ", () => {
    cy.log("=== Test Case 2 ===");
  });
});
```

## Code

```
▼ After Hook
✓ Test Case 1
▼ TEST BODY
1    log    === Test Case 1===

✓ Test Case 2
▼ TEST BODY
1    log    === Test Case 2 ===

▼ AFTER ALL
1    log    === After last test case===
```

## Execution

# afterEach

This hook runs after each test in a test suite. It's useful for cleaning up after individual tests.

## Example :



```
describe("After Each", () =>
{
  afterEach(function ()
  {
    cy.log("=== After Each test case===");
  });

  it("Test Case 1 ", () =>
  {
    cy.log("=== Test Case 1===");
  });

  it("Test Case 2 ", () =>
  {
    cy.log("=== Test Case 2 ===");
  });
});
```

## Code



```
▼ After Each
✓ Test Case 1
  ▼ TEST BODY
    1    log  === Test Case 1===
  ▼ AFTER EACH
    1    log  === After Each test case===
✓ Test Case 2
  ▼ TEST BODY
    1    log  === Test Case 2 ===
  ▼ AFTER EACH
    1    log  === After Each test case===
```

## Execution

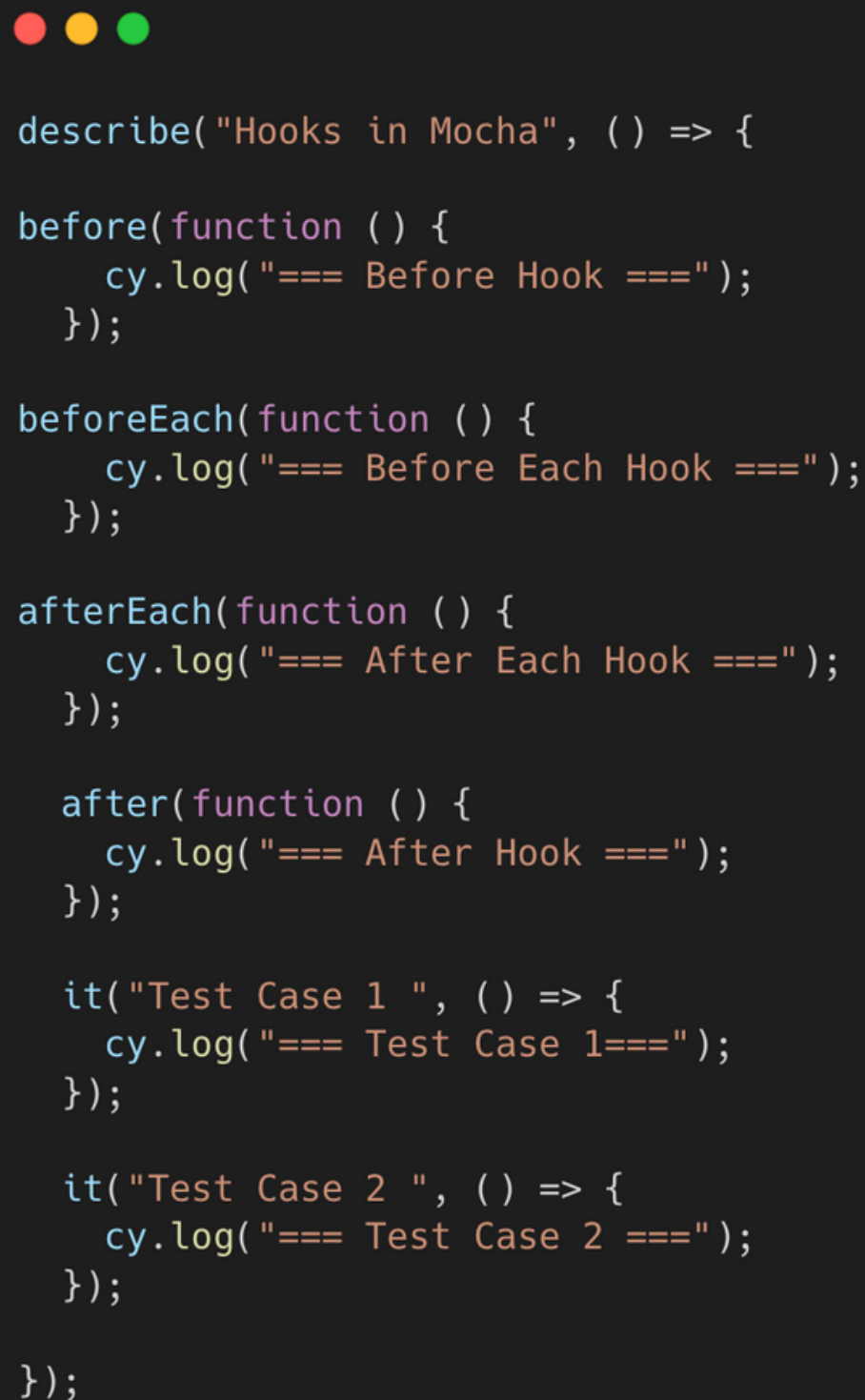
# Execution sequence of Mocha Hooks :

The sequence of hooks in Mocha is important because it affects the way tests are executed. The following is the order in which hooks are executed in Mocha:



# Execution sequence of Mocha Hooks :

## Code



```
describe("Hooks in Mocha", () => {  
  
  before(function () {  
    cy.log("=== Before Hook ===");  
  });  
  
  beforeEach(function () {  
    cy.log("=== Before Each Hook ===");  
  });  
  
  afterEach(function () {  
    cy.log("=== After Each Hook ===");  
  });  
  
  after(function () {  
    cy.log("=== After Hook ===");  
  });  
  
  it("Test Case 1 ", () => {  
    cy.log("=== Test Case 1===");  
  });  
  
  it("Test Case 2 ", () => {  
    cy.log("=== Test Case 2 ===");  
  });  
  
});
```



# Execution sequence of Mocha Hooks :

## Execution

```

  Hooks in Mocha
  ✓ Test Case 1
  BEFORE ALL
    1 log === Before Hook ===
  BEFORE EACH
    1 log === Before Each Hook ===
  TEST BODY
    1 log === Test Case 1===
  AFTER EACH
    1 log === After Each Hook ===
  ✓ Test Case 2
  BEFORE EACH
    1 log === Before Each Hook ===
  TEST BODY
    1 log === Test Case 2 ===
  AFTER EACH
    1 log === After Each Hook ===
  AFTER ALL
    1 log === After Hook ===

```

**For a more detailed understanding of MOCHA Hooks, please refer to the video/blog link provided below.**

**Video : <https://youtu.be/xK438tnQxDA>**

**Blog : <https://youtu.be/xK438tnQxDA>**



**Anshita Bhasin**