

Topic: Youtube

Authentication

```
describe("User Authentication", () => {
  beforeEach(() => {
    // Set up the test data using a fixture
    cy.fixture("user.json").as("user");
  });

  it("Should login successfully", () => {
    // Clear all cookies and local storage before the test
    cy.clearCookies();
    cy.clearLocalStorage();

    // Set the viewport to a specific device
    cy.viewport("iphone-6");

    // Visit the login page
    cy.visit("/login");

    // Enter the username and password from the fixture
    cy.get("#username").type(this.user.username);
    cy.get("#password").type(this.user.password);

    // Submit the login form
    cy.get('button[type="submit"]').click();

    // Assert that the user is logged in
    cy.contains("Welcome, Muhammad Azfar").should("be.visible");
  });

  it("Should display error message for invalid login", () => {
    // Clear all cookies and local storage before the test
    cy.clearCookies();
    cy.clearLocalStorage();

    // Set the viewport to a specific device
    cy.viewport("macbook-15");
```

```
// Visit the login page
cy.visit("/login");

// Enter invalid username and password
cy.get("#username").type("invalid_user");
cy.get("#password").type("invalid_password");

// Submit the login form
cy.get('button[type="submit"]').click();

// Assert that an error message is displayed
cy.contains("Invalid username or password").should("be.visible");
});
});
```

Creating Channel

```
describe('Create YouTube Channel', () => {
  it('Create a New Channel', () => {
    // Visit YouTube homepage
    cy.visit('https://www.youtube.com/');

    // Click on the "Sign In" button
    cy.get('ytd-button-renderer#text.style-scope.ytd-masthead').click();

    // Enter login credentials and sign in

    // Click on the user profile icon
    cy.get('ytd-button-renderer#avatar.style-scope.ytd-masthead').click();

    // Click on "Your Channel" option
    cy.contains('Your channel').click();

    // Click on "Create channel" button
    cy.contains('Create channel').click();

    // Enter channel name and description
    cy.get('input#channel-name').type('My New Channel');
    cy.get('textarea#channel-description').type('This is my new YouTube channel');
  });
});
```

```
// Click on "Create" button
cy.get('button#submit-button').click();

// Assert that the channel creation is successful
cy.contains('Your channel has been created').should('be.visible');
cy.contains('My New Channel').should('be.visible');
});
});
```

Adding comment

```
describe('YouTube Comments', () => {

  it('Add Comment to a Video', () => {
    // Visit the YouTube video page
    cy.visit('https://www.youtube.com/watch?v=UFqUCGYbVpk');

    // Scroll to the comments section
    cy.get('#comments').scrollIntoView();

    // Type the comment in the input box
    cy.get('#simplebox-placeholder').type('This is my comment');

    // Submit the comment
    cy.get('#submit-button').click();

    // Assert that the comment is visible
    cy.contains('This is my comment').should('be.visible');
  });
});
```

Correct Title

```
describe('YouTube Test Suite', () => {
  it('should display the correct title', () => {

    // Visit the YouTube homepage
    cy.visit('https://www.youtube.com')

    // Assert that the title contains 'YouTube'
    cy.title().should('include', 'YouTube')
  })
})
```

Like & Subscribe

```
describe('YouTube Test', () => {
  it('Like a Video', () => {
    // Visit the YouTube video page
    cy.visit('https://www.youtube.com/watch?v=a-7I7_NjPMg');

    // Wait for 5 seconds
    cy.wait(5000);

    // Click on the like button of the video
    cy.get('div[class="yt-spec-touch-feedback-shape yt-spec-touch-feedback-shape-
-touch-response"]').first().click({ force: true });
  });

  // Visit the YouTube video page
  it('Subscribe to a Channel', () => {
    cy.visit('https://www.youtube.com/watch?v=a-7I7_NjPMg');

    // Click on the subscribe button of the channel
    cy.get('#subscribe-button').click({ force: true });
  });
});
```

Login

```
describe('YouTube Login', () => {
  it('Login with valid credentials', () => {
    // Visit the YouTube homepage
    cy.visit('https://www.youtube.com/');

    // Click on the login button
    cy.get('a[class="yt-spec-button-shape-next yt-spec-button-shape-next--outline yt-spec-button-shape-next--call-to-action yt-spec-button-shape-next--size-m yt-spec-button-shape-next--icon-leading "]').first().click();

    // Get the URL of the redirected page and visit it
    cy.location().then((location) => {
      const url = location.href;
      // Log the URL for debugging purposes
      console.log(url);
      cy.visit(url);
    });

    // Click on the 'Sign in' button
    cy.contains('Sign in').click();

    // Enter the username
    cy.get('#identifierId').type('azfar425');

    // Click on the 'Next' button
    cy.contains('Next').click();

    // Enter the password
    cy.get('input[type="password"]').type('@zfar425');

    // Click on the 'Next' button
    cy.contains('Next').click();

    // Assert that the 'Home' page is visible after successful login
    cy.contains('Home').should('be.visible');
  });
});
```

Open Menu

```
describe('YouTube Menu', () => {
  it('Open YouTube Menu', () => {

    // Visit the YouTube homepage
    cy.visit('https://www.youtube.com/');

    // Click on the YouTube menu icon
    cy.get('#guide-icon').click({ force: true });
  });
});
```

Intercept video playback API

```
describe("YouTube Video Playback", () => {
  it("Should intercept and mock video playback API", () => {
    // Intercept the video playback API request and mock the response
    //we use the cy.intercept() command to intercept the GET request to the video
    // playback API (get_video_info)
    cy.intercept("GET", "**/get_video_info*", (req) => {
      req.reply((res) => {
        // Modify the response body to simulate a successful video playback
        const modifiedBody = res.body.replace("status=fail", "status=ok");
        res.body = modifiedBody;
      });
    }).as("videoPlayback");

    // Visit the video page
    cy.visit("https://www.youtube.com/watch?v=abcdef12345");

    // Assert that the video playback request was intercepted
    cy.wait("@videoPlayback").then((interception) => {
      const response = interception.response;
      expect(response.statusCode).to.equal(200);
      expect(response.body).to.include("status=ok");
    });

    // Assert that the video is playing
    cy.get("video").should("be.visible");
    cy.get("video").should("have.attr", "src");
  });
});
```

```
});  
});
```

Search & Play

```
describe('YouTube Test', () => {  
  it('Search and Verify Search Results', () => {  
    // Define the search query  
    const searchQuery = 'weather';  
  
    // Visit the YouTube homepage  
    cy.visit('https://youtube.com');  
  
    // Type the search query in the search input field and press 'Enter'  
    cy.get('input#search').should('be.visible').type(`${searchQuery}{enter}`);  
  
    // Assert that the search results container is visible  
    cy.get('#contents').should('be.visible');  
  });  
  
  // Define the search query  
  it('Play random Video from Search Results', () => {  
    const searchQuery = 'Advent of Islam';  
  
    // Visit the YouTube homepage  
    cy.visit('https://youtube.com');  
  
    // Type the search query in the search input field and press 'Enter'  
    cy.get('input#search').should('be.visible').type(`${searchQuery}{enter}`);  
  
    // Assert that the search results container is visible  
    cy.get('#contents').should('be.visible');  
  
    // Click on the first video from the search results  
    cy.get('#contents ytd-video-renderer').eq(0).click();  
  
    // Assert that the video player is visible  
    cy.get('#movie_player').should('be.visible');
```

```
// Assert that the video player is playing (either state 1 or 3)
//We assert that the video player is in either state 1 (playing) or state 3
(buffering)
// using a custom assertion with cy.get().should() and expect().to.be.oneOf()
cy.get('#movie_player').should(($player) => {
  expect($player.get(0).getPlayerState()).to.be.oneOf([1, 3]);
});

});
});
```

Recommended Video

```
//verifying that the second recommended video on the YouTube homepage is the same
as
// the video displayed on its video page.

it('should verify the second recommended video', () => {
  // Visit the YouTube homepage
  cy.visit('https://www.youtube.com')

  // Get the list of recommended videos
  cy.get('#video-title').eq(1).invoke('text').as('secondVideoTitle')

  // Click on the second recommended video
  cy.get('#thumbnail').eq(1).click()

  // Get the title of the video from the video page
  cy.get('.title.style-scope.ytd-video-primary-info-renderer').invoke('text').as('videoTitle')

  // Compare the titles
  cy.get('@secondVideoTitle').then((secondVideoTitle) => {
    cy.get('@videoTitle').should('eq', secondVideoTitle)
  })
})
```


Saving for Watch Later

```
describe('Save Video to Watch Later on YouTube', () => {
  it('Should save a random video to Watch Later', () => {
    // Visit YouTube website
    cy.visit('https://www.youtube.com/');

    // Wait for 5 seconds (may be used to allow page elements to load)
    cy.wait(5000);

    // Click on the three dots (more options) button
    cy.get('Button[aria-label="Action menu"]')
      .click();

    // Click on the "Save to Watch later" option from the menu
    cy.get('.style-scope.ytd-menu-service-item-renderer')
      .contains('Save to Watch later')
      .click();

  });

  it("Complete Sprint", () => {

    // Visit YouTube website
    cy.visit("https://www.youtube.com/");

    // Click on the three dots (more options) button of the first video
    cy.get('button[aria-label="Action menu"]').first().click();

    // Click on the first menu item button
    cy.get('button[class="menu-item-button"]').first().click();
  })
});
```

Start & Stop

```
describe('Play and Stop Video', () => {
  it('Should play the video and stop after 5 seconds', () => {
    // Visit the YouTube video URL
    cy.visit('https://www.youtube.com/watch?v=qjhf0bj3yzQ&ab_channel=Automation
StepbyStep');

    // Click on the video player to start playing
    cy.get('video').click();

    // Wait for 5 seconds
    cy.wait(5000);

    // Pause the video
    cy.get('video').invoke('pause');
  });
});
```