

YAKSHA

APP WITH JAVASCRIPT AND CYPRESS PL1-8

Use case summary

Project Name:opensource-demo.orangehrmlive.com app – HR management system.

Use Case Summary: opensource-demo.orangehrmlive.com is designed to manage Employee records.

Records (HRMS). It allows users to view, search, and manage employee records. It features functionality such as adding/editing employee details, filtering data by department and role, and exporting records. The primary use case is to automate the process of employee data management, ensuring efficient and reliable operations for human resources teams in organizations

Technology Stack:

- **Automation Tool:** Cypress (for testing)

Key Features:

- **Patient Record Management:** Add, edit, and delete Employee records.
- **Filtering and Search:** Search Employee records by date range, name, department, and more.
- **Export Functionality:** Export records for offline access.

Expected Outcomes:

- Automate key HR operations like employee record handling, filtering, and validation.
- Ensure the accurate retrieval and modification of employee records, enhancing operational efficiency.

Overview of the application

Pages/Features that are to be focused for the application

Please use the Application URL <https://opensource-demo.orangehrmlive.com>

PROBLEM STATEMENT

Need to automate the following activities using cypress+javascript

You will be given a few Json files in the fixtures folder like LoginData.json Etc.

Path	File	Description
\fixtures	LoginData.json TestCase_9.pdf	1. Contains data to read from the json file.

PageObjects\Pages	<ul style="list-style-type: none"> LoginProfilePage ProfilePage 	<ol style="list-style-type: none"> All core activities to be performed here. The comments associated with each templated method here describe the expectation. Declare any variable/object you need to share data/status between different methods. <p>Do not modify the signature of methods declared here.</p>
-------------------	---	---

Here's a detailed table format for the test cases to be tested

Test Case No.	Test Case Name	Test Steps to be performed	Path & Method Used	Expected Result
1	Verify the 'My Info' tab Loads Successfully	<ol style="list-style-type: none"> Login using valid credentials 2. Click on "My Info" tab <p>Images must exist in cypress/fixtures Users must be logged in.</p>	<p>Located in /ProfilePage.js, spec.js</p> <p>Methods: myInfoTabAppear()</p>	Verify the My Info tab loads successfully
2	Verify the 'Required' field error Message is displayed on leaving name field blank	<ol style="list-style-type: none"> Login using valid credentials 2. Click on "My Info" tab 3. Clear the name present in the name inputbar <p>Users must be logged in with Admin privileges.</p>	<p>Reference path e2e/ProfilePage.js:</p> <p>Methods: checkRequiredFieldError()</p>	Required' field error message should be displayed

3	Verify the name gets edited successfully	<ol style="list-style-type: none"> 1. Login using valid credentials 2. Click on "My Info" tab 3. Clear the name present in the name inputbar 4. Enter the new name and hit save button 5. Hit refresh <p>Users must be logged in with Admin access.</p>	<p>Reference path</p> <p>e2e/ProfilePage.js:</p> <p>Methods</p> <p>editProfileName()</p>	Verify the name gets updated in the top right corner
4	Verify the 'My Info' tab's subtabs have unique URL	<ol style="list-style-type: none"> 1. Login using valid credentials 2. Click on "My Info" tab 3. Click on top 3 items from the sidebar('personal details', contact details,emergency contact') <p>User must be logged in and on the home page</p>	<p>Reference path</p> <p>e2e/ProfilePage.js</p> <p>Methods</p> <p>checkPersonalTab() checkContactTab() checkEmergencyTab()</p>	Verify the url from the three is unique
5	Verify new Dependant could be added to the list	<ol style="list-style-type: none"> 1. Login using valid credentials 2. Click on "My Info" tab 3. Navigate to 'Dependents' subtab 4. Fill name ,select 'child' from relat <p>User must be logged in with Admin privileges.</p>	<p>Reference path</p> <p>e2e/ProfilePage.js</p> <p>Methods</p> <p>assignDependentRecord()</p>	Verify the presence of the child in the list

6	Verify new Dependant 'Specify' inputbar only displays when 'Other' option is selected from the relationship dropdown	<ol style="list-style-type: none"> 1. Login using valid credentials 2. Click on "My Info" tab 3. Navigate to 'Dependents' subtab 4. Fill name ,select 'other' from relationship 5. enter the name of other relationship <p>Images must exist in cypress/fixtures User must be logged in</p>	<p>Located in e2e/ProfilePage.js:</p> <p>Methods: checkRelationOtherInputBar()</p>	Verify the specify inputbar is not present before entering the other in relationship inputbar
7	Verify the dependants could be edited from the list	<ol style="list-style-type: none"> 1. Login using valid credentials 2. Click on "My Info" tab 3. Navigate to 'Dependents' subtab 4. Create a new dependant using previous steps 5. click the edit icon and change the name of the dependant <p>At least one user should exist in the user list User must be logged in with Admin privileges.</p>	<p>Reference path e2e/ProfilePage.js:</p> <p>Methods: editDependentRecord();</p>	Verify the dependant details get updated
8	Verify the dependant's could be deleted from the list	<ol style="list-style-type: none"> 1. Login using valid credentials 2. Click on "My Info" tab 3. Navigate to 'Dependents' subtab 4. Create a new dependant using previous steps 5. click the delete icon and verify the candidate gets deleted from the list <p>User must be logged in with Admin access At least two records with different Usernames must exist</p>	<p>Reference path e2e/ProfilePage.js:</p> <p>Methods deleteDependentRecord()</p>	Verify the dependant details get deleted

9	Verify the sample pdf file could be uploaded in the attachments	<ol style="list-style-type: none"> 1. Login using valid credentials 2. Click on "My Info" tab 3. Navigate to 'Dependents' subtab 4. Click on 'Add' Attachments button 5. Browse the pdf from the device 6. Click save button <p>User must be logged in and on the home page</p>	<p>Reference path</p> <p>e2e/ProfilePage.js</p> <p>Methods</p> <p>uploadDependentPDF()</p>	Verify the pdf gets uploaded
10	Verify the Comment inputbar has limit on length	<ol style="list-style-type: none"> 1. Login using valid credentials 2. Click on "My Info" tab 3. Navigate to 'Dependents' subtab 4. Click on 'Add' Attachments button 5. Click Comments input bar and add really long text <p>User must be logged in and on the home page</p>	<p>Reference path</p> <p>e2e/ProfilePage.js</p> <p>Methods</p> <p>checkForCommentLengthError()</p>	verify the 'Should not exceed 200 characters' is displayed

Learners will gain experience in building strongly-typed applications using React.js and managing data flow with **JavaScript**. They'll learn how to define interfaces, use types for error prevention, and improve code maintainability.

With **Cypress**, learners will learn to write and execute automated tests for the <https://opensource-demo.orangehrmlive.com> app.

Key skills include:

- **Browser Automation:** Interacting with web elements and testing multiple browsers.
 - **Assertions & Validations:** Ensuring app behaviour meets expected results.
 - **End-to-End Testing:** Automating real user interactions and validating overall app functionality.
-

IMPLEMENTATION/FUNCTIONAL REQUIREMENT

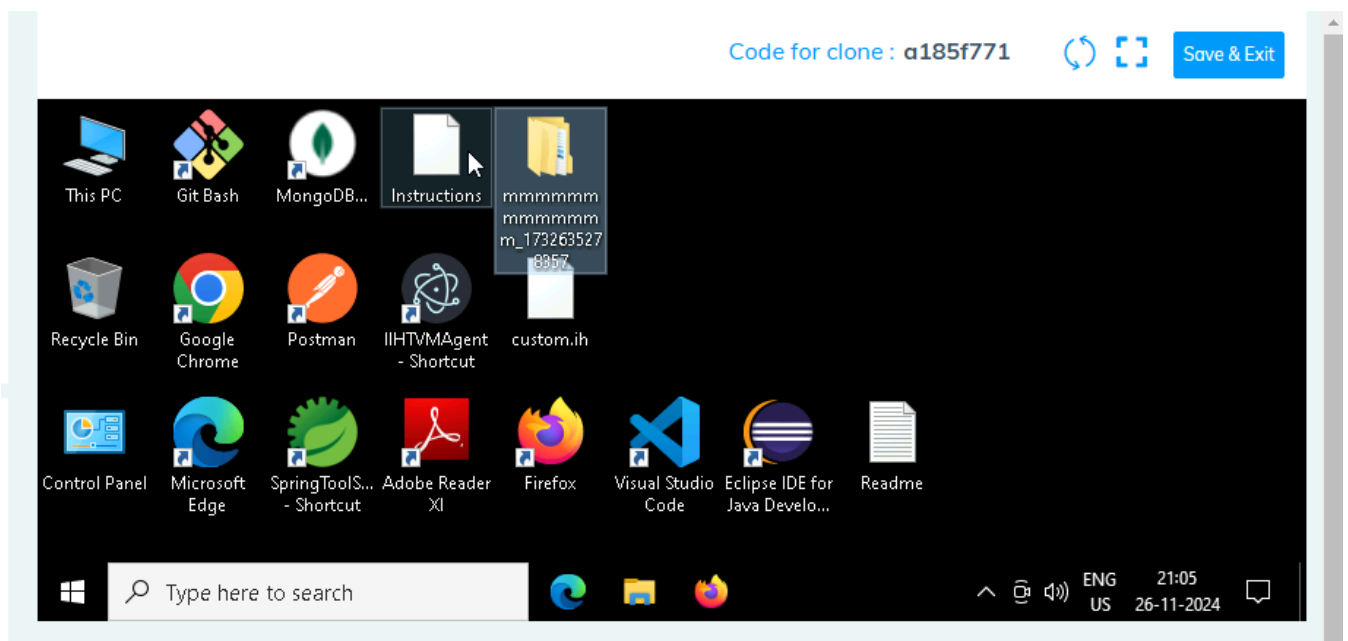
1.1 CODE QUALITY/OPTIMIZATIONS

1. Associates should have written clean code that is readable.
2. Associates need to follow SOLID programming principles.

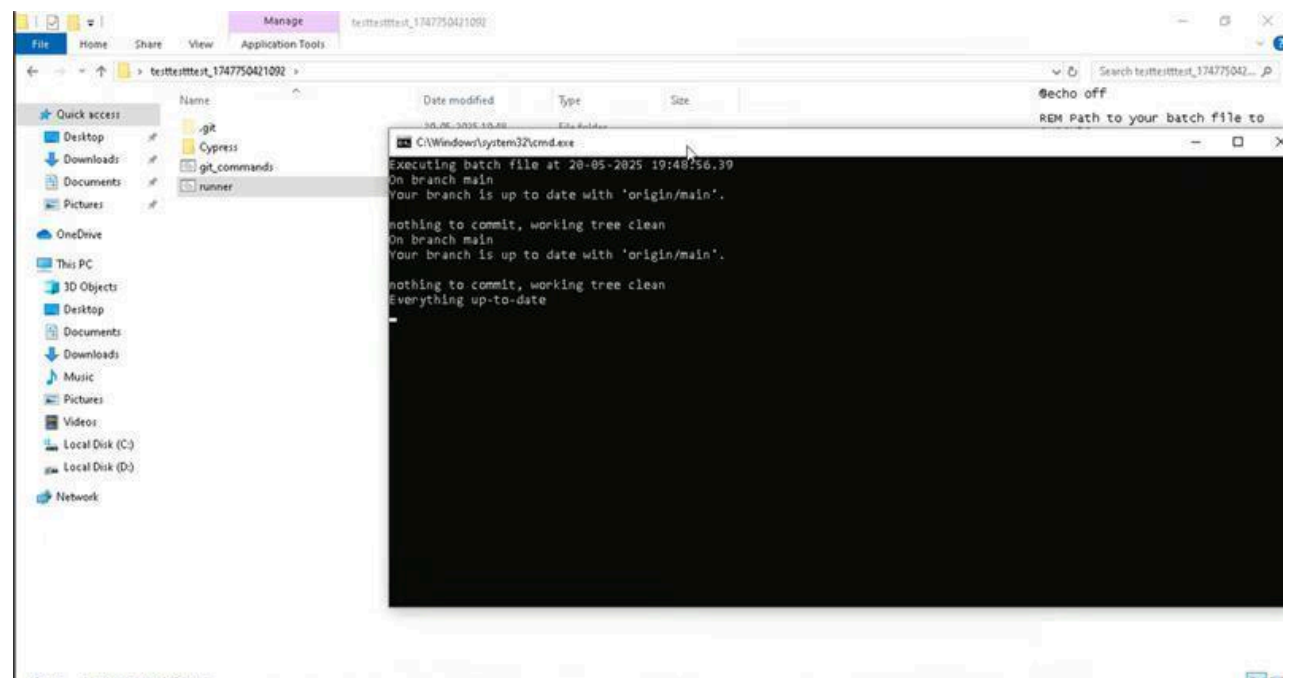
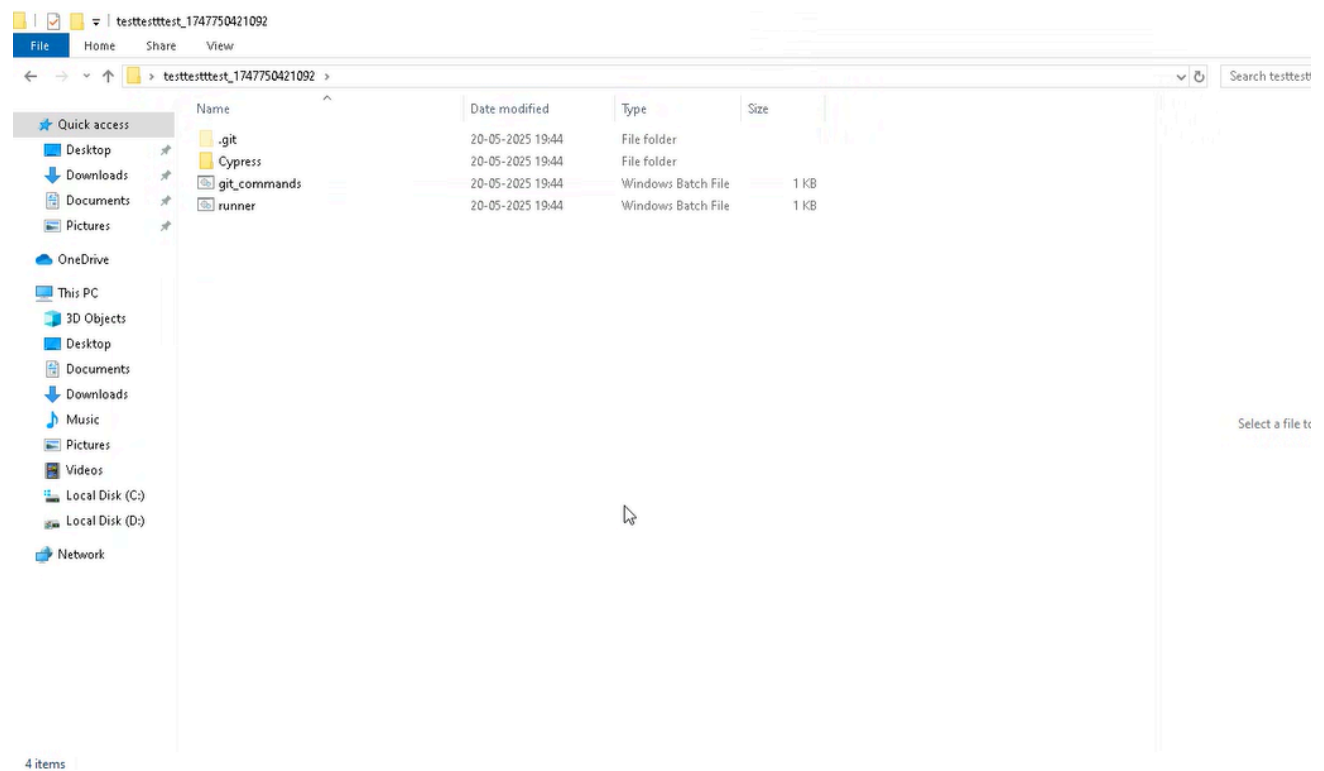
Execution Steps:

Steps for Execution:

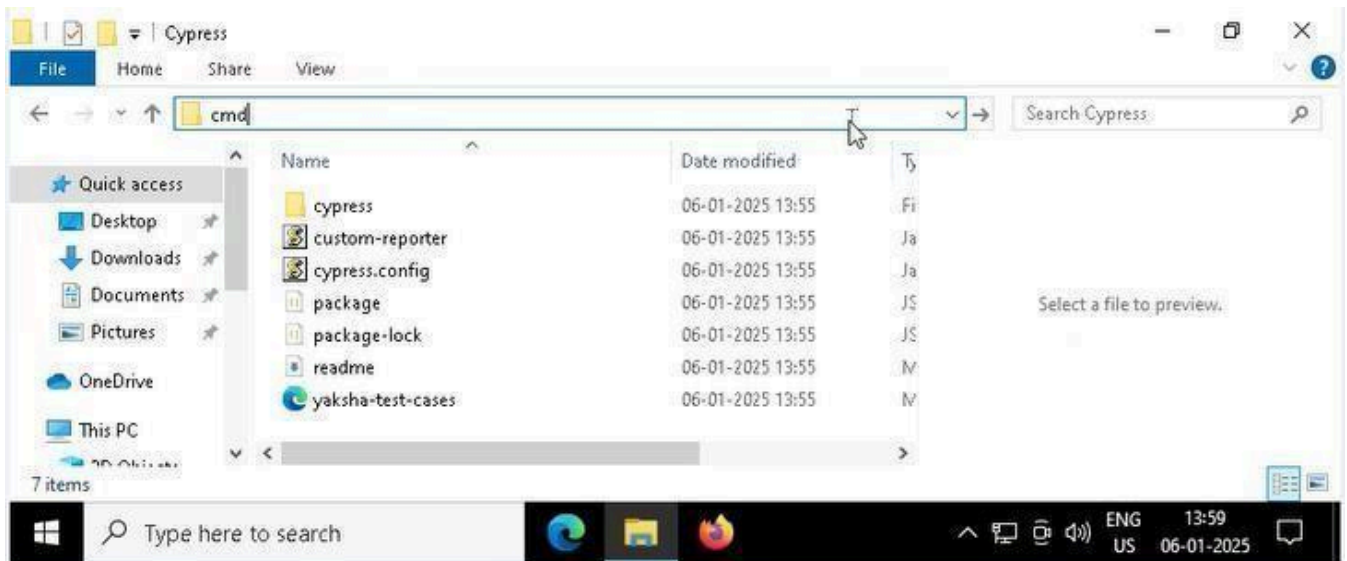
1. Please open the folder created on the desktop with the email name you used to login.

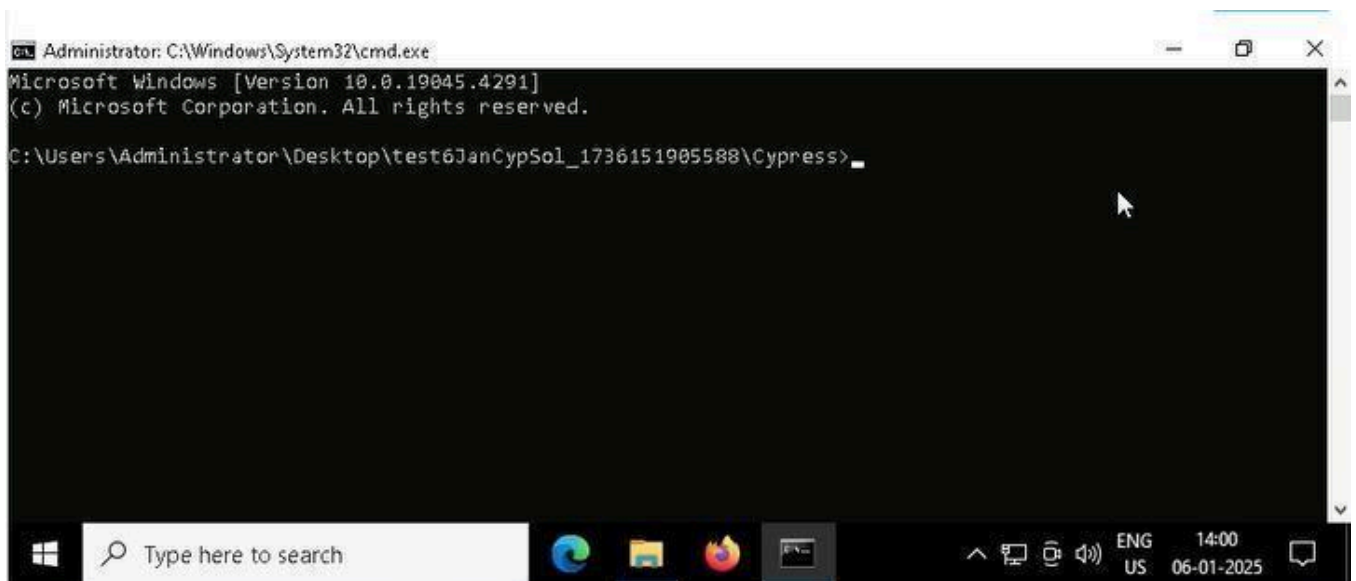


2. Execute this “runner” file. This will keep pushing the code at regular intervals.



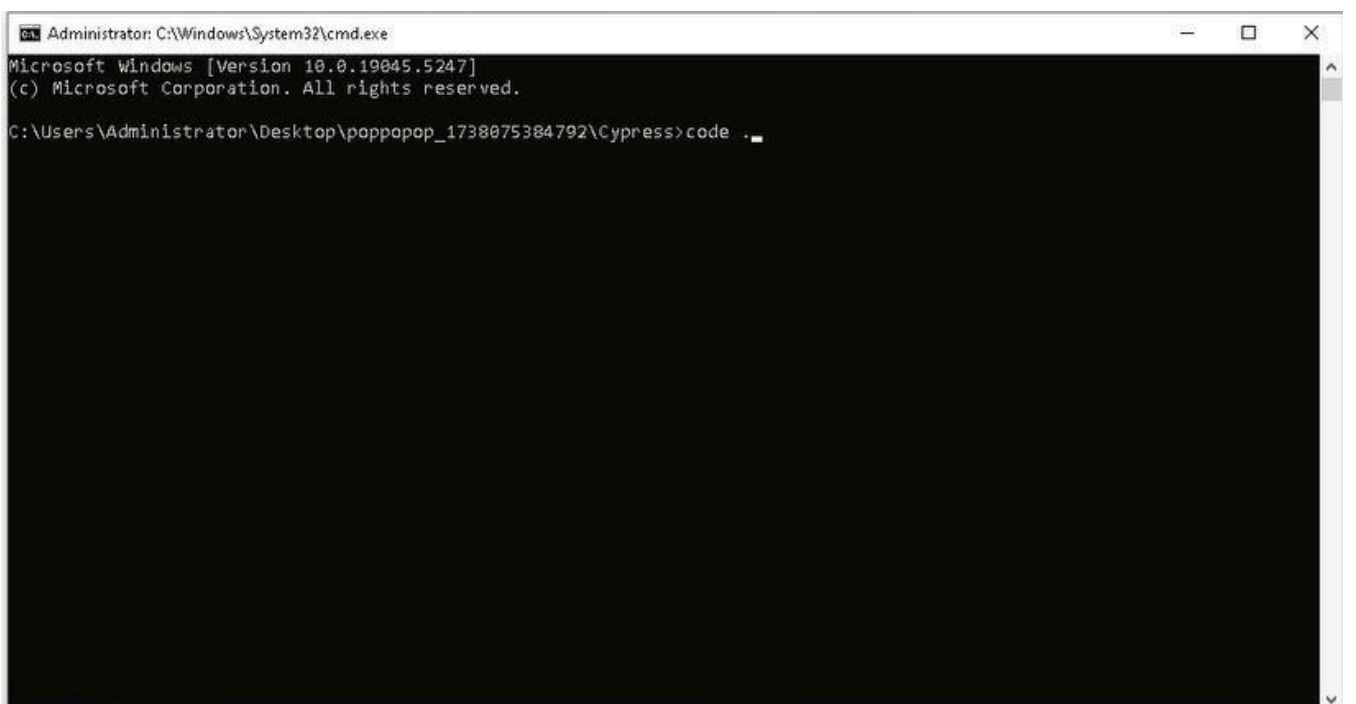
3. Go into the Cypress folder and Open command prompt with its location using “cmd” in search bar



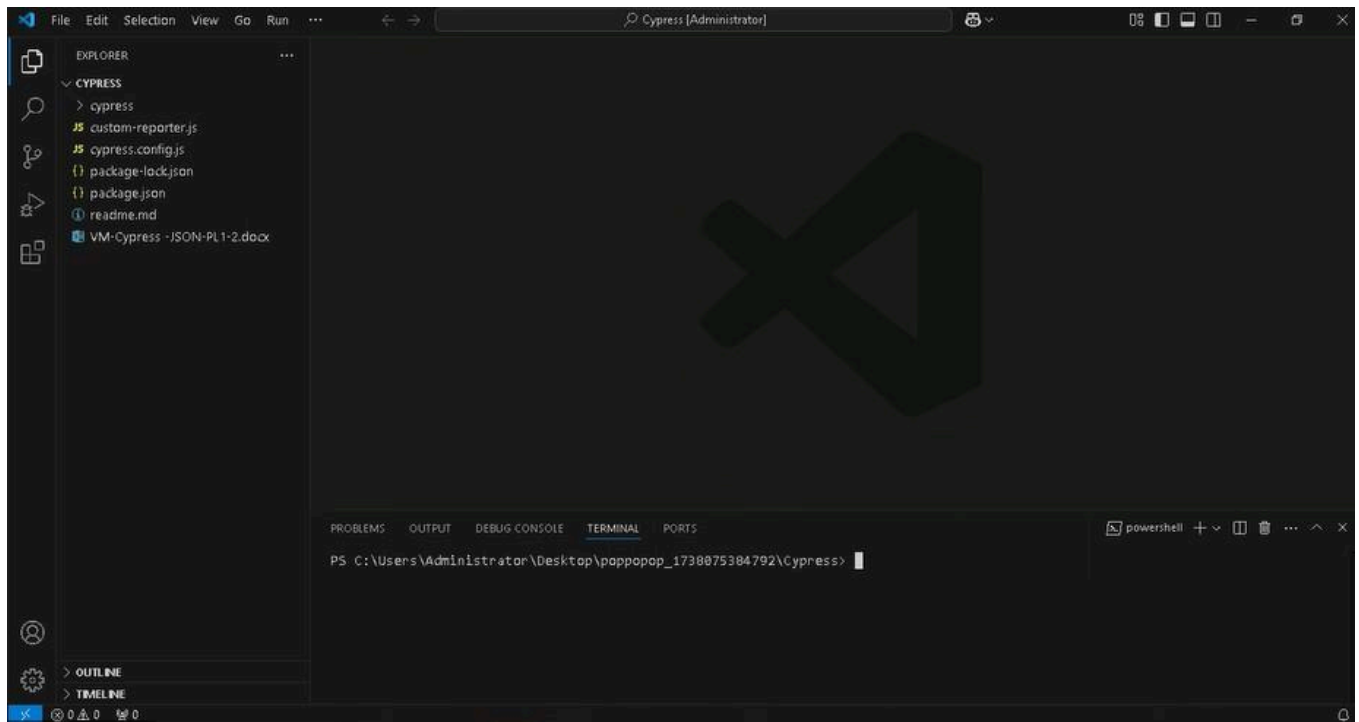


4. Open VS Code through cmd using below

command: code .

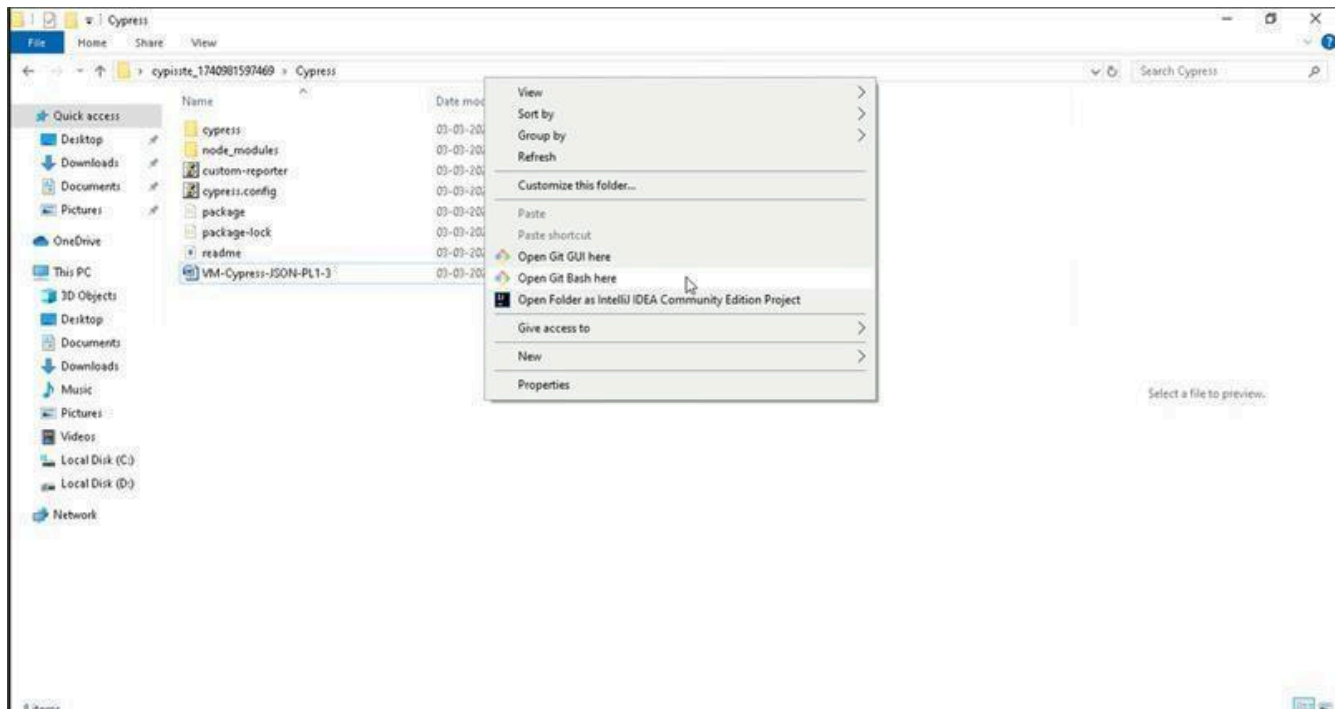


5. Once VsCode is open. Please open the terminal in Cypress folder:



1. Install all dependencies in the Cypress folder
path using: `npm install`
2. Run the following command to open the interactive Cypress Test Runner in the Cypress folder path: `npx cypress open`
3. Run the following command to run test cases in headless mode in the Cypress folder path: `npx cypress run`

If you face any issue in executing above commands i.e npx cypress open and npx cypress run in cmd. Please try to run these in git bash from cypress folder as:



and then execute commands like:

