YAKSHA HEALTH APP WITH JAVASCRIPT AND CYPRESS PL1-5

Mymedic automation using cypress

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 record.

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 few Json files in fixtures folder like LoginData.json.

Path	**File**	**Description**
\fixtures	LoginData.json skyImage.jpg	1. Contains data to read from json file.

PageObjects\Pages	● AdminPage ● BuzzPage ● LoginPage ● ProfilePage	1. All core activities to be performed here. 2. The comments associated with each templated method here describe the expectation. 3. Declare any variable/object you need to share data/status between different methods. Do not modify the signature of methods declared here.

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 Image Could Be Uploaded as Profile Pic	1. Navigate to **My Info** section. 2. Open the profile picture section. 3. Upload image. 4. Click Save **Images must exist in cypress/fixtures** **User must be logged in.**	**Located in** PageObjects/Pages/ProfilePage.js: **Methods:** uploadProfilePicture('yourimage.jpg')	Profile image is successfully updated and visible
2	Verify Admin Can Edit Record	1. Navigate to **Admin Page.** 2.Click **Edit** button on the first listed user. 3. Verify edit form appears. **At least one user should exist in the user list** **User must be logged in with Admin privileges.**	**Reference path** PageObjects/Pages/AdminPage.js: Methods: editUserFormAppear();	Edit user form should appear for the selected record

Test Case No.	Test Case Name	Test Steps to be performed	Path & Method Used	Expected Result

3	Verify Admin Can Sort the Records	1. Navigate to **Admin Page** 2. Click on the **Username** column 3. Validate the usernames are sorted. **User must be logged in with Admin access At least two records with different usernames must exist**	**Reference path** PageObjects/Pages/ AdminPage.js: **Methods** goToAdminAndSortByUsername();	The list of usernames should be sorted in ascending order after clicking the sort button.
4	Verify Upgrade Page Loads Correctly After Clicking the Button	1. Navigate to the **Home Page** 2. Locate and click the **Upgrade** button/link. 3. Validate its href and target attributes using verifyUpgradeLinkAttributes() **User must be logged in and on the home page**	**Reference path** PageObjects/Pages/HomePage.js **Methods** NavigateToAdminPage()	The Upgrade button should: - Have a valid href to upgrade URL - Open in a new tab (target="_blank")
5	Verify on Hover Shows Help Text	1. Navigate to the **Admin Page** 2. Locate the Help icon. 3. Trigger hover 4. Verify title="Help" or tooltip text appears **User must be logged in and on the home page**	**Reference path** PageObjects/Pages/AdminPage.js **Methods** visitAdminForTolltip()	The Help icon should display tooltip or attribute containing the text **"Help"**.

Test Case No.	Test Case Name	Test Steps to be performed	Path & Method Used	Expected Result
6	Verify Admin Can Search Users by Role	1. Navigate to **Admin Page** 2. Open User Role dropdown 3. Select **Admin** 4. Click Search 5. Validate filtered results. **User must be logged in with valid credentials and have access to the Admin Page**	**Reference path** PageObjects/Pages/AdminPage .js **Methods** searchAdminUsersByRole()	The user list should only display users with **Admin** role

7	Verify Image Upload and Share via Drag and Drop	1. Go to **Buzz** tab 2. Click **Share Photo** button 3. Perform drag-and-drop of image 4. Add text to post 5. Click **Share** 6. Verify image is visible in shared post. **User must be logged in and navigated to the Buzz page**	**Reference path** PageObjects/Pages/BuzzPage.js **Methods** postImageWithTimestampText()	Image is successfully uploaded and displayed with post.

8	Verify Like Button is Function-al	1. Navigate to **Buzz** tab
		2. Wait for the first post to load
		3. Capture current like count
		4. Click the heart/like button
		5. Verify like count increased by 1.
		User must be logged in and at least one post must be visible in Buzz.
			Reference path
			PageObjects/Pages/BuzzPage.js:
			Methods :
			getFirstPost();
			getLikeCount();
			clickLikeButton();
				Like count should increment by 1 on clicking the heart Icon.

9	Verify Comment could be added to a post	1. Navigate to **Buzz** tab
		2. Click comment icon on the first post
		3. Type a comment
		4. Press Enter or submit
		5. Verify success toast.
		User must be logged in and at least one post should be present on the Buzz page
			Reference path
			PageObjects/Pages/BuzzPage.js;
			Methods:
			addCommentToFirstPost();
				The comment should be added successfully, and a "Successfully Saved" toast message should appear

Test Case No.	**Test Case Name**	**Test Steps to be performed**	**Path & Method Used**	**Expected Result**
10	Verify Buzz Post Editing Function-ality	1. Navigate to **Buzz** tab 2. Click the **three-dot menu** on the first post 3. Select **Edit Post** 4. Modify the post content 5. Click **Post** to save changes 6. Verify **success toast** appears. **Note: - User must be logged in and must have permission to edit their own Buzz post**	**Reference path** PageObjects/Pages/BuzzPage.js **Methods** editFirstPostWithText()	A success confirmation popup with the message: "**Imaging and lab order add successfully**" should appear.

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://healthapp.yaksha.com

app. Key skills include:

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

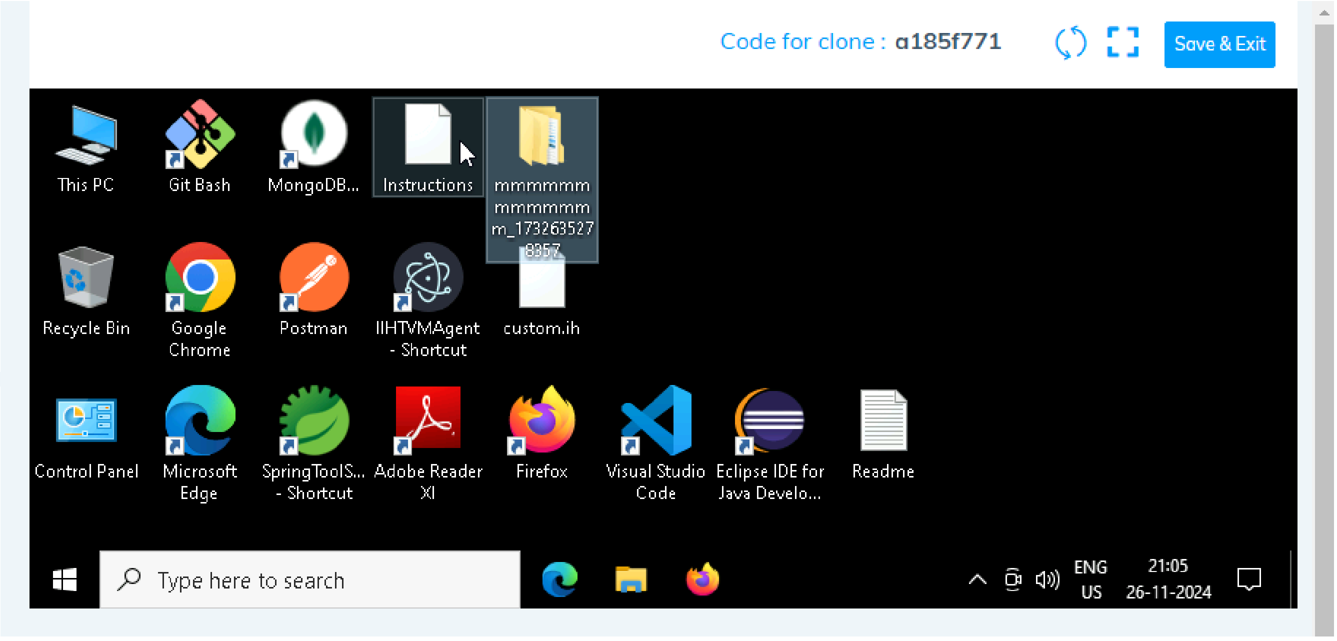
IMPLEMENTATION/FUNCTIONAL REQUIREMENT

1.1 Code Quality/Optimizations

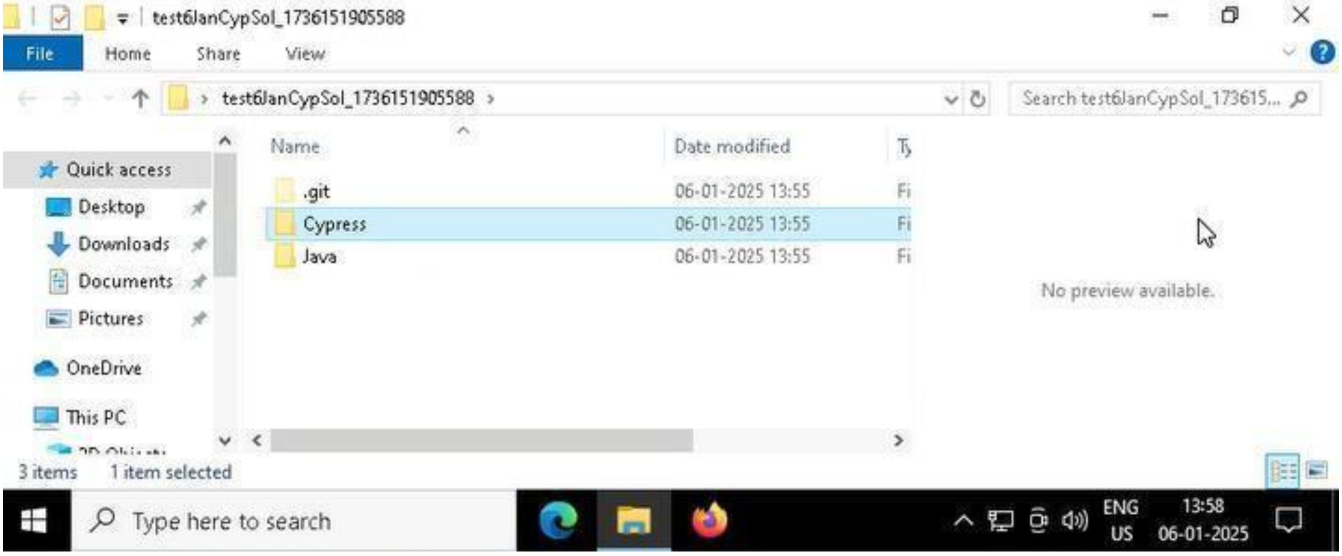
1. Associates should have written clean code that is readable.
2. Associates need to follow SOLID programming principles.

Execution Steps:
Steps for Execution:

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

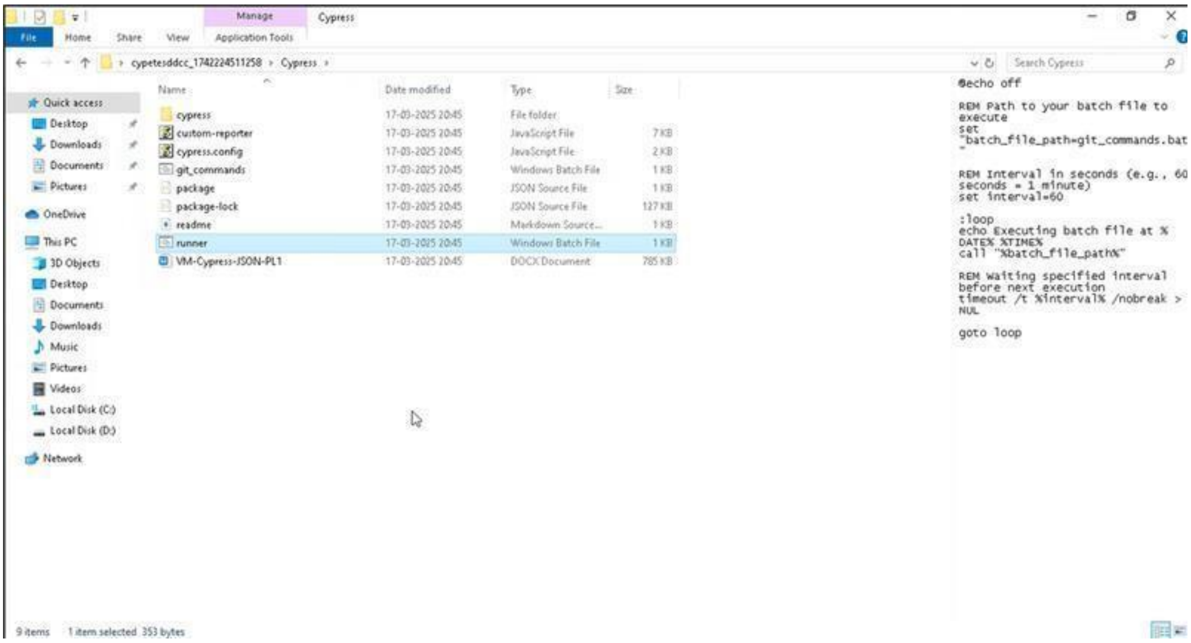


Code for clone : a185f771



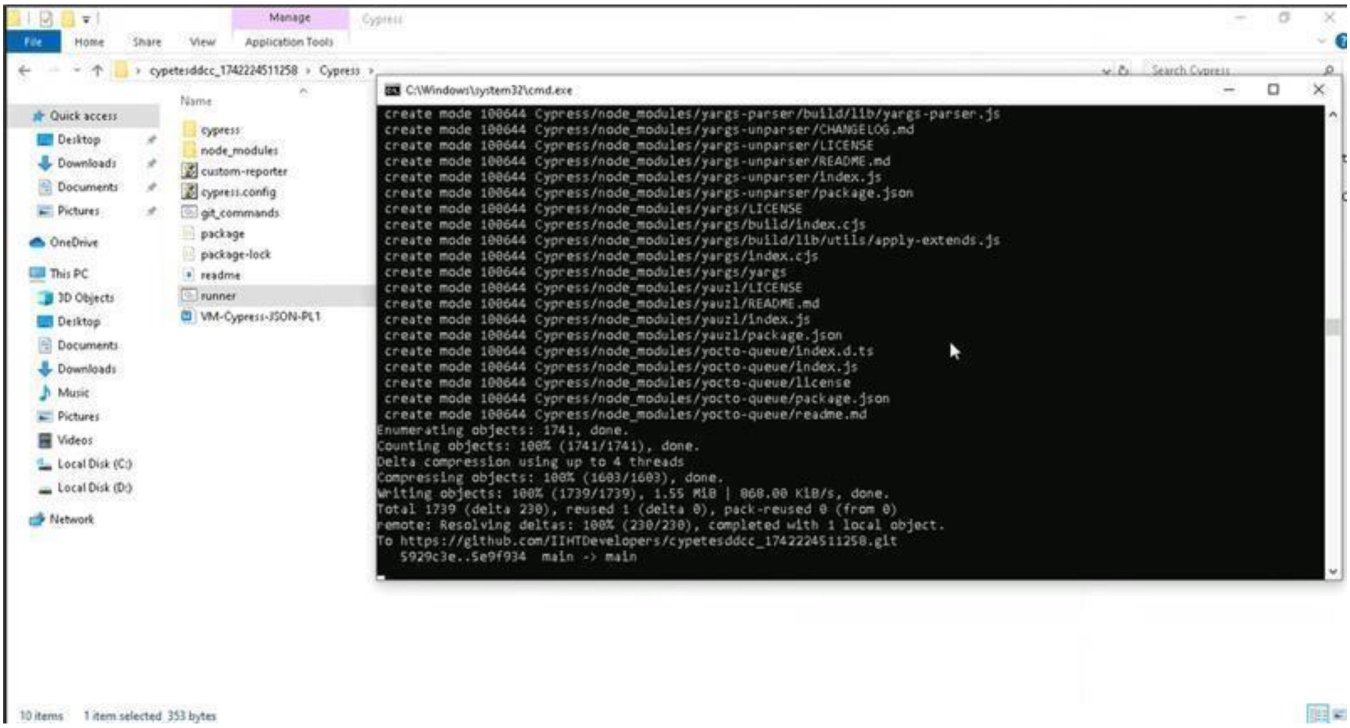
Name	Date modified	Ty
.git	06-01-2025 13:55	Fi
Cypress	06-01-2025 13:55	Fi
Java	06-01-2025 13:55	Fi

2. **Go into the Cypress folder and execute this “runner” file. This will keep pushing the code at regular intervals.**



Name	Date modified	Type	Size
cypress	17-03-2025 20:45	File folder	
custom-reporter	17-03-2025 20:45	JavaScript File	7 KB
cypress.config	17-03-2025 20:45	JavaScript File	2 KB
git_commands	17-03-2025 20:45	Windows Batch File	1 KB
package	17-03-2025 20:45	JSON Source File	1 KB
package-lock	17-03-2025 20:45	JSON Source File	127 KB
readme	17-03-2025 20:45	Markdown Source...	1 KB
runner	17-03-2025 20:45	Windows Batch File	1 KB
VM-Cypress-JSON-PL1	17-03-2025 20:45	DOCX Document	785 KB

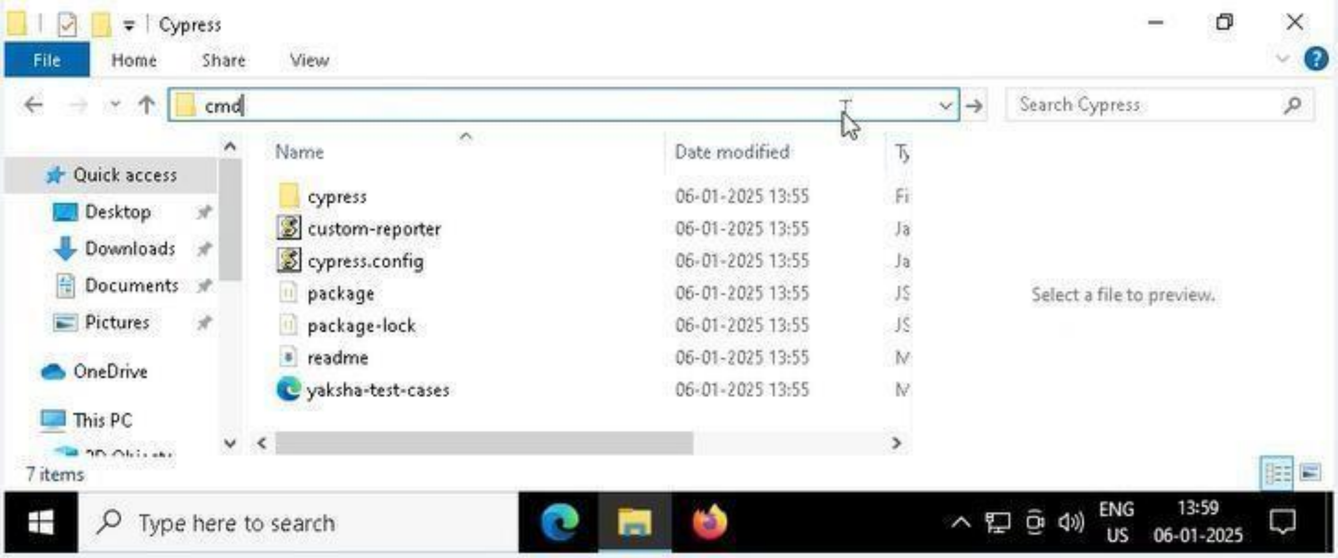
```
@echo off

REM Path to your batch file to
execute
set
"batch_file_path=git_commands.bat
"

REM Interval in seconds (e.g., 60
seconds = 1 minute)
set interval=60

:loop
echo Executing batch file at %
DATE% %TIME%
call "%batch_file_path%"

REM Waiting specified interval
before next execution
timeout /t %interval% /nobreak >
NUL

goto loop
```



C:\Windows\system32\cmd.exe

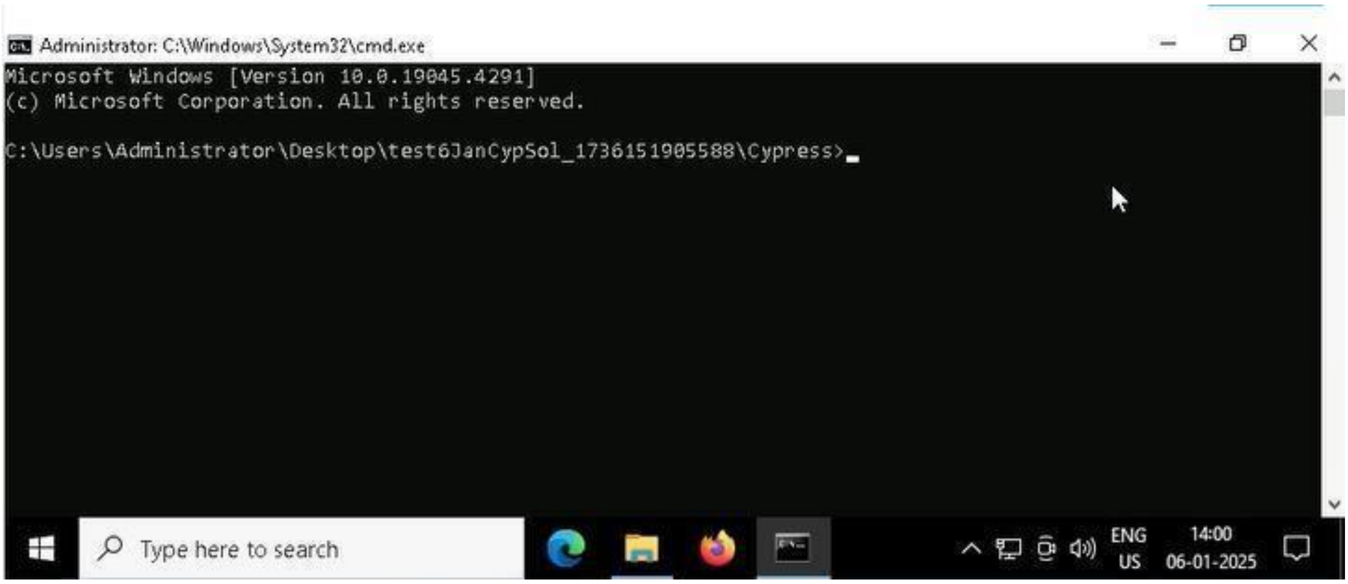
```
 create mode 100644 Cypress/node_modules/yargs-parser/build/lib/yargs-parser.js
 create mode 100644 Cypress/node_modules/yargs-unparser/CHANGELOG.md
 create mode 100644 Cypress/node_modules/yargs-unparser/LICENSE
 create mode 100644 Cypress/node_modules/yargs-unparser/README.md
 create mode 100644 Cypress/node_modules/yargs-unparser/index.js
 create mode 100644 Cypress/node_modules/yargs-unparser/package.json
 create mode 100644 Cypress/node_modules/yargs/LICENSE
 create mode 100644 Cypress/node_modules/yargs/build/index.cjs
 create mode 100644 Cypress/node_modules/yargs/build/lib/utils/apply-extends.js
 create mode 100644 Cypress/node_modules/yargs/index.cjs
 create mode 100644 Cypress/node_modules/yargs/yargs
 create mode 100644 Cypress/node_modules/yauzl/LICENSE
 create mode 100644 Cypress/node_modules/yauzl/README.md
 create mode 100644 Cypress/node_modules/yauzl/index.js
 create mode 100644 Cypress/node_modules/yauzl/package.json
 create mode 100644 Cypress/node_modules/yocto-queue/index.d.ts
 create mode 100644 Cypress/node_modules/yocto-queue/index.js
 create mode 100644 Cypress/node_modules/yocto-queue/license
 create mode 100644 Cypress/node_modules/yocto-queue/package.json
 create mode 100644 Cypress/node_modules/yocto-queue/readme.md
Enumerating objects: 1741, done.
Counting objects: 100% (1741/1741), done.
Delta compression using up to 4 threads
Compressing objects: 100% (1603/1603), done.
Writing objects: 100% (1739/1739), 1.55 MiB | 868.00 KiB/s, done.
Total 1739 (delta 230), reused 1 (delta 0), pack-reused 0 (from 0)
remote: Resolving deltas: 100% (230/230), completed with 1 local object.
To https://github.com/IIHTDevelopers/cypetesddcc_1742224511258.git
   5929c3e..5e9f934  main -> main
```

3. Open command prompt with its location using “cmd” in search bar



cmd

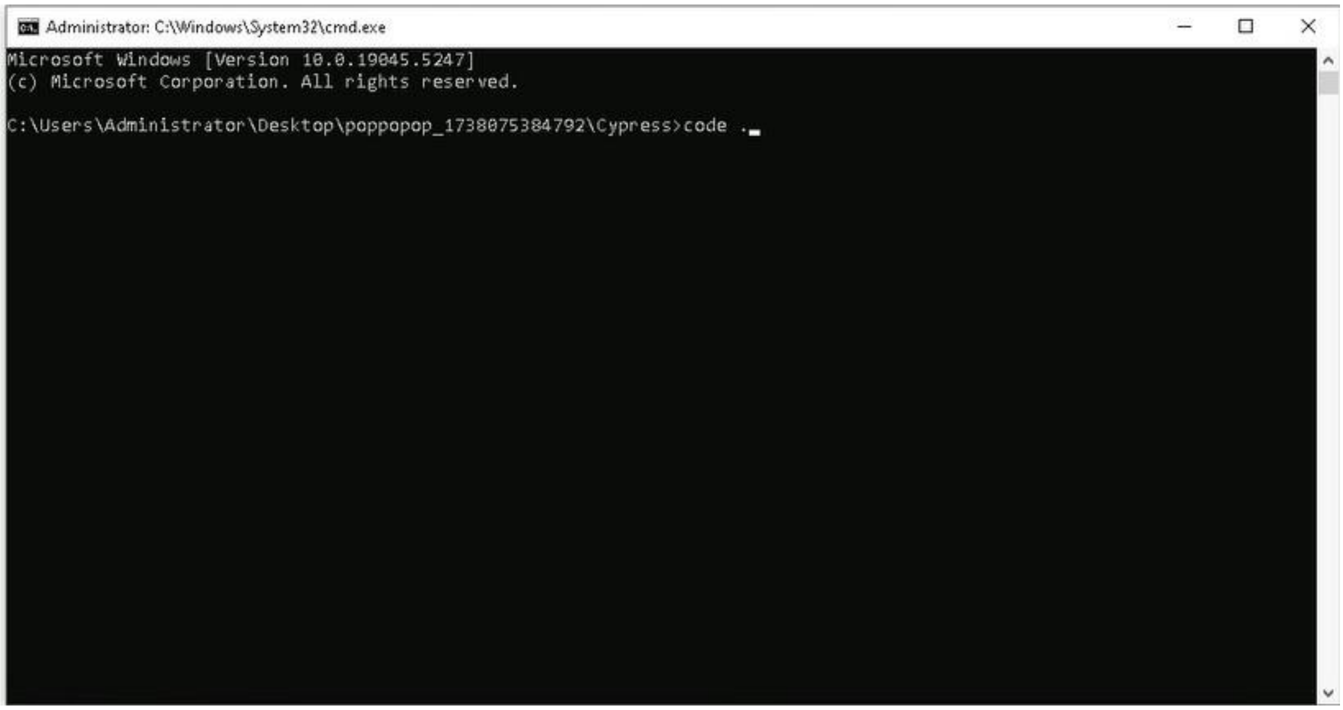
Name	Date modified	Ty
cypress	06-01-2025 13:55	Fi
custom-reporter	06-01-2025 13:55	Ja
cypress.config	06-01-2025 13:55	Ja
package	06-01-2025 13:55	JS
package-lock	06-01-2025 13:55	JS
readme	06-01-2025 13:55	M
yaksha-test-cases	06-01-2025 13:55	M



```
Microsoft Windows [Version 10.0.19045.4291]
(c) Microsoft Corporation. All rights reserved.

C:\Users\Administrator\Desktop\test6JanCypSol_1736151905588\Cypress>
```

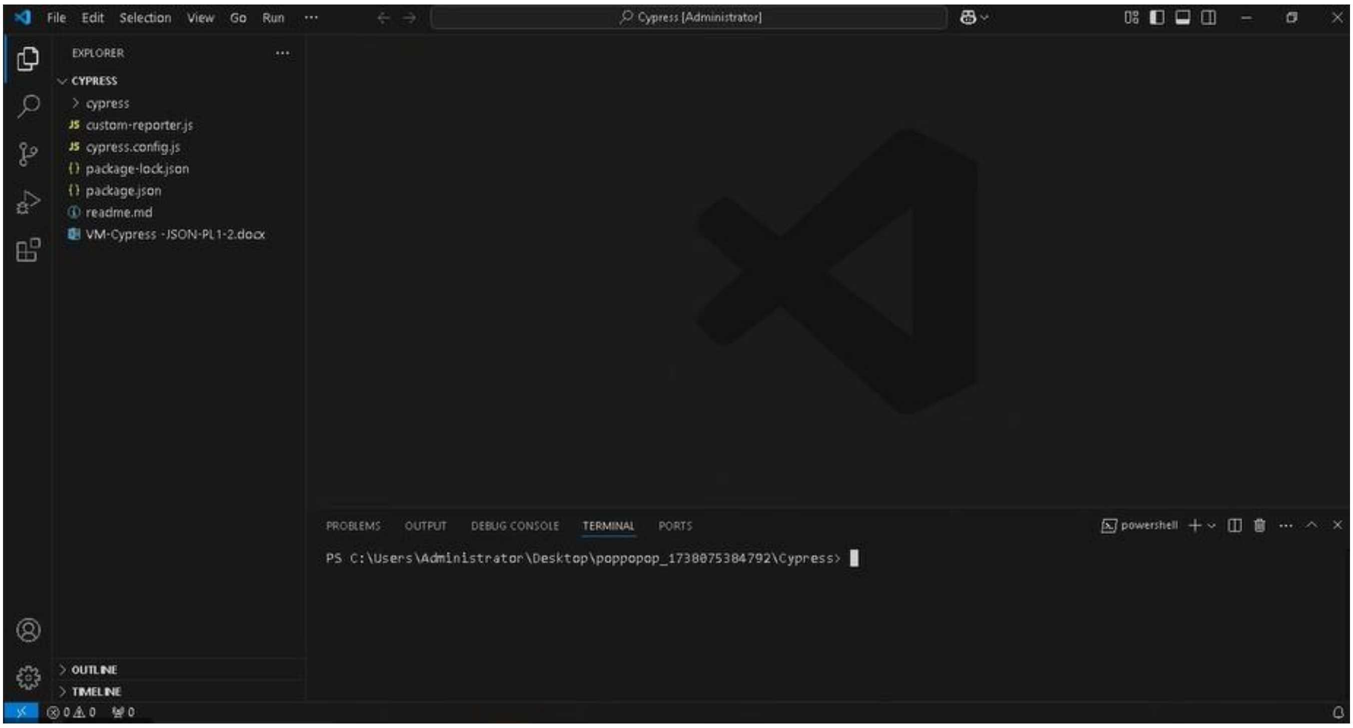
4. **Open VS Code through cmd using below command:**

   code .



```
Microsoft Windows [Version 10.0.19045.5247]
(c) Microsoft Corporation. All rights reserved.

C:\Users\Administrator\Desktop\poppopop_1738075384792\Cypress>code .
```

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



```
PS C:\Users\Administrator\Desktop\poppopop_1738075384792\Cypress>
```

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:**

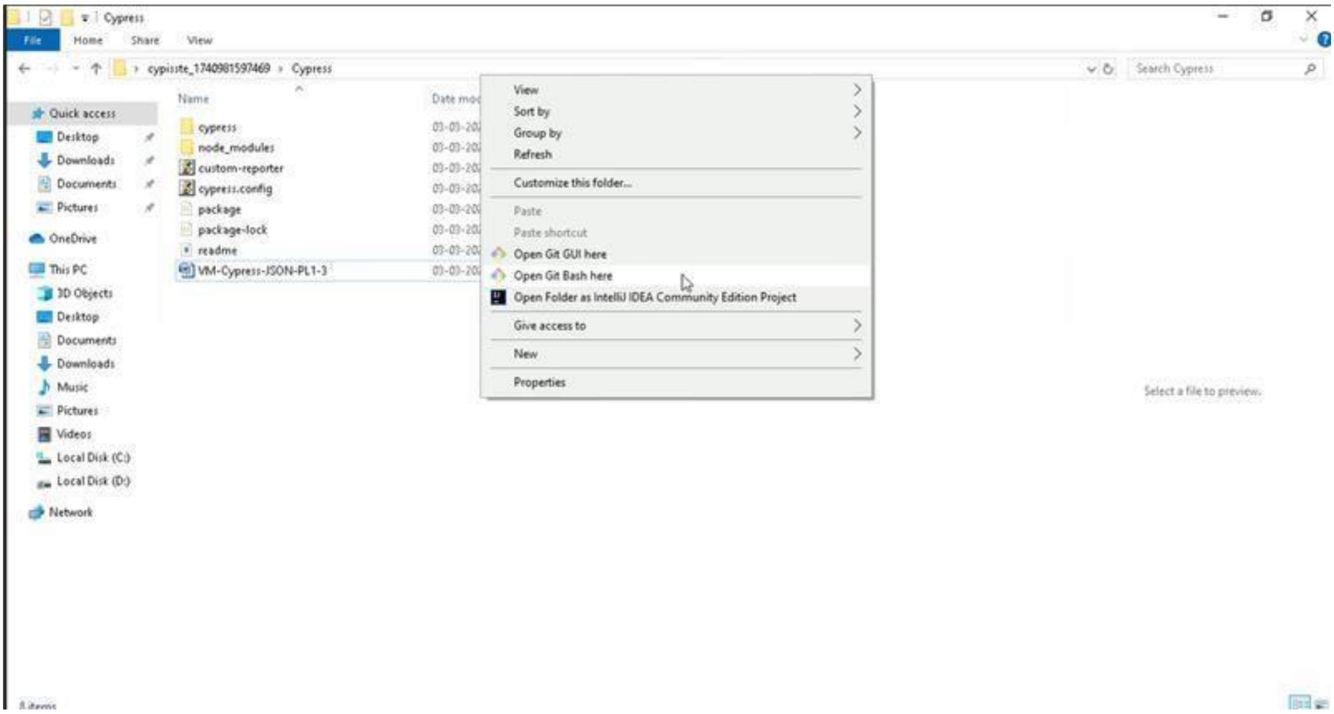
   npx cypress open

3. **Run the following command to run test cases in headless mode in the Cypress folder path:**

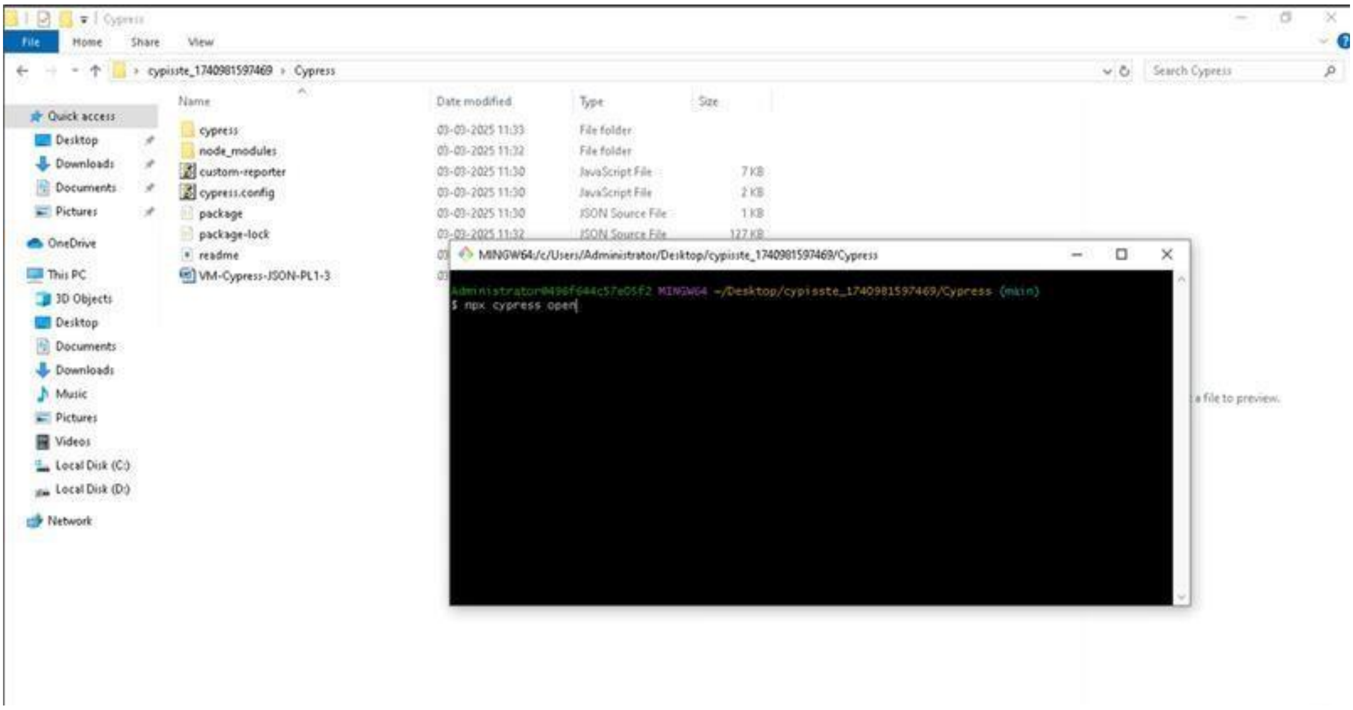
   npx cypress run

****NOTE: - It is a must to execute npx cypress run, this runs test cases in headless mode, shows summary in the end and submits the test cases on server.**

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:



Name	Date modified	Type	Size
cypress	03-03-2025 11:33	File folder	
node_modules	03-03-2025 11:32	File folder	
custom-reporter	03-03-2025 11:30	JavaScript File	7 KB
cypress.config	03-03-2025 11:30	JavaScript File	2 KB
package	03-03-2025 11:30	JSON Source File	1 KB
package-lock	03-03-2025 11:32	JSON Source File	127 KB
readme	03		
VM-Cypress-JSON-PL1-3	03		

MINGW64:/c/Users/Administrator/Desktop/cypisste_1740981597469/Cypress

```
Administrator@496f644c57e05f2 MINGW64 ~/Desktop/cypisste_1740981597469/Cypress (main)
$ npx cypress open
```