**Objective:**

This workshop is intended for sharing fundamental aspects of E2E Testing using Cypress tool. This workshop will give hands on experience to the participants to start implementing Cypress. Some basic to intermediate topics of Cypress will be covered in this workshop.

At the end of the workshop, a sample reference project will be shared with the audience.

**Duration: 120 mins**

**Target Audience:** Anyone having knowledge of Programming with no prior knowledge of Cypress.

**Takeaway:** Hands-on knowledge of Cypress

**Cypress Workshop**

1. Introduction to Cypress (T) (5 mins)

* What is Cypress
* Architecture
* Benefits
* Browsers supported as of now

1. Installation of Cypress (T) (5 mins)

* Command to install Cypress
* Folder structure
* Cypress.json file (Config file)

1. Getting started with Cypress(T+H) (25 mins)

* Mocha Framework
* Cypress commands
* Locators (css-selectors)
* Assertions(Chai and Built In)
* Test Execution (Command Line)
* Configuring scripts via package.json

1. Hands On (15 mins)
2. Q&A (10 mins)
3. Break (5 mins)
4. Fixtures(T) (2 mins)
5. Page Object Model (T) (18 mins)
6. Hands On (15 mins)
7. Advanced Topics (T) (10 mins)

* Mochawesome Reports
* Adding plugins
* CI-CD (Jenkins)
* Promise Handling (its JS feature)
* API Testing (basic)

1. Challenges
2. Q&A session (5 mins)

**Prerequisite**

**Installation for MAC machines**

1. **Install Node.js**

* Visit <https://nodejs.org/en/download/> and install node.js. Click on Mac OS installer.pkg(64 bit)
* In case node.js is already installed upgrade nodeJs to >8

Graphical user interface

Description automatically generated

* Verify if node is properly installed by going to Iterm/Terminal and type

node -v

You must be able to see installed version of node

* Download Chrome and Edge if you don’t have

Chrome= <https://www.google.com/chrome/>

Edge= <https://www.microsoft.com/en-us/edge>

1. **Create Workspace**   
   Create an empty folder(this will work as your workspace) (eg Users/mukulmaheshwari/Documents/CypressAutomation)

Graphical user interface, application

Description automatically generated

1. **Install IDE**

<https://code.visualstudio.com/download> and Open it

1. Open the folder(CypressAutomation) you created in step 2 from your Editor(VSCode)

Graphical user interface, text

Description automatically generated

A screenshot of a computer

Description automatically generated

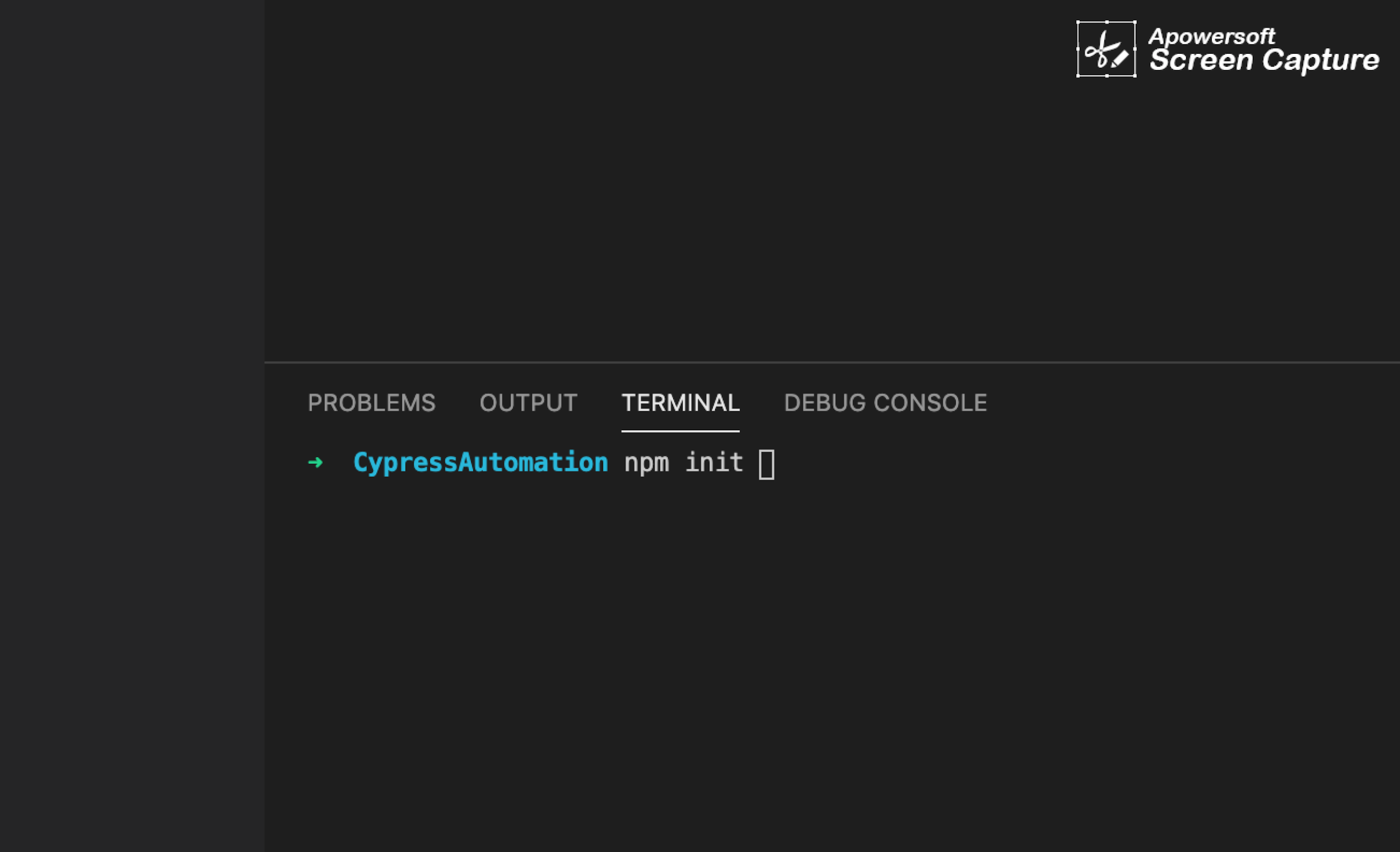
1. Generate package.json file for your project

* package.json is just like your pom.xml file in selenium which contains all the libraries we will be using
* In your editor open terminal and type

npm init

Graphical user interface, application, website

Description automatically generated



Keep pressing enter till it installs

Expand your folder CypressAutomation to see if package.json has appeared

Graphical user interface, text

Description automatically generated

1. **Install Cypress**It will install cypress and will also save it as dev dependency in your package.json  
   Now in your terminal type   
   npm install cypress - -save-dev

Make sure you get a node\_modules folder inside your workspace

Graphical user interface, text

Description automatically generated

1. **Open Cypress**   
   Type  
   npx cypress open

in your IDE’s terminal   
Graphical user interface, application

Description automatically generated

Click OK and close the window

And installation is Done!

Text

Description automatically generated

Make sure you get the folder structure as in above screenshot inside the cypress folder – Fixtures, Integration, plugins, support

1. Write a small test and verify if everything is working fine

Create a folder inside integration folder and name it as ‘tests’

Graphical user interface, text

Description automatically generated

* Now create a file inside the folder, name the file as ‘google.js’   
  Paste this code inside the file google.js   
  ///<reference types="Cypress"/>

describe("Verify google", function () {

it("Verify URL of google", function () {

cy.visit("https://www.google.com/")

cy.url().should("contain", "google")

})

})  
And execute the code by typing below command in terminal   
npx cypress open   
Text

Description automatically generatedOn the interface click on google.js   
 Graphical user interface, text, application, email

Description automatically generated

1. Make sure your code executes successfully   
   Graphical user interface, application

   Description automatically generated