# 

# Beforeeach will be executed before every IT block…

# Inspect for Table –

# Do CNTRL F and search as table in console OR

# Use css selector to locate table

# 

# Under table, u can see tbody – that contains all rows in table & thead contain all columns data

# table[class=’table table-bordered table-hover’]>tbody --- to select tbody

# 

# table[class=’table table-bordered table-hover’]>tbody>tr will shows all rows in table 10 rows

# 

# 

# Now, under table, u can see thead – that contains all collumns in table

# table[class=’table table-bordered table-hover’]>thead>tr>td --- to select thead (for all columns)

# 

# 

# In order to get particular element data from table, Identify the row number and column # data u want to identify..

# Below we are looking for 5th row and 3rd column data.. in example

# table[class=’table table-bordered table-hover’]>tbody>tr:nth-child(5)>td:nth-child(3)

# 

# 

# above we are looking for 5th row and 3rd column data.. in example

# table[class=’table table-bordered table-hover’]>tbody>tr:nth-child(5)>td:nth-child(3)

# tbody – contains row data

# thead – contains column data

# 

# 

# $row – gets row data

# Index -> starts with zero

# $rows -> gets all row data

# $col - gets colmn data current

# Index – starts with zero

# $cols – gets all column data

# Output 🡪 logs all data 🡪 Above code can be used to get all DATA if PAGINATION is not present

# PAGINATION --🡪

# Capture total number of pages first…

# 

# 

# 

# Below code is to look upto 5 pages, read email address for all rows upto 5th page

# 

# PAGE13 -🡪MOUSE EVENTS Cypress Trigger we use as per documentation

# Use css inspector r.click and get OR

# Use cupress open and use locate inspector and get locator OR use inspect and find locators..

# 

# 

# Need to use trigger mouseover as first step to locate element under that…

# So, before trigger is not visible and after trigger is visible…

# Right-Click use trigger context menu

# 

# OR

# 

# Double-Click:

# 

# 

# 

# Drag and Drop : install plugin npm install –save-dev @4tw/cypresss-drag-drop

# 

# Add below line on top of Java script:

# 

# 

# 

# get(drap locator).drag(drop locator)…

# Sometimes, eklement may be hidden and may not work will throw some error..

# So, in that case use brute force

# 

# SCROLL to VIEW:

# 

# 

# We can use {duration:2000} milliseconds to view

# 

# If yu want to get footer element on page (end of page) search for tag footer in console body and get locator..

# Below is footer view..

# 

# PART 14: 🡪 FILE UPLOAD

# Files will be read only from Fixtures folder in Cypress..

# 

# 

# Add this on top of script..

# 

# 

# 

# Rename a file while it is uploading --🡪

# 

# File Upload Drag and Drop -🡪

# 

# Drop file

# 

# MULTIPLE FILE UPLOAD:

# 

# 

# Some elements are inside shadow DOM (Document object model) inside that there will be another DOM..called sub DOM document object model..

# 

# There is something calledshadow root under that we have this element..

# 

# 

# Above both locators browse button and also after attach test.pdf message is also under shadow DOM..

# PART 15 – HOOKS and TAGS

# HOOKS-🡪

# Before - runs once before all tests starts execution

# After - runs once after all tests completed bexecution

# Beforeeach – runs before every IT block

# Aftereach – runs after every IT block

# TAGS-🡪 it.Only and it.skip

# 

# PART 16 🡪 FIXTURES and DATA DRIVEN TEST:

# 

# File extension .json is not mandatory is optional to pass.. filename is important

# 

# To re-use same data file for Multiple IT blocks, we can use through HOOKS and store fixture data in variable and reuse..

# Create BEFORE hook and inside it use fixture and capture data..store into variable..

# 

# DATA DRIVEN TEST-🡪 Multiple sets of data

# 

# 

# We have written if condition here, if valid data used…it is valid credentials..else, invalid

# 

# PART 17 🡪 CUSTOM COMMANDS

# 

# 

# 

# LOGIN PAGE create custom command:

# 

# 

# Click link we already created earlier, just use label to locate LOGIN link

# OVERWRITE EXISTING CUSTOM COMMAND-🡪 To Overwrite contains command we have created for CLICKLINK custom

# To accept all case sensitive labels as well…

# 

# 

# FORWARD and BACK 🡪

# 

# 

# Screenshot and video--🡪

# 

# Creates screenshot under screenshot/video folder…with file name as specified

# To Automatically capture failures screenshot and videos, We need to run command thru command line

# Ex: npx cypress run –spec cypress/e2e/filename.cy.js

# 

# Headless mode… below is screenshot and video of failed assertion..

# 

# PART 20---🡪 GENERATE REPORT HTML -🡪 using mochaawesome

# Step1: run command

# 

# Step2: go to cypress.config.js and add statements as

# 

# Save the file..

# Step4 : go to support and e2e.js and add below statement

# 

# Save file..

# Run testes through command line

# 

# Will create folder as REPORTS and create file..

# 

# r.click on html file, copy path and open on browser….

# 

# By Default executes on ELECTRON browser in headless mode…

# You can run by specifying browser u want---

# 

# PART 21 🡪 PAJE OBJECT MODEL-🡪

# Create folder under cypress – pageobjects

# 

# Create class for login page as loginPage.js

# Create tetslogin.cy.js and write test spec..

# 

# 

# Create object to use its methods and use import to call class where it resides…

# If want to use fixtures to pass data..

# 

# 