POM Banking Project by Pavan

Step 1. Create a maven project

Step 2. Add dependencies:

selenium-java- 3.141.59,

testNG- 6.14.3,

WebDriverManager- Latest version always,

apache poi, poi-ooxml, **apache poi 3.17 version**

Apache Ooxml-schemas, apache xmlBeans, apache xml commons collections, log4j, extent reports.

Step 2. Create packages:

pageObjects,

testCases,

utilities,

testData,

Step 3. Create Folders:

Reports,

screenShots,

xmlFiles,

reports: [ inside extent-config.xml file ]

[ listener class (reporting.java)] : source > Overrite/Implement methods > select methods as needed.

configurationFiles,

Step 4. create log4j file under project: right click on project > new > file(log4j.properties) > copy log4j file from website then customize > paste to the file.

Step 5.

**How to run only failed test cases ?**

: first fix the test cases then run them… follow bellow.

: test-output > click on testng-failed.xml > rightclick > run as > TestNG Suite.

-------------------------------------------------------------------------------------------------------------------------

**package** com.inetbanking.utilities;

**public** **class** ReadConfig {

Properties pro;

**public** ReadConfig()

{

File src = **new** File("./Configuration/config.properties");

**try** {

FileInputStream fis = **new** FileInputStream(src);

pro = **new** Properties();

pro.load(fis);

} **catch** (Exception e) {

System.***out***.println("Exception is " + e.getMessage());

}

}

**public** String getApplicationURL()

{

String url=pro.getProperty("baseURL");

**return** url;

}

**public** String getUsername()

{

String username=pro.getProperty("username");

**return** username;

}

**public** String getPassword()

{

String password=pro.getProperty("password");

**return** password;

}

**public** String getChromePath()

{

String chromepath=pro.getProperty("chromepath");

**return** chromepath;

}

**public** String getIEPath()

{

String iepath=pro.getProperty("iepath");

**return** iepath;

}

**public** String getFirefoxPath()

{

String firefoxpath=pro.getProperty("firefoxpath");

**return** firefoxpath;

}

}

-------------------------------------------------------------------------------------------------------------------------

**Extent Reports.....**

**package** com.inetbanking.utilities;

**public** **class** Reporting **extends** TestListenerAdapter

{

**public** ExtentHtmlReporter htmlReporter;

**public** ExtentReports extent;

**public** ExtentTest logger;

**public** **void** onStart(ITestContext testContext)

{

String timeStamp = **new** SimpleDateFormat("yyyy.MM.dd.HH.mm.ss").format(**new** Date());//time stamp

String repName="Test-Report-"+timeStamp+".html";

htmlReporter=**new** ExtentHtmlReporter(System.*getProperty*("user.dir")+ "/test-output/"+repName);//specify location of the report

htmlReporter.loadXMLConfig(System.*getProperty*("user.dir")+ "/extent-config.xml");

extent=**new** ExtentReports();

extent.attachReporter(htmlReporter);

extent.setSystemInfo("Host name","localhost");

extent.setSystemInfo("Environemnt","QA");

extent.setSystemInfo("user","pavan");

htmlReporter.config().setDocumentTitle("InetBanking Test Project"); // Tile of report

htmlReporter.config().setReportName("Functional Test Automation Report"); // name of the report

htmlReporter.config().setTestViewChartLocation(ChartLocation.***TOP***); //location of the chart

htmlReporter.config().setTheme(Theme.***DARK***);

}

**public** **void** onTestSuccess(ITestResult tr)

{

logger=extent.createTest(tr.getName()); // create new entry in th report

logger.log(Status.***PASS***,MarkupHelper.*createLabel*(tr.getName(),ExtentColor.***GREEN***)); // send the passed information to the report with GREEN color highlighted

}

**public** **void** onTestFailure(ITestResult tr)

{

logger=extent.createTest(tr.getName()); // create new entry in th report

logger.log(Status.***FAIL***,MarkupHelper.*createLabel*(tr.getName(),ExtentColor.***RED***)); // send the passed information to the report with GREEN color highlighted

String screenshotPath=System.*getProperty*("user.dir")+"\\Screenshots\\"+tr.getName()+".png";

File f = **new** File(screenshotPath);

**if**(f.exists())

{

**try** {

logger.fail("Screenshot is below:" + logger.addScreenCaptureFromPath(screenshotPath));

}

**catch** (IOException e)

{

e.printStackTrace();

}

}

}

**public** **void** onTestSkipped(ITestResult tr)

{

logger=extent.createTest(tr.getName()); // create new entry in th report

logger.log(Status.***SKIP***,MarkupHelper.*createLabel*(tr.getName(),ExtentColor.***ORANGE***));

}

**public** **void** onFinish(ITestContext testContext)

{

extent.flush();

}

}

**Updated Extent reports..**

**public** **class** BaseClass120122 {

**public** WebDriver driver;

**public** **static** ExtentReports *extent* = **new** ExtentReports();

**public** **static** ExtentSparkReporter *spark* = **new** ExtentSparkReporter("./reports120122/onlineBanking"+System.*currentTimeMillis*()+".html");

**public** **static** ExtentTest *logger*;

@BeforeSuite

**public** **void** suiteSetup() {

*extent*.attachReporter(*spark*);

}

@AfterSuite

**public** **void** teardown() {

*extent*.flush();

//driver.close();

}

@AfterMethod

**public** **void** getResult(ITestResult result) {

**if**(result.getStatus()==ITestResult.***SUCCESS***) {

*logger*.pass("test is success");

}

**else** **if**(result.getStatus()==ITestResult.***FAILURE***) {

*logger*.fail("test is failed");

}

**else** **if**(result.getStatus()==ITestResult.***SKIP***) {

*logger*.skip("test is skipped");

}

*extent*.flush();

}

}

**public** **class** Extent\_test1 **extends** BaseClass120122 {

WebDriver driver;

@Test

**public** **void** test\_a() {

WebDriverManager.*chromedriver*().setup();

driver = **new** ChromeDriver();

driver.get("http://facebook.com");

*logger* = *extent*.createTest("test a");

*logger*.info("test a passed");

String path = ScreenshotTaker120122.*take\_screenshot*(driver, "bank"+System.*currentTimeMillis*()+"screen");

*logger*.addScreenCaptureFromPath(path);

driver.close();

}

@Test

**public** **void** test\_b() {

WebDriverManager.*chromedriver*().setup();

driver = **new** ChromeDriver();

driver.get("http://facebook.com");

*logger* = *extent*.createTest("test b");

*logger*.info("test b passed");

String path = ScreenshotTaker120122.*take\_screenshot*(driver, "bank"+System.*currentTimeMillis*()+"screen");

*logger*.addScreenCaptureFromPath(path);

driver.close();

}

}

-------------------------------------------------------------------------------------------------------------------------

**package** com.inetbanking.utilities;

**public** **class** XLUtils {

**public** **static** FileInputStream *fi*;

**public** **static** FileOutputStream *fo*;

**public** **static** XSSFWorkbook *wb*;

**public** **static** XSSFSheet *ws*;

**public** **static** XSSFRow *row*;

**public** **static** XSSFCell *cell*;

**public** **static** **int** getRowCount(String xlfile,String xlsheet) **throws** IOException

{

*fi*=**new** FileInputStream(xlfile);

*wb*=**new** XSSFWorkbook(*fi*);

*ws*=*wb*.getSheet(xlsheet);

**int** rowcount=*ws*.getLastRowNum();

*wb*.close();

*fi*.close();

**return** rowcount;

}

**public** **static** **int** getCellCount(String xlfile,String xlsheet,**int** rownum) **throws** IOException

{

*fi*=**new** FileInputStream(xlfile);

*wb*=**new** XSSFWorkbook(*fi*);

*ws*=*wb*.getSheet(xlsheet);

*row*=*ws*.getRow(rownum);

**int** cellcount=*row*.getLastCellNum();

*wb*.close();

*fi*.close();

**return** cellcount;

}

**public** **static** String getCellData(String xlfile,String xlsheet,**int** rownum,**int** colnum) **throws** IOException

{

*fi*=**new** FileInputStream(xlfile);

*wb*=**new** XSSFWorkbook(*fi*);

*ws*=*wb*.getSheet(xlsheet);

*row*=*ws*.getRow(rownum);

*cell*=*row*.getCell(colnum);

String data;

**try**

{

DataFormatter formatter = **new** DataFormatter();

String cellData = formatter.formatCellValue(*cell*);

**return** cellData;

}

**catch** (Exception e)

{

data="";

}

*wb*.close();

*fi*.close();

**return** data;

}

**public** **static** **void** setCellData(String xlfile,String xlsheet,**int** rownum,**int** colnum,String data) **throws** IOException

{

*fi*=**new** FileInputStream(xlfile);

*wb*=**new** XSSFWorkbook(*fi*);

*ws*=*wb*.getSheet(xlsheet);

*row*=*ws*.getRow(rownum);

*cell*=*row*.createCell(colnum);

*cell*.setCellValue(data);

*fo*=**new** FileOutputStream(xlfile);

*wb*.write(*fo*);

*wb*.close();

*fi*.close();

*fo*.close();

}

}

-------------------------------------------------------------------------------------------------------------------------

**package** com.inetbanking.pageObjects;

**public** **class** AddCustomerPage {

WebDriver ldriver;

**public** AddCustomerPage(WebDriver rdriver) {

ldriver=rdriver;

PageFactory.*initElements*(rdriver, **this**);

}

@FindBy(how = How.***XPATH***, using ="/html/body/div[3]/div/ul/li[2]/a")

@CacheLookup

WebElement lnkAddNewCustomer;

@FindBy(how = How.***NAME***, using = "name")

@CacheLookup

WebElement txtCustomerName;

@FindBy(how = How.***NAME***, using = "rad1")

@CacheLookup

WebElement rdGender;

@CacheLookup

@FindBy(how = How.***ID\_OR\_NAME***, using = "dob")

WebElement txtdob;

@CacheLookup

@FindBy(how = How.***NAME***, using = "addr")

WebElement txtaddress;

@CacheLookup

@FindBy(how = How.***NAME***, using = "city")

WebElement txtcity;

@CacheLookup

@FindBy(how = How.***NAME***, using = "state")

WebElement txtstate;

@CacheLookup

@FindBy(how = How.***NAME***, using = "pinno")

WebElement txtpinno;

@CacheLookup

@FindBy(how = How.***NAME***, using = "telephoneno")

WebElement txttelephoneno;

@CacheLookup

@FindBy(how = How.***NAME***, using = "emailid")

WebElement txtemailid;

@CacheLookup

@FindBy(how = How.***NAME***, using = "password")

WebElement txtpassword;

@CacheLookup

@FindBy(how = How.***NAME***, using = "sub")

WebElement btnSubmit;

**public** **void** clickAddNewCustomer() {

lnkAddNewCustomer.click();

}

**public** **void** custName(String cname) {

txtCustomerName.sendKeys(cname);

}

**public** **void** custgender(String cgender) {

rdGender.click();

}

**public** **void** custdob(String mm,String dd,String yy) {

txtdob.sendKeys(mm);

txtdob.sendKeys(dd);

txtdob.sendKeys(yy);

}

**public** **void** custaddress(String caddress) {

txtaddress.sendKeys(caddress);

}

**public** **void** custcity(String ccity) {

txtcity.sendKeys(ccity);

}

**public** **void** custstate(String cstate) {

txtstate.sendKeys(cstate);

}

**public** **void** custpinno(String cpinno) {

txtpinno.sendKeys(String.*valueOf*(cpinno));

}

**public** **void** custtelephoneno(String ctelephoneno) {

txttelephoneno.sendKeys(ctelephoneno);

}

**public** **void** custemailid(String cemailid) {

txtemailid.sendKeys(cemailid);

}

**public** **void** custpassword(String cpassword) {

txtpassword.sendKeys(cpassword);

}

**public** **void** custsubmit() {

btnSubmit.click();

}

}

**package** com.inetbanking.pageObjects;

**public** **class** LoginPage {

WebDriver ldriver;

**public** LoginPage(WebDriver rdriver)

{

ldriver=rdriver;

PageFactory.*initElements*(rdriver, **this**);

}

@FindBy(name="uid")

@CacheLookup

WebElement txtUserName;

@FindBy(name="password")

@CacheLookup

WebElement txtPassword;

@FindBy(name="btnLogin")

@CacheLookup

WebElement btnLogin;

@FindBy(xpath="/html/body/div[3]/div/ul/li[15]/a")

@CacheLookup

WebElement lnkLogout;

**public** **void** setUserName(String uname)

{

txtUserName.sendKeys(uname);

}

**public** **void** setPassword(String pwd)

{

txtPassword.sendKeys(pwd);

}

**public** **void** clickSubmit()

{

btnLogin.click();

}

**public** **void** clickLogout()

{

lnkLogout.click();

}

}

-------------------------------------------------------------------------------------------------------------------------

**package** com.inetbanking.testCases;

**public** **class** BaseClass {

ReadConfig readconfig=**new** ReadConfig();

**public** String baseURL=readconfig.getApplicationURL();

**public** String username=readconfig.getUsername();

**public** String password=readconfig.getPassword();

**public** **static** WebDriver *driver*;

**public** **static** Logger *logger*;

@Parameters("browser")

@BeforeClass

**public** **void** setup(String br)

{

*logger* = Logger.*getLogger*("ebanking");

PropertyConfigurator.*configure*("Log4j.properties");

**if**(br.equals("chrome"))

{

System.*setProperty*("webdriver.chrome.driver",readconfig.getChromePath());

*driver*=**new** ChromeDriver();

}

**else** **if**(br.equals("firefox"))

{

System.*setProperty*("webdriver.gecko.driver",readconfig.getFirefoxPath());

*driver* = **new** FirefoxDriver();

}

**else** **if**(br.equals("ie"))

{

System.*setProperty*("webdriver.ie.driver",readconfig.getIEPath());

*driver* = **new** InternetExplorerDriver();

}

*driver*.manage().timeouts().implicitlyWait(10,TimeUnit.***SECONDS***);

*driver*.get(baseURL);

}

@AfterClass

**public** **void** tearDown()

{

*driver*.quit();

}

**public** **void** captureScreen(WebDriver driver, String tname) **throws** IOException {

TakesScreenshot ts = (TakesScreenshot) driver;

File source = ts.getScreenshotAs(OutputType.***FILE***);

File target = **new** File(System.*getProperty*("user.dir") + "/Screenshots/" + tname + ".png");

FileUtils.*copyFile*(source, target);

System.***out***.println("Screenshot taken");

}

**public** String randomestring()

{

String generatedstring=RandomStringUtils.*randomAlphabetic*(8);

**return**(generatedstring);

}

**public** **static** String randomeNum() {

String generatedString2 = RandomStringUtils.*randomNumeric*(4);

**return** (generatedString2);

}

}

-------------------------------------------------------------------------------------------------------------------------

**package** com.inetbanking.testCases;

**public** **class** TC\_LoginTest\_001 **extends** BaseClass

{

@Test

**public** **void** loginTest() **throws** IOException

{

*logger*.info("URL is opened");

LoginPage lp=**new** LoginPage(*driver*);

lp.setUserName(username);

*logger*.info("Entered username");

lp.setPassword(password);

*logger*.info("Entered password");

lp.clickSubmit();

**if**(*driver*.getTitle().equals("Guru99 Bank Manager HomePage"))

{

Assert.*assertTrue*(**true**);

*logger*.info("Login test passed");

}

**else**

{

captureScreen(*driver*,"loginTest");

Assert.*assertTrue*(**false**);

*logger*.info("Login test failed");

}

}

}

-------------------------------------------------------------------------------------------------------------------------

**package** com.inetbanking.testCases;

**public** **class** EditCustomer {

//Edit customer

}

**package** com.inetbanking.testCases;

**public** **class** TC\_LoginDDT\_002 **extends** BaseClass

{

@Test(dataProvider="LoginData")

**public** **void** loginDDT(String user,String pwd) **throws** InterruptedException

{

LoginPage lp=**new** LoginPage(*driver*);

lp.setUserName(user);

*logger*.info("user name provided");

lp.setPassword(pwd);

*logger*.info("password provided");

lp.clickSubmit();

Thread.*sleep*(3000);

**if**(isAlertPresent()==**true**)

{

*driver*.switchTo().alert().accept();//close alert

*driver*.switchTo().defaultContent();

Assert.*assertTrue*(**false**);

*logger*.warn("Login failed");

}

**else**

{

Assert.*assertTrue*(**true**);

*logger*.info("Login passed");

lp.clickLogout();

Thread.*sleep*(3000);

*driver*.switchTo().alert().accept();//close logout alert

*driver*.switchTo().defaultContent();

}

}

**public** **boolean** isAlertPresent() //user defined method created to check alert is presetn or not

{

**try**

{

*driver*.switchTo().alert();

**return** **true**;

}

**catch**(NoAlertPresentException e)

{

**return** **false**;

}

}

@DataProvider(name="LoginData")

String [][] getData() **throws** IOException

{

String path=System.*getProperty*("user.dir")+"/src/test/java/com/inetbanking/testData/LoginData.xlsx";

**int** rownum=XLUtils.*getRowCount*(path, "Sheet1");

**int** colcount=XLUtils.*getCellCount*(path,"Sheet1",1);

String logindata[][]=**new** String[rownum][colcount];

**for**(**int** i=1;i<=rownum;i++)

{

**for**(**int** j=0;j<colcount;j++)

{

logindata[i-1][j]=XLUtils.*getCellData*(path,"Sheet1", i,j);//1 0

}

}

**return** logindata;

}

}

-------------------------------------------------------------------------------------------------------------------------

**package** com.inetbanking.testCases;

**public** **class** TC\_AddCustomerTest\_003 **extends** BaseClass

{

@Test

**public** **void** addNewCustomer() **throws** InterruptedException, IOException

{

LoginPage lp=**new** LoginPage(*driver*);

lp.setUserName(username);

*logger*.info("User name is provided");

lp.setPassword(password);

*logger*.info("Passsword is provided");

lp.clickSubmit();

Thread.*sleep*(3000);

AddCustomerPage addcust=**new** AddCustomerPage(*driver*);

addcust.clickAddNewCustomer();

*logger*.info("providing customer details....");

addcust.custName("Pavan");

addcust.custgender("male");

addcust.custdob("10","15","1985");

Thread.*sleep*(5000);

addcust.custaddress("INDIA");

addcust.custcity("HYD");

addcust.custstate("AP");

addcust.custpinno("5000074");

addcust.custtelephoneno("987890091");

String email=randomestring()+"@gmail.com";

addcust.custemailid(email);

addcust.custpassword("abcdef");

addcust.custsubmit();

Thread.*sleep*(3000);

*logger*.info("validation started....");

**boolean** res=*driver*.getPageSource().contains("Customer Registered Successfully!!!");

**if**(res==**true**)

{

Assert.*assertTrue*(**true**);

*logger*.info("test case passed....");

}

**else**

{

*logger*.info("test case failed....");

captureScreen(*driver*,"addNewCustomer");

Assert.*assertTrue*(**false**);

}

}

}

===================== implemented by me ====================

**package** com.iNetBanking.utilities;

**public** **class** ScreenShotTaker {

//static WebDriver driver = null;

**public** **static** String take\_screenshot(WebDriver driver, String screenName) {

//String timeStamp = new SimpleDateFormat("MM.dd.yyyy\_HH.mm.ss\_").format(new Date());

String ScreenShotPath = **null**;

**try** {

TakesScreenshot ts = (TakesScreenshot)driver;

File sourceFile = ts.getScreenshotAs(OutputType.***FILE***);

File destFile = **new** File("./screenShots/"+screenName+"\_"+System.*currentTimeMillis*()+".png");

FileUtils.*copyFile*(sourceFile, destFile);

ScreenShotPath = destFile.getAbsolutePath();

} **catch** (Exception e) {

System.***out***.println("Excepton : "+e.getMessage());

}

**return** ScreenShotPath;

}

**public** **static** **void** main(String[] args) {

WebDriverManager.*chromedriver*().setup();

WebDriver driver = **new** ChromeDriver();

System.***out***.println("browser opened");

driver.get("https://www.facebook.com/");

System.***out***.println("facebook page opened");

ScreenShotTaker.*take\_screenshot*(driver, "facebookScreen");

System.***out***.println("screenShot taken");

}

}

-------------------------------------------------------------------------------------------------------------------------

**public** **class** Reporting **extends** TestListenerAdapter{

**public** WebDriver driver;

**public** ExtentHtmlReporter htmlReporter = **null**;

**public** ExtentReports reports = **null**;

**public** ExtentTest logger = **null**;

**public** **void** onStart(ITestContext testContext) {

String timeStamp = **new** SimpleDateFormat("MM.dd.yyyy.HH.mm.ss").format(**new** Date());

//String reportName = "Test-report- "+timeStamp+".html";

htmlReporter = **new** ExtentHtmlReporter("./extentReports/extent"+timeStamp+".html");

//htmlReporter.loadXMLConfig("./extentReports/extent"+timeStamp+".xml");

reports = **new** ExtentReports();

reports.attachReporter(htmlReporter);

reports.setSystemInfo("hostName", "localHost");

reports.setSystemInfo("invironment", "QA");

reports.setSystemInfo("user", "Jawad");

htmlReporter.config().setDocumentTitle("Banking project");

htmlReporter.config().setReportName("Functonal Test Report");

htmlReporter.config().setTestViewChartLocation(ChartLocation.***TOP***);

htmlReporter.config().setTheme(Theme.***DARK***);

Reporter.*log*("on Start"+**true**);

//reports.flush();

}

**public** **void** onTestSuccess(ITestResult result) {

**try** {

logger = reports.createTest(result.getName());

logger.pass("test passed"+ result.getName()+" "+ExtentColor.***GREEN***);

Reporter.*log*("on test success"+**true**);

// String imagePath = ScreenShotTaker.take\_screenshot(driver, "facebookScreen");

// File file = new File(imagePath);

// if(file.exists()) {

// logger.addScreenCaptureFromPath(imagePath);

// }

reports.flush();

} **catch** (Exception e) {

System.***out***.println("Exception : "+ e.getMessage());

}

}

**public** **void** onTestFailure(ITestResult result) {

**try** {

logger = reports.createTest(result.getName());

logger.pass("test failed"+ result.getName()+" "+ExtentColor.***RED***);

Reporter.*log*("on test failure"+**true**);

// String imagePath = ScreenShotTaker.take\_screenshot(driver, "facebookScreen");

// File file = new File(imagePath);

// if(file.exists()) {

// logger.addScreenCaptureFromPath(imagePath);

// }

reports.flush();

} **catch** (Exception e) {

System.***out***.println("Exception : "+ e.getMessage());

}

}

}

-------------------------------------------------------------------------------------------------------------------------

**public** **class** Read\_excel\_xls\_data {

**public** String[][] testData() **throws** Exception {

**return** **null**;

}

**public** **static** **void** main(String[] args) **throws** Exception {

File file = **new** File("./Test\_Data/TestData.xlsx");

FileInputStream fis = **new** FileInputStream(file);

XSSFWorkbook wb = **new** XSSFWorkbook(fis);

**int** row = wb.getSheetAt(0).getLastRowNum();

**int** cel = wb.getSheetAt(0).getRow(0).getLastCellNum();

//String data = wb.getSheetAt(0).getRow(1).getCell(1).getStringCellValue();

//System.out.println(data);

**for**(**int** i=0; i<=row; i++) {

**for**(**int** j=0; j<=cel; j++) {

String data = wb.getSheetAt(0).getRow(i).getCell(j).getStringCellValue();

System.***out***.println(data);

}

}

}

}

-------------------------------------------------------------------------------------------------------------------------

**public** **class** Read\_config {

Properties pro = **new** Properties();

**public** Read\_config() {

File file = **new** File("./configurationFiles/Config.properties");

**try** {

FileInputStream fis = **new** FileInputStream(file);

pro.load(fis);

} **catch** (Exception e) {

System.***out***.println("exception message : "+e.getMessage());

System.***out***.println("exception cause : "+e.getCause());

}

}

**public** String getApplicationURL() {

String url = pro.getProperty("baseUrl");

**return** url;

}

**public** String getUserName() {

String uid = pro.getProperty("userName");

**return** uid;

}

**public** String getPassWord() {

String pass\_word = pro.getProperty("passWord");

**return** pass\_word;

}

**public** **static** **void** main(String[] args) {

Read\_config rc = **new** Read\_config();

System.***out***.println("url : "+rc.getApplicationURL());

System.***out***.println("userName :"+rc.getUserName());

System.***out***.println("passWord : "+rc.getPassWord());

}

}

-------------------------------------------------------------------------------------------------------------------------

**public** **class** RandomStringData {

**public** **static** String getRandomStringdata() {

String generatedString = RandomStringUtils.*randomAlphabetic*(8);

**return** generatedString;

}

**public** **static** String getRandomStringNumber() {

String generatedNumber = RandomStringUtils.*randomNumeric*(8);

**return** generatedNumber;

}

}

-------------------------------------------------------------------------------------------------------------------------

**public** **class** ExtentReport {

**static** WebDriver *driver*;

ExtentReports extent;

**static** ExtentTest *logger*;

**public** **void** report3() **throws** IOException {

String timeStamp = **new** SimpleDateFormat("MM.dd.yyyy\_HH.mm.ss").format(**new** Date());

ExtentHtmlReporter htmlReporter = **new** ExtentHtmlReporter("./extentReports/extent"+timeStamp+".html");

extent = **new** ExtentReports();

extent.attachReporter(htmlReporter);

*logger* = extent.createTest("extentReporter3Test");

*logger*.log(Status.***INFO***, "test started");

*logger*.pass("test passed");

*logger*.fail("test failed");

WebDriverManager.*chromedriver*().setup();

*driver* = **new** ChromeDriver();

*driver*.get("http://facebook.com/");

String imagePath = ScreenShotTaker.*take\_screenshot*(*driver*, "facebookScreen");

*logger*.addScreenCaptureFromPath(imagePath);

extent.flush();

}

**public** **void** report4() {

}

**public** **void** report5() {

}

**public** **static** **void** main(String[] args) {

**try** {

ExtentReport er = **new** ExtentReport();

er.report3();

} **catch** (Exception e) {

System.***out***.println("exception : "+e.getMessage());

}

}

}

-------------------------------------------------------------------------------------------------------------------------

**public** **class** BrowserFactory {

**static** WebDriver *driver* = **null**;

//@Parameters("browserName")

**public** **static** WebDriver startBrowser(String browserName, String url) {

**if**(browserName.equals("chrome")) {

WebDriverManager.*chromedriver*().setup();

*driver* = **new** ChromeDriver();

}

**else** **if**(browserName.equals("firefox")) {

WebDriverManager.*firefoxdriver*().setup();

*driver* = **new** FirefoxDriver();

}

**else** **if**(browserName.equals("edge")) {

WebDriverManager.*edgedriver*().setup();

*driver* = **new** EdgeDriver();

}

*driver*.manage().window().maximize();

*driver*.manage().timeouts().pageLoadTimeout(20, TimeUnit.***SECONDS***);

Reporter.*log*("browser opened", **true**);

*driver*.get(url);

//Reporter.log("page is opened", true);

**return** *driver*;

}

}

-------------------------------------------------------------------------------------------------------------------------

**public** **class** AlertHelper {

**public** **static** WebDriver *driver*;

**public** **static** **boolean** isAlertPresent(WebDriver driver) {

**try** {

driver.switchTo().alert();

**return** **true**;

}

**catch**(NoAlertPresentException e) {

**return** **false**;

}

}

}

-------------------------------------------------------------------------------------------------------------------------

**public** **class** LoginPage {

WebDriver driver = **null**;

**public** LoginPage(WebDriver driver) {

**this**.driver = driver;

PageFactory.*initElements*(driver, **this**);

}

@FindBy(xpath ="//input[@name='uid']")

WebElement txtUserId;

@FindBy(xpath ="//input[@name='password']")

WebElement txtPassword;

@FindBy(xpath ="//input[@name='btnLogin']")

WebElement loginButton;

@FindBy(xpath ="//a[normalize-space()='Log out']")

WebElement logOutButton;

**public** **void** typeUserName(String userName) {

txtUserId.sendKeys(userName);

}

**public** **void** typePassword(String passWord) {

txtPassword.sendKeys(passWord);

}

**public** **void** clickLoginButton() {

loginButton.click();

}

**public** **void** clickLogOutButton() {

logOutButton.click();

}

}

-------------------------------------------------------------------------------------------------------------------------

**public** **class** AddCustomerPage {

**public** WebDriver driver = **null**;

**public** AddCustomerPage(WebDriver driver) {

**this**.driver = driver;

PageFactory.*initElements*(driver, **this**);

}

@FindBy(xpath ="/html/body/div[3]/div/ul/li[2]/a")

WebElement newCustomer;

@FindBy(how=How.***XPATH***, using="//input[@name='name']")

WebElement customerName;

@FindBy(how=How.***XPATH***, using="//input[@value='m']")

WebElement radiomale;

@FindBy(how=How.***XPATH***, using="//input[@value='f']")

WebElement radioFemale;

@FindBy(how=How.***XPATH***, using="//input[@id='dob']")

@CacheLookup

WebElement birthDate;

@FindBy(how=How.***XPATH***, using="//textarea[@name='addr']")

@CacheLookup

WebElement address;

@FindBy(how=How.***XPATH***, using="//input[@name='city']")

@CacheLookup

WebElement city;

@FindBy(how=How.***XPATH***, using="//input[@name='state']")

WebElement state ;

@FindBy(how=How.***XPATH***, using="//input[@name='pinno']")

WebElement pinNo;

@FindBy(how=How.***XPATH***, using="//input[@name='telephoneno']")

WebElement phoneNo ;

@FindBy(how=How.***XPATH***, using="//input[@name='emailid']")

WebElement email;

@FindBy(how=How.***XPATH***, using="//input[@name='password']")

WebElement password;

@FindBy(how=How.***XPATH***, using="//input[@name='sub']")

WebElement submitButton;

@FindBy(how=How.***XPATH***, using="//input[@name='res']")

WebElement resetButton;

@FindBy(how=How.***XPATH***, using="//a[normalize-space()='Log out']")

WebElement logOutButton;

**public** **void** clickNewCust() {

newCustomer.click();

}

**public** **void** enterName(String name) {

WebDriverWait wait = **new** WebDriverWait(driver, 10);

wait.until(ExpectedConditions.*visibilityOf*(customerName));

customerName.sendKeys(name);

}

**public** **void** radio\_male() {

radiomale.click();

}

**public** **void** radio\_female() {

radioFemale.click();

}

**public** **void** dob(String mm, String dd, String yyyy) {

birthDate.sendKeys(mm);

birthDate.sendKeys(dd);

birthDate.sendKeys(yyyy);

}

**public** **void** Address(String addresses) {

address.sendKeys(addresses);

}

**public** **void** enterCity(String cityname) {

city.sendKeys(cityname);

}

**public** **void** enterState(String stateName) {

state.sendKeys(stateName);

}

**public** **void** enterPin(String pin\_no) {

pinNo.sendKeys(pin\_no);

}

**public** **void** enterPhoneNo(String phone\_no) {

phoneNo.sendKeys(phone\_no);

}

**public** **void** enterEmail(String emailId) {

email.sendKeys(emailId);

}

**public** **void** enterPass\_word(String pass\_word) {

password.sendKeys(pass\_word);

}

**public** **void** clickSubmitButton() {

submitButton.click();

}

**public** **void** clickResetButton() {

resetButton.click();

}

**public** **void** clickLogOutButton() {

logOutButton.click();

}

}

-------------------------------------------------------------------------------------------------------------------------

**public** **class** BaseClass {

**public** **static** WebDriver *driver* = **null**;

**static** Read\_config *readConfig* = **new** Read\_config();

**public** **static** String *baseUrl* = *readConfig*.getApplicationURL();

**public** String userName = *readConfig*.getUserName();

**public** String passWord = *readConfig*.getPassWord();

@Parameters("browserName")

@BeforeClass

**public** **void** setup(String browserName) {

**if**(browserName.equals("chrome")) {

WebDriverManager.*chromedriver*().setup();

*driver* = **new** ChromeDriver();

}

**else** **if**(browserName.equals("firefox")) {

WebDriverManager.*firefoxdriver*().setup();

*driver* = **new** FirefoxDriver();

}

**else** **if**(browserName.equals("edge")) {

WebDriverManager.*edgedriver*().setup();

*driver* = **new** EdgeDriver();

}

*driver*.manage().window().maximize();

*driver*.manage().timeouts().pageLoadTimeout(20, TimeUnit.***SECONDS***);

Reporter.*log*("browser opened", **true**);

*driver*.get(*baseUrl*);

Reporter.*log*("page is opened", **true**);

}

//@AfterMethod

**public** **void** tearDown1(){

*driver*.close();

System.***out***.println("page closed");

}

//@AfterClass

**public** **void** tearDown() {

*driver*.close();

//driver.quit();

System.***out***.println("browser closed");

}

**public** **static** String take\_screenshot(WebDriver driver, String screenName) {

//String timeStamp = new SimpleDateFormat("MM.dd.yyyy\_HH.mm.ss\_").format(new Date());

String ScreenShotPath = **null**;

**try** {

TakesScreenshot ts = (TakesScreenshot)driver;

File sourceFile = ts.getScreenshotAs(OutputType.***FILE***);

File destFile = **new** File("./screenShots/"+screenName+"\_"+System.*currentTimeMillis*()+".png");

FileUtils.*copyFile*(sourceFile, destFile);

ScreenShotPath = destFile.getAbsolutePath();

} **catch** (Exception e) {

System.***out***.println("Excepton : "+e.getMessage());

}

**return** ScreenShotPath;

}

**public** **static** **void** main(String[] args) {

System.***out***.println("base url : "+*baseUrl*);

}

}

-------------------------------------------------------------------------------------------------------------------------

**public** **class** TC\_loginTest\_001 **extends** BaseClass{

@Test(dataProvider = "login\_data")

**public** **void** loginTest(String userName, String passWord) **throws** Exception {

*driver*.manage().timeouts().implicitlyWait(10, TimeUnit.***SECONDS***);

LoginPage login = **new** LoginPage(*driver*);

Reporter.*log*("LoginPage object created", **true**);

login.typeUserName(userName);

Reporter.*log*("userName entered", **true**);

login.typePassword(passWord);

Reporter.*log*("passWord entered", **true**);

login.clickLoginButton();

Reporter.*log*("clicked login button", **true**);

//if(isAlertPresent()==true) {

**if**(AlertHelper.*isAlertPresent*(*driver*)==**true**) {

*driver*.switchTo().alert().accept();

*driver*.switchTo().defaultContent();

Assert.*assertTrue*(**false**); //login failed

}

**else** {

Assert.*assertTrue*(**true**); //login successful

login.clickLogOutButton();

Thread.*sleep*(2000);

*driver*.switchTo().alert().accept();

*driver*.switchTo().defaultContent();

}

// if(driver.getTitle().equals("Guru99 Bank Manager HomePage")) {

//

// Assert.assertTrue(true);

//

// }

// else {

//

// Assert.assertTrue(false);

// }

}

//user define method.

**public** **static** **boolean** isAlertPresent() {

**try** {

*driver*.switchTo().alert();

**return** **true**;

}

**catch**(Exception e) {

**return** **false**;

}

}

@DataProvider(name = "login\_data")

**public** Object[][] testData(){

**return** **new** Object[][] {

{"mngr455792", "UsEzadU"},

{"thjhffjfdj", "hjlillh"},

{"shsdgffflh", "hliltlh"},

{"mngr455792", "UsEzadU"},

{"shgffddslh", "hlilffh"}

};

}

}

-------------------------------------------------------------------------------------------------------------------------

**public** **class** TC\_loginTest\_002 **extends** BaseClass{

@Test(dataProvider = "login\_data")

**public** **void** loginTest(String userName, String passWord) **throws** Exception {

*driver*.manage().timeouts().implicitlyWait(10, TimeUnit.***SECONDS***);

LoginPage login = **new** LoginPage(*driver*);

Reporter.*log*("LoginPage object created", **true**);

login.typeUserName(userName);

Reporter.*log*("userName entered", **true**);

login.typePassword(passWord);

Reporter.*log*("passWord entered", **true**);

login.clickLoginButton();

Reporter.*log*("clicked login button", **true**);

//if(isAlertPresent()==true) {

**if**(AlertHelper.*isAlertPresent*(*driver*)==**true**) {

*driver*.switchTo().alert().accept();

*driver*.switchTo().defaultContent();

Assert.*assertTrue*(**false**); //login failed

}

**else** {

Assert.*assertTrue*(**true**); //login successful

login.clickLogOutButton();

Thread.*sleep*(2000);

*driver*.switchTo().alert().accept();

*driver*.switchTo().defaultContent();

}

// if(driver.getTitle().equals("Guru99 Bank Manager HomePage")) {

//

// Assert.assertTrue(true);

//

// }

// else {

//

// Assert.assertTrue(false);

// }

}

**public** **static** **boolean** isAlertPresent() {

**try** {

*driver*.switchTo().alert();

**return** **true**;

}

**catch**(Exception e) {

**return** **false**;

}

}

@DataProvider(name = "login\_data")

**public** Object[][] testData() **throws** Exception{

File file = **new** File("");

FileInputStream fis = **new** FileInputStream(file);

XSSFWorkbook wb = **new** XSSFWorkbook(fis);

XSSFSheet s1 = wb.getSheetAt(0);

**int** rowCount = s1.getLastRowNum();

**int** celCount = s1.getLastRowNum();

String login\_data[][] = **new** String[rowCount][celCount];

**for**(**int** i=1; i<=rowCount; i++) {

**for**(**int** j=0; j<=celCount; j++) {

login\_data[i-1][j] = s1.getRow(i).getCell(j).getStringCellValue();

}

}

**return** **null**;

}

}

-------------------------------------------------------------------------------------------------------------------------

**public** **class** AddCustomerPageTest **extends** BaseClass{

@Test

**public** **void** addNewCustomer() **throws** Exception {

LoginPage login = **new** LoginPage(*driver*);

login.typeUserName(userName);

Reporter.*log*("provided userName", **true**);

login.typePassword(passWord);

Reporter.*log*("provided password", **true**);

login.clickLoginButton();

Reporter.*log*("clicked submit Button", **true**);

Thread.*sleep*(2000);

AddCustomerPage addcust = **new** AddCustomerPage(*driver*);

Reporter.*log*("providing customer details", **true**);

addcust.clickNewCust();

Thread.*sleep*(1000);

Reporter.*log*("clicked on new customer", **true**);

//handle unwanted popups.

Robot robot = **new** Robot();

robot.mousePress(InputEvent.***BUTTON1\_DOWN\_MASK***);

robot.mouseRelease(InputEvent.***BUTTON1\_DOWN\_MASK***);

addcust.enterName("jawad");

addcust.radio\_male();

addcust.dob("06", "24", "1985");

addcust.Address("32554 Lane");

addcust.enterCity("Bay");

addcust.enterState("carolina");

addcust.enterPin("123456");

addcust.enterPhoneNo("1244365464");

String Email = RandomStringData.*getRandomStringdata*()+"@gmail.com";

addcust.enterEmail(Email);

addcust.enterPass\_word("ehjkllkj");

addcust.clickSubmitButton();

Reporter.*log*("clicked submit button2", **true**);

Thread.*sleep*(2000);

**boolean** message = *driver*.getPageSource().contains("Customer Registered Successfully!!!");

**if**(message=**true**) {

Assert.*assertTrue*(**true**);

Reporter.*log*("new customer added successfully", **true**);

}

**else** {

ScreenShotTaker.*take\_screenshot*(*driver*, "new\_customer\_page");

Reporter.*log*("could not add new customer successfully", **true**);

Assert.*assertTrue*(**false**);

}

}

**public** String randomString() {

String generatedString = RandomStringUtils.*randomAlphabetic*(8);

**return** generatedString;

}

**public** String randomNumber() {

String generatedNumber = RandomStringUtils.*randomNumeric*(8);

**return** generatedNumber;

}

**public** **boolean** isAlertPresent() {

**try**{*driver*.switchTo().alert();

**return** **true**;

}

**catch**(NoAlertPresentException e) {

**return** **false**;

}

}

}

-------------------------------------------------------------------------------------------------------------------------

**public** **class** DataDrivenTest {

WebDriver driver;

@BeforeMethod

**public** **void** launchBrowser() {

WebDriverManager.*chromedriver*().setup();

driver = **new** ChromeDriver();

driver.get("https://demo.guru99.com/V4/index.php");

driver.manage().window().maximize();

driver.manage().timeouts().implicitlyWait(10, TimeUnit.***SECONDS***);

}

@Test(dataProvider = "login\_data")

**public** **void** loginPage(String userName, String passWord) **throws** Exception {

driver.findElement(By.*xpath*("//input[@name='uid']")).sendKeys(userName);

driver.findElement(By.*xpath*("//input[@name='password']")).sendKeys(passWord);

driver.findElement(By.*xpath*("//input[@name='btnLogin']")).click();

Thread.*sleep*(2000);

**if**(isAlertPresent()==**true**) {

driver.switchTo().alert().accept();

driver.switchTo().defaultContent();

Assert.*assertTrue*(**false**); //login failed

}

**else** {

Assert.*assertTrue*(**true**); //login successful

driver.findElement(By.*xpath*("//a[normalize-space()='Log out']")).click();

Thread.*sleep*(2000);

driver.switchTo().alert().accept();

driver.switchTo().defaultContent();

}

}

@DataProvider(name = "login\_data")

**public** Object[][] testData(){

**return** **new** Object[][] {

{"mngr455792", "UsEzadU"},

{"thjhffjfdj", "hjlillh"},

{"shsdgffflh", "hliltlh"},

{"mngr455792", "UsEzadU"},

{"shgffddslh", "hlillfffh"}

};

}

**public** **boolean** isAlertPresent() {

**try** {

driver.switchTo().alert();

**return** **true**;

}

**catch**(NoAlertPresentException e) {

**return** **false**;

}

}

@AfterMethod

**public** **void** tearDown() {

driver.close();

}

}

-------------------------------------------------------------------------------------------------------------------------

**public** **class** DemoTest {

WebDriver driver = **null**;

@Test

**public** **void** test1() {

WebDriverManager.*chromedriver*().setup();

driver = **new** ChromeDriver();

driver.get("https://demo.guru99.com/V4/index.php");

driver.manage().window().maximize();

driver.manage().timeouts().implicitlyWait(20, TimeUnit.***SECONDS***);

driver.findElement(By.*xpath*("//input[@name='uid']")).sendKeys("mngr455792");

driver.findElement(By.*xpath*("//input[@name='password']")).sendKeys("UsEzadu");

driver.findElement(By.*xpath*("//input[@name='btnLogin']")).click();

}

}

