PART C: Framework

1: Data Driven Testing:

1: DDT(Data Driven Testing):

-->If we wan to modify data in data Hard coding its difficult process

-->We store the data in external resource which will be utilized in script by fetching data from external resource

-->that external resource can be property file, excel , database etc

-->Reading the data from external resource and run the test is called as DDT

Ques: What is advantages of DDT?

1: Maintenance of the test data is easy

2: Modification of the test data in external resource is easy

3: Running test scripts in different credentials is easy

**Property File

Ques: What is properties file?

-->Properties is a java feature file where we can store the data in form of key & value pair,

key value data type should be always string.

Ques: How to read data from properties file:

1: Create a property file

-->select project-->right click-->New-->file-->enter file name with extension .properties--->finish

2: Store data in property file using key=value format

Examp: email=admin@gmail.com

password=123

3: Fetch data from property file

A: Get the java representation object of physical file using FileInputStream

B: Create a object of "Properties" class & load all the keys

C: Read data from property file using getProperty("Key");

D: Utlize data in script:

Script:

```
//Step 1: Used to connect the path of physical file
        FileInputStream f = new FileInputStream("./src/test/resources/data.properties");
        //Step 2: load all the keys
        Properties p = new Properties();
        p.load(f);
        //Step 3: using getproperty() call the keys/fetch data
        String str = p.getProperty("email");
        String str1 = p.getProperty("password");

        WebDriver driver = new FirefoxDriver();
        driver.get("https://www.facebook.com/");
        //utilize data
        driver.findElement(By.id("email")).sendKeys(str);
        driver.findElement(By.id("pass")).sendKeys(str1);
```

Script:

```
//Step 1: Used to connect the path of physical file
        FileInputStream f = new
FileInputStream("./src/test/resources/data.properties");
        //Step 2: load all the keys
        Properties p = new Properties();
        p.setProperty("p1","v1");
        FileOutputStream f1 = new
FileOutputStream("./src/test/resources/data.properties");
        p.store(f1, "commondata");
```

Script:

```
//Step 1: Used to connect the path of physical file
```

```
FileInputStream f = new
FileInputStream("./src/test/resources/data.properties");
//Step 2: load all the keys
Properties p = new Properties();
p.load(f);
//Step 3: using getproperty() call the keys/fetch data
String URL = p.getProperty("url");
String USERNAME = p.getProperty("username");
String PASSWORD = p.getProperty("password");

WebDriver driver = new FirefoxDriver();
driver.manage().window().maximize();
driver.get(URL);
driver.findElement(By.name("user-name")).sendKeys(USERNAME);
driver.findElement(By.name("password")).sendKeys(PASSWORD);
driver.findElement(By.id("login-button")).click();
```

Script:

```
//INserting data to property file
Properties p = new Properties();
p.setProperty("url", "https://www.saucedemo.com");
p.setProperty("username", "standard_user");
p.setProperty("password", "secret_sauce");
FileOutputStream f1 = new
FileOutputStream("./src/test/resources/data1.properties");
p.store(f1,"CommonData");
```

***Data Driven Testing from Excel

-->To handle excel we need to configure Apache POI library to project

-->Procedure to read data from excel

1: Identify path of the excel which need to be handled

2: To open Excel sheet, WorkbookFactory class static method create() is used

WorkbookFactory.create(FileInputStream arg.);

3: To open specific sheet we use workbook Interface methods getSheet().

4: To identify specific row in a sheet we use getRow() method present in Sheet Interface

5: To identify specific cell in a row we use getcell() method present in Row Interface

6: To display cell data as an output we use Cell Interface method, getStringCellValue() or getNumericCellValue()

Script:

```
//Step 1: Path Connection
        String path ="./src/test/resources/excel.xlsx";
        FileInputStream file = new FileInputStream(path);
        //Step 2: Keep the Workbook/excel file in read mode
        Workbook book = WorkbookFactory.create(file);
        //OPen excel sheet
        Sheet s = book.getSheet("Sheet1");
        //identify row
        Row r = s.getRow(0);
        //identify cell
        Cell c = r.getCell(0);
        String data = c.getStringCellValue();
        System.out.println(data);
```

Note:

1: NullPOinterException: If we try to display empty cell data, this exception will happen

2: illegalStateException: This exception will happen if we try to display string datatype by using getNumericCellValue() method.

3: FileNotFoundException: If we have given wrong path this exception will come

-------==========

***getLastRowNum(): This method is used to identify total no. of rows used in a sheet

Script:

```
//Step 1: Path Connection
String path ="./src/test/resources/excel.xlsx";
FileInputStream file = new FileInputStream(path);
//Step 2: Keep the Workbook/excel file in read mode
Workbook book = WorkbookFactory.create(file);
//OPen excel sheet
Sheet s = book.getSheet("Sheet1");
int rowcount = s.getLastRowNum();
System.out.println(rowcount);
```

Output: Output will be in index format

***getLastCellNum() - This method is used to identify total no. of cell used in a row

Script:

```
//Step 1: Path Connection
String path ="./src/test/resources/excel.xlsx";
FileInputStream file = new FileInputStream(path);
//Step 2: Keep the Workbook/excel file in read mode
Workbook book = WorkbookFactory.create(file);
//OPen excel sheet
Sheet s = book.getSheet("Sheet1");
Row r = s.getRow(2);
short cellcount = r.getLastCellNum();
System.out.println(cellcount);
```

Output: it will be in normal format

***setCellValue(): This method is used to override or to enter cell data

**Write(): This method is used to save Excel Changes

--->It should be the last command in test script.

Script:

```
String path ="./src/test/resources/excel.xlsx";
FileInputStream file = new FileInputStream(path);
//Step 2: Keep the Workbook/excel file in read mode
Workbook book = WorkbookFactory.create(file);
//OPen excel sheet
Sheet s = book.getSheet("Sheet2");
//identify row
Row r = s.getRow(0);
//identify cell
Cell c = r.getCell(0);
c.setCellValue("Automation");
FileOutputStream out = new FileOutputStream(path);
book.write(out);
```

**createCell() - This method is used to create new cell in existing row

**createRow() - This methdo is used to create new row.

**createSheet() - This method is used to create new sheet

Script:

```
String path ="./src/test/resources/excel.xlsx";
FileInputStream file = new FileInputStream(path);
//Step 2: Keep the Workbook/excel file in read mode
Workbook book = WorkbookFactory.create(file);
//OPen excel sheet
Sheet s = book.createSheet("Testing");
Row r = s.createRow(1);
```

```
Cell c = r.createCell(1);
c.setCellValue("Selenium");
FileOutputStream out = new FileOutputStream(path);
book.write(out);
```

**removeCell(cell arg): This method is used to remove cell.

**removeRow(row arg): This method is used to remove existing row

**removeSheetAt(int arg): This method is used to remove sheet

Script:

```
String path ="./src/test/resources/excel.xlsx";
FileInputStream file = new FileInputStream(path);
//Step 2: Keep the Workbook/excel file in read mode
Workbook book = WorkbookFactory.create(file);
//OPen excel sheet
Sheet s = book.getSheet("Sheet1");
Row r = s.getRow(0);
s.removeRow(r);
Row r1 = s.getRow(4);
Cell c = r1.getCell(3);
r1.removeCell(c);
book.removeSheetAt(2);
FileOutputStream out = new FileOutputStream(path);
book.write(out);
```

Ques: WATS to login acttitme app by fetching data from excel

Excel: UN PWD

admin manager

Script:

```
WebDriver driver = new FirefoxDriver();
```

```
driver.get("http://localhost/login.do");
//fetch data from excel
Workbook book = WorkbookFactory.create( new
FileInputStream("./src/test/resources/excel.xlsx"));
 Row tgt_row = book.getSheet("Sheet3").getRow(1);
String un = tgt_row.getCell(0).toString();
String pwd = tgt_row.getCell(1).toString();
//enter username
driver.findElement(By.id("username")).sendKeys(un);
driver.findElement(By.name("pwd")).sendKeys(pwd);
driver.findElement(By.id("loginButton")).click();
```

AssignQues: WATS to login to https://www.saucedemo.com/ by fetching data from excel

---

Nested Frame Script:

```
WebDriver driver = new FirefoxDriver();
driver.get("https://the-internet.herokuapp.com/nested_frames");
driver.switchTo().frame("frame-top");
WebElement leftframe = driver.findElement(By.xpath("//frame[@name='frame-
left']"));
driver.switchTo().frame(leftframe);
String str = driver.findElement(By.xpath("//body[contains(text(),'LEFT')]")).getText();
System.out.println(str);
driver.switchTo().parentFrame();
driver.switchTo().frame("frame-middle");
String str1 = driver.findElement(By.xpath("//div[text()='MIDDLE']")).getText();
System.out.println(str1);
driver.close();
```
