

Parameterization

Selenium WebDriver is the most used automation tool for the automation of web applications. Now, we know that these web applications are used by multiple users, and each one of those uses the applications as per their own data.

So, considering the usage, it becomes the primary responsibility of the QAs also to test the web applications with varying data sets.

Now the user journeys will be the same, but the data set will be different.

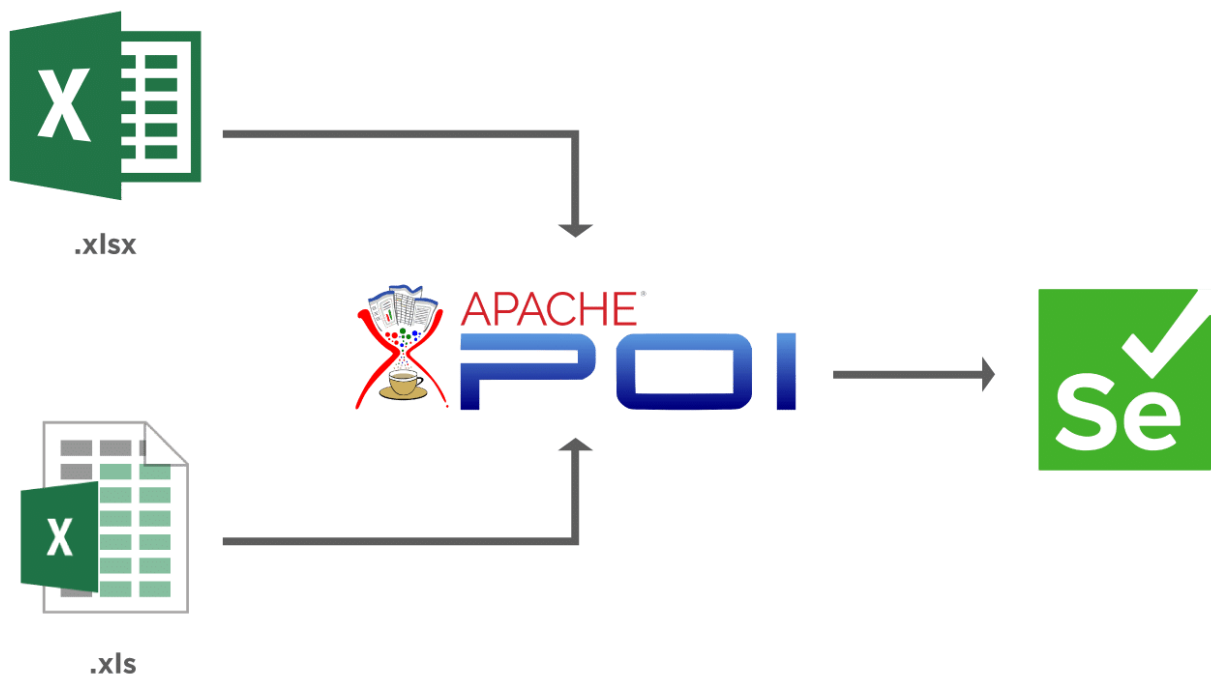
Therefore, it makes more sense to execute the same test case with different data, instead of writing a separate test case for each user journey with each data set.

Microsoft Excel is one of the favorite tools for storing test data.

Apache POI is one of the most used libraries, which provides various classes and methods to read/write data from various formats of Excel files (*xls, xlsx etc*).

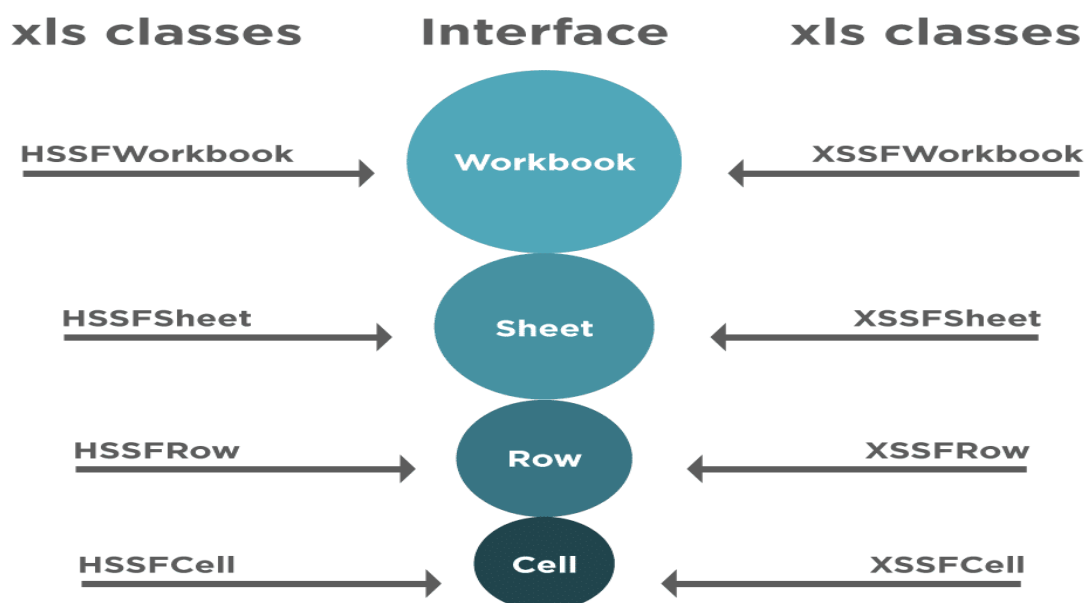
What is Apache POI?

Apache POI, where **POI stands for** (*Poor Obfuscation Implementation*) is an *API* that offers a collection of Java libraries that helps us to *read, write, and manipulate* different Microsoft files such as *excel sheets, power-point, and word files*.



Term	Details
Workbook	A workbook represents a Microsoft Excel file. It can be used for creating and maintaining the spreadsheet. A workbook may contain many sheets .
Sheet	A sheet refers to a page in a Microsoft Excel file that contains the number of rows and columns .
Row	A row represents a collection of cells , which is used to represent a row in the spreadsheet.
Cell	A cell is indicated by a row and column combination. Data entered by a user is stored in a cell. Data can be of the type such as string, numeric value, or formula.

Interfaces and classes in Apache POI



- **HSSF** - These class methods are used to read/write data to Microsoft Excel file in **.xls** format. It is compatible with MS-Office versions 97–2003.
- **XSSF** - These class methods are used to read-write data to Microsoft Excel in **.xls** or **.xlsx** format. It is compatible with MS-Office versions 2007 or later.

How to read data from Excel in Selenium tests using Apache POI?

If You want to read the **value of a particular cell**,

1. The first step is to **obtain the Excel Workbook** based upon its location on the computer. We can create an object of the workbook by referring to the **FileInputStream** object that points to the excel file.

```
File file = new File("E:\\TestData\\TestData.xls");
FileInputStream inputStream = new FileInputStream(file);

HSSFWorkbook wb=new HSSFWorkbook(inputStream);
```

2. Once we create the Workbook, the next step is to create a **Sheet** in the Workbook.

```
HSSFSheet sheet=wb.getSheet("STUDENT_DATA");
```

3. can also create a sheet based upon the index using the **getSheetAt (int index)**

```
HSSFSheet sheet1=wb.getSheetAt(1);
```

4. After the sheet creation, we have to obtain the **row** of the sheet,

```
HSSFRow row1=sheet.getRow(1);
```

5. Once you have got the row, you can get the cell of the row

```
sheet.getRow(1).getCell(1)
```

6. After you obtain the **cell** that contains the data, you can read the data in different formats like **String, Date, Number** using the different methods which are based upon the format of the cell you specify in the excel sheet.

String - getStringCellValue() [It can be used to read Name of the student from Excel]

Number - getNumericCellValue() [It can be used to read the mobile number of the student]

Date - getDateCellValue() [It can be used to read the Date of Birth of the student]

How to write to a new cell in an existing row?

```
XSSFRow row = sheet.getRow(0);
```

1. Create cell

```
row.createCell(2).setCellValue("Pass");
```

2. use fileoutputstream

```
FileOutputStream fos = new FileOutputStream(file);
```

3. use write method -

```
wb.write(fos);
```

How to read the entire Excel sheet?

[Read & Write Data from Excel in Selenium with Example : Apache POI \(toolsqa.com\)](#)