
# Automation Testing

# Automation Framework for Testing




simplilearn

# A Day in the Life of an Automation Test Engineer

Now John has learned the reporting tools, exporting reports to excel, publishing reports to tomcat, and XML parsing.

He now needs to know various frameworks and POM, which reduces code duplication and enhances test case maintenance. Frameworks, which are helpful because IT processes can be extensive and complex, and data-driven testing, should typically be tested at this point for both functionality and actual application logic.

To achieve the above, he will learn about various frameworks, POM, data-driven testing, and other concepts in this lesson.

# Learning Objectives

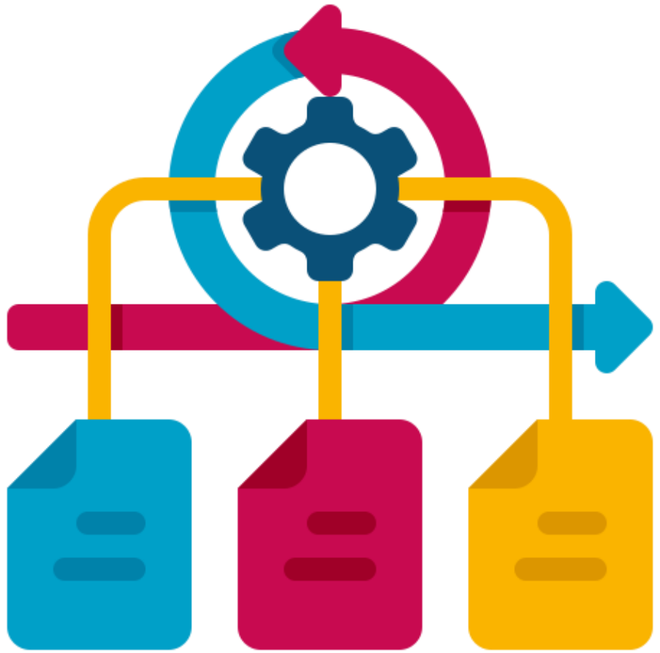
By the end of this lesson, you will be able to:

- List out the various frameworks
- Describe Page Object Model
- Analyze data-driven testing
- Explain Apache POI

# Introduction to Various Frameworks

# Automation Frameworks

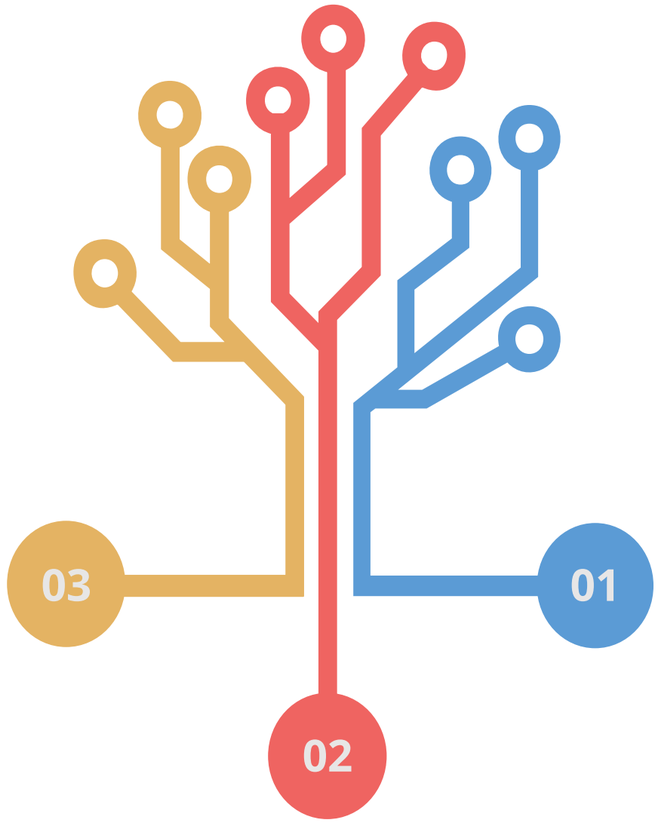
An Automation Framework is a group of tools and procedures that work together to support automated testing of any application.



# Importance of Automation Framework

- The automation framework not only enables the reuse of the code in different scenarios but also helps the team standardize the creation of test scripts.
- The framework supports automation testing as a technical implementation guideline.
- The framework can be helpful when users need to run the same test scripts repeatedly with different builds to analyze the application and confirm output.

# Benefits of Automation Framework




The test code is simple to maintain.

The URL or Application can be tested accurately.

The tests are conducted more quickly than usual.

# Test Automation Framework Types

Some of the frameworks are:

**01** Linear Automation

**02** Modular-Driven

**03** Behavior-Driven

**04** Data-Driven

**05** Keyword-Driven

**06** Hybrid Testing

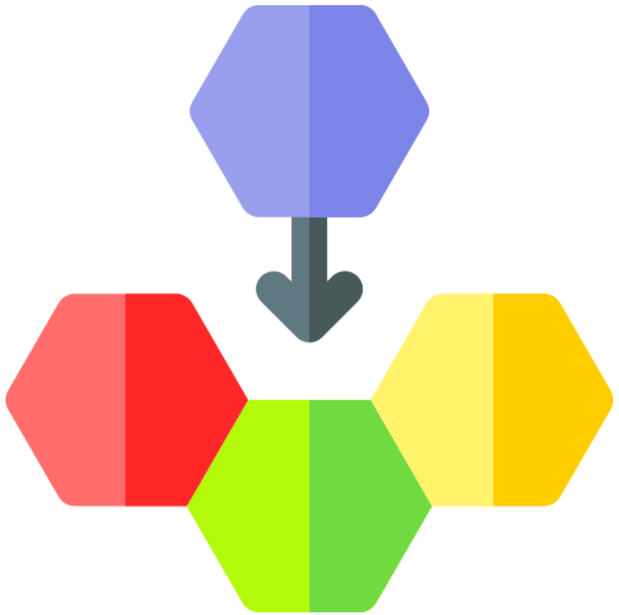
# Linear Automation Framework

Testing of small applications frequently uses the Linear Automation Framework.

The term record and playback framework is also used to describe this framework.

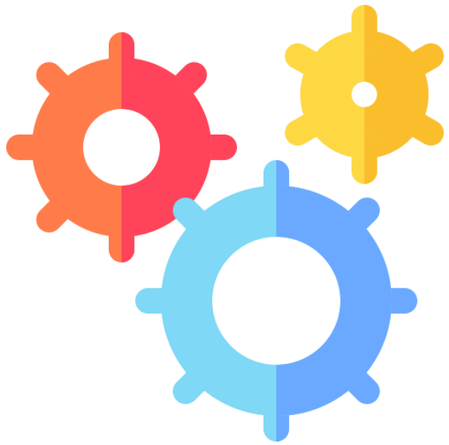
# Modular Driven Framework

The tester can write test scripts individually by segmenting the entire application into smaller modules as per the client's requirements.



# Behavior-driven Development Framework

The Behavior-Driven Development Framework aims to build a platform that encourages active participation from all users, including developers, testers, and business analysts.



Additionally, it improves cooperation on your project between the developers and testers.

# Data-Driven Framework

The test script reads test data from external files, like Excel files, text files, CSV files, ODBC sources, and DAO objects, and loads that data into variables.

Users can write test automation scripts using the data-driven framework by passing various test data sets.

# Keyword-Driven Framework

The keyword-Driven Testing framework is also known as Table-Driven Testing. This framework is suitable only for small projects or applications.

The automation test scripts performed are based on the keywords specified in the excel sheet of the project.

# Page Object Design Pattern

# Page Object Design Pattern

The Page Object Model (POM) is a design pattern or framework used in automation tests to create an object repository for various web elements.

Page Object Model simplifies test maintenance and reduces code duplication.

# Page Object Model

01 In the Page Object Model, users create a class file for each web page.

02 Each class file represents an element on the page.

03 Test scripts use these elements to perform different tasks.

# Implement Page Object Model

To implement Page Object Model, the user must follow the below steps:

01 Define Page Classes

02 Define Test Classes

03 Define POM rescue in case of frequent locator changes

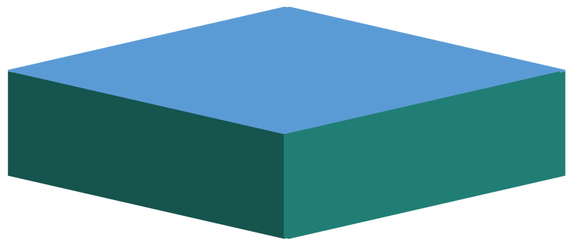
# Advantages of Page Object Design Pattern

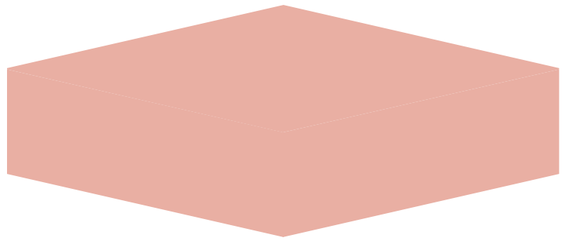
Some of the advantages are:

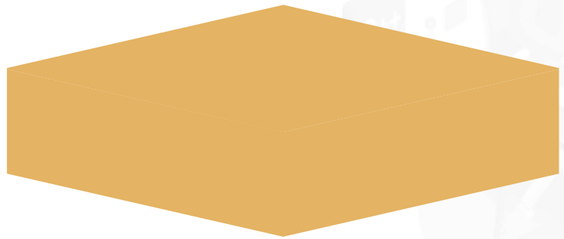
Reusability

Maintainability

Readability







# Data-Driven Testing

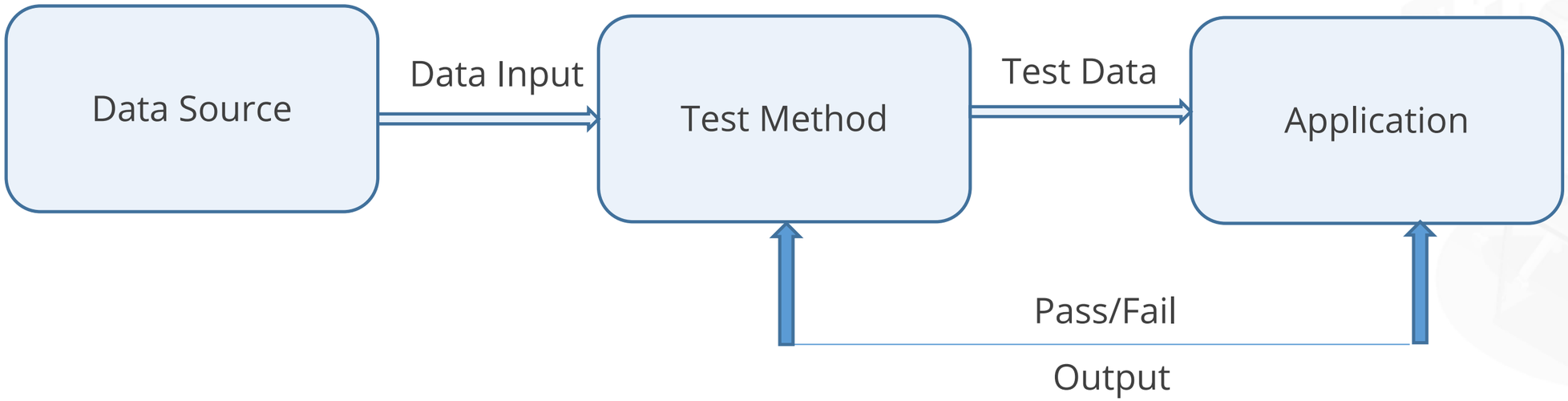
# Data-Driven Testing

Data Driven Testing involves the use of data stored in a data source, such as an XML file, property file, or excel spreadsheet, as input for a test.

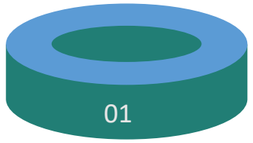
One test script will execute multiple times by utilizing different data.

# Data-Driven Framework

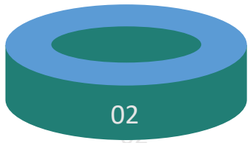
A data-driven framework isolates data from test scripts, allowing the same script to run for various combinations of input test data.



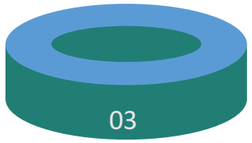
# Advantages of Data-Driven Testing




Test data is not affected by script changes.




It provides a clear separation between the logic and data of test cases or scripts.




Users can run a test case multiple times, reducing scripts and test cases.

04

# Data Provider

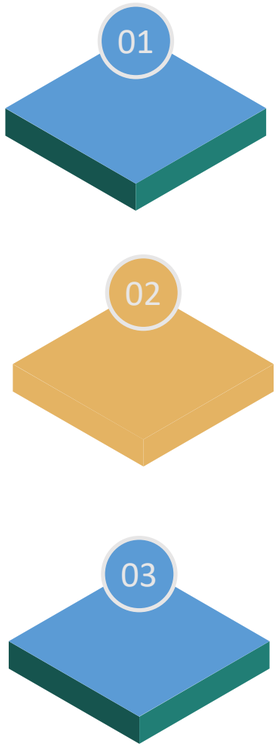
Parameters can be passed directly to TestNG methods in two ways:

**01** XML file

**02** Data providers

# Data Provider

The following categories can be used to categorize data sets:



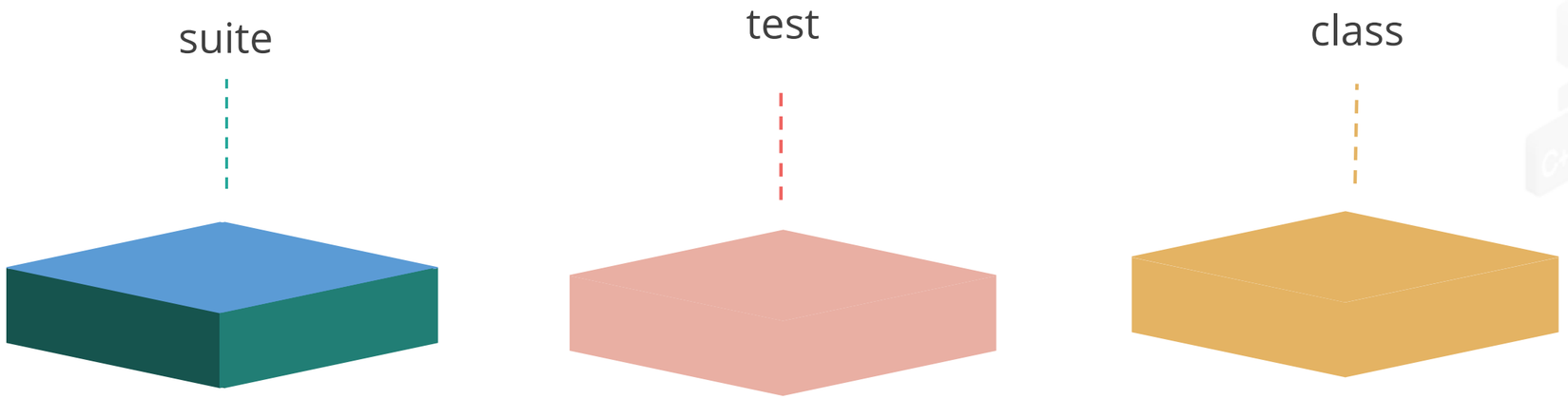

dataprovider Annotation

dataprovider Attribute

dataProviderclass Attribute

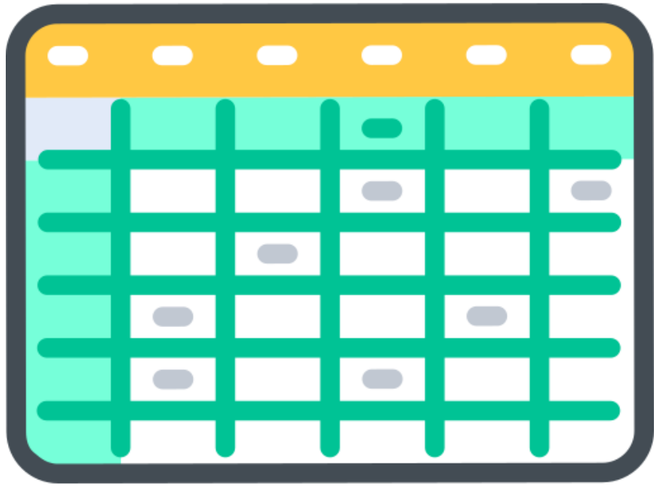
# XML File

An XML file stores and transports data for user testing.
Some of the tags allowed in TestNG's XML file:



# Data Provider Annotation

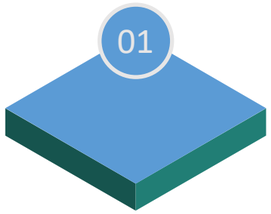
Data Provider Annotation can supply hardcoded data or retrieve data from a data source such as excel spreadsheets, databases, properties files, or CSV files.



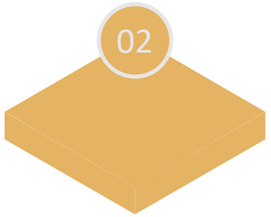
The values can be of any Java data type, including int, boolean, and string.

# Purpose of Data Provider Annotation

Data sets are hard coded to serve two purposes simultaneously.



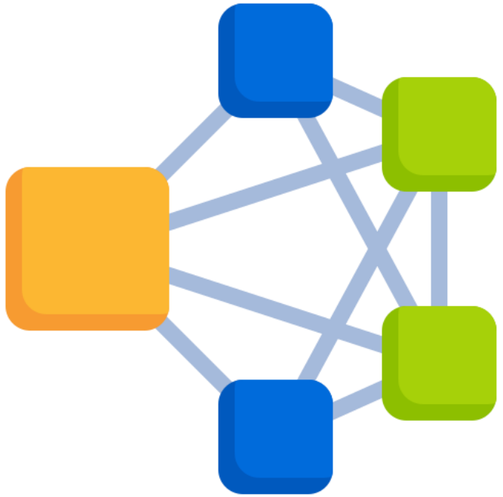

Users can pass an unlimited number of values to the Test method.




A Test method can be invoked with different data sets.

# Data Provider Attribute

A dataProvider attribute maps DataProvider annotation to Test method. A dataProvider attribute must be added to the @Test annotation along with the name of the DataProvider.



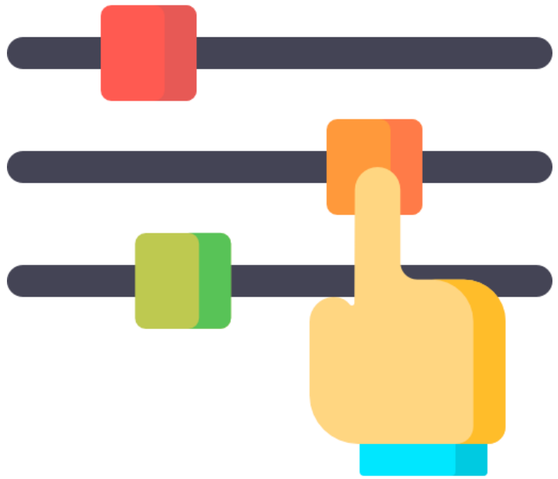
Annotations begin with an uppercase letter **DataProvider**, whereas attributes begin with a lowercase letter **dataProvider**

# Data Provider Class Attribute

The dataProviderClass Attribute permits the DataProvider method and Test method to reside in different classes.

# Parameters Tag

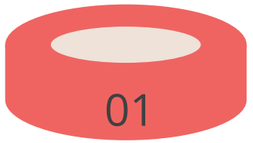
TestNG has a parameter tag that facilitates data-driven testing, ideally used for cross-browser testing.



The parameter tag can be placed within the suite level and the test level with a name and value.

# How to Add Parameters Tag

There are three parameter tags: one at the suite level and two at the test level.




Add the parameter tag to the suite level if each test requires the same value


Add the parameter tag to the test level if each test requires a unique value

# Apache POI

# Introduction to Apache

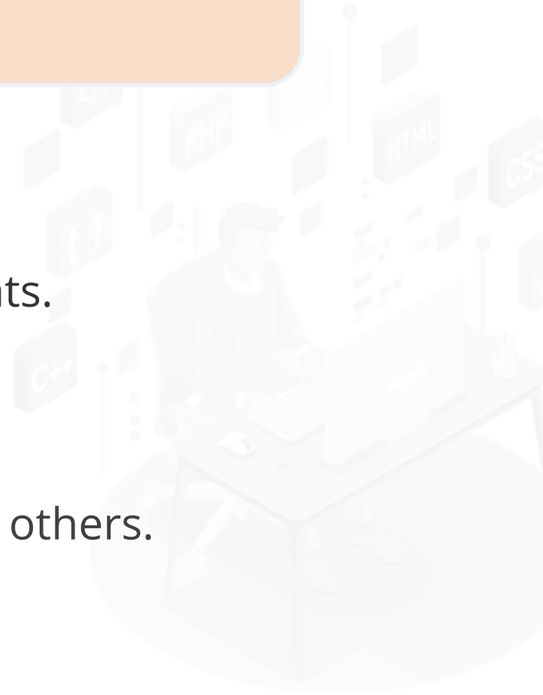
Apache POI (Poor Obfuscation Implementation) is a Java library that manages Microsoft Office documents. It is an open-source library that Java programmers can use to export MS Office files with a read-only API.

# Why Apache POI?

The Apache POI creates and maintains Java APIs to read and write MS Excel files.

01 It can manipulate various Microsoft Office and Open Office file formats.

02 It supports Open XML Format, like XML, XLSX, DOCX, PPTX, and many others.



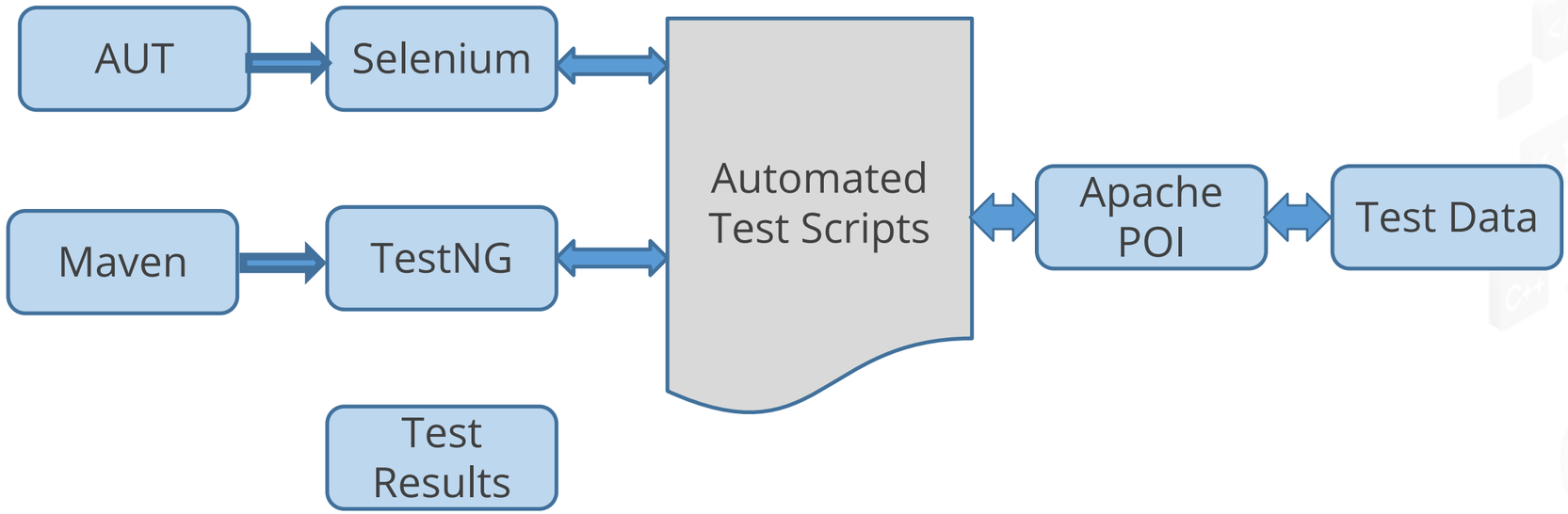
# POI Components

An Apache POI architecture consists of the following components:

- HDGF
  Supports pure Java and limited read capability for version 97-2003
- HPBF
  Supports Java file  format and limited read capability for version 97-2003
- HMEF
  Supports Microsoft TNEF (Transport Neutral Encoding Format)
- HSMF
  Supports Microsoft Outlook message file format in Java
- HWPF
  Supports read and limited write capabilities for versions 97-2003
- HSLF
  Supports read and write capabilities for versions 97-2003

# POI Architecture Framework

In data-driven testing, POI is used to fetch and export data to Excel.



# POI Installation

The POI installation can be done by following the below steps:

01 Download **JAR** files

02 Set the path environment

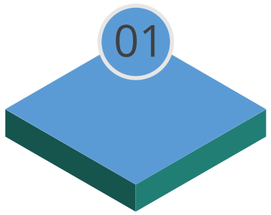
# Fetch Data From POI

The POI parameters are used to fetch data from Excel sheets.

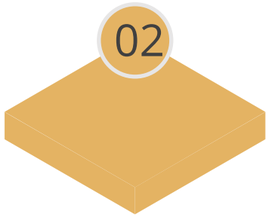
| | |
|---|---|
| Path | The path to the Excel file containing the data relative to the Java project |
| Sheet Name | The name of the Excel spreadsheet that contains the data |

# Advantages of POI

POI offers the following advantages:




It is suitable for large files and uses less memory.




An Excel chart can be copied, and images can be inserted easily.

# Key Takeaways

- The Automation Framework provides tools and procedures to automate any application's testing.
- Page Object Models (POMs) are design patterns or frameworks used in automation tests to create web element object repositories.
- A data-driven test uses data stored in a data source, such as an XML file or property file, as input.
- The Apache POI library manages Microsoft Office documents.