**Auto IT**

It is file an open source tool which is integrated in selenium to handle OS level popups such file Upload, File Download and Windows Popups

It is an open source tool which was developed to automate standalone application inn windows

It is a high level programming language and has lot of library methods through which we can automate window application and it follows syntax like VB Scripting

How to Download AutoIt

1. Go to Link:-"https:www.autoscript.com/site/autoit/download/"
2. Scroll Down and look download for AutoIt icon and click that icon, a zip file will be downloaded
3. unzip the file and then as per the instruction click on next and next and install

Advantages of AutoIt

1. Easy to understand
2. Its free source

Disadvantages of AutoIt

1. It can be used only on windows
2. In order to use this, we need to compile and add it to the project then we should execute this, hence it’s a bit lengthy process

**Data Driven testing**

As a test engineer it’s a good practice to write the test script independent of the test data is called as Data Driven testing

Here data will be stored in external sources like excel sheets and property file.

Data Provider

In order to run single test case with multiple set of data's is called as Data Driven testing in order to achieve this we use Data provider annotation from TestNG

Procedure

1. Create a method and annotate it with at data provider and give the name using name Argument
2. Method must Return Object 2D Array or object[]
3. Fill the Array with all the required data inside the method and Return the Array
4. Add the DataProvider Name for @Test Annotated method
5. In @Test Method declare the number of arguments depending on the number of data in each sub Array
6. When this data cases run the number of execution depends on number of sub array's and it will inject the data based on each sub Array

How to read a data from the data provider which is present in another class?

1. By Assigning the name of the class in which @DataProvider Annotated method is available to DataProvider Class argument for @Test Annotation.
2. We should assign Name of the DataProvider to the DataProvider argument of @Test Annotated method

**Cross Browser Execution: -**

How to Perform

We should use parameterization in xml file and in @Test Annotation

1. Select the TestCasses in the package Right Click> TestNG> Convert to TestNG> Finish
2. In TestNG.xml File under <suit> tag create 2 test tags having same class name and the name of tags based on browser name for example

Chrome Test or Fire Fox Test

1. Add parameter tag for each Test tags with Name and Value Ex: <Parameter name="browser" value="chrome"></parameter>
2. In Java file use Meta Annotation @ Parameters and receive the parameters from xml file Ex: @Parameter("browser")
3. Create a String argument for @Test Annotated Method to receive the value of parameter browser from xml file
4. Write a logic inside @Test Annotated Method to create the Browser object based on received parameter using if else block

**Maven Project**

1. Maven is a build testing tool or build management tool which is used for build creation and build testing and build deployment
2. Developers use this tool for build creation, Build Installation and Build Deployment, in Automation we are using it for Build Testing and to handle dependencies

Build Testing

Here we are checking Compilation and Integration issue between the Source Code

Build Creation Process

The process of converting Source Code into Executable File format is called as Build Creation

Build Deployment

Deploy the Executable File in the server is called as Build Deployment Process

Why Maven in Automation

When Multiple Engineer working with the same Framework their might be chances one engineer modification might affect the entire Framework in order to resolve these kind of issues we will go for maven

Advantages of Maven

1. Check the Integration and compilation issues between the Framework components
2. To handle Dependency Jars
3. To create a good folder structure, so everyone can maintain uniform structure
4. We can Run the Test Script in Command Line without Eclipse
5. Maven supports Jenkins
6. Maven supports Profiling

What are the Maven software requiring to handle Maven project?

1. Maven Eclipse plugin available in Eclipse IDE which is used to create Maven project in Eclipse
2. Maven command line plugin:-It should be installed explicitly in local computer which is used to handle Maven project in command line

Procedure to create Maven project

1. Eclipse > New > Project > Maven > Maven Project > Entire Group Id and Artifact Id > Finish

Note: Where Group Id is of name of Community or Organization Artifact Id is Name of Project

Project Folder Structure

Src/main/java------> Generic Library and P.O.M Repository

Src/main/resources--->driver.exe's & files related to project

Src/test/java----->All the test Script (module wise)

Src/test/resources--->test resources/data required for test scripts

target-------------> in this folder all the reports will (emailable-report. html be stored if we run the test scripts from Maven (Emailable\_report.html)

Installation steps for Maven Complier

1. Go to Google and search for download Maven
2. Click on first link
3. Click on binary zip archive-->the zip file start downloading
4. Go to Downloads extract the zip file open the file and copy the bin path
5. Go to Environment variables and paste this path under system variable (If it’s not working paste the path under user variables)
6. To check whether Maven is configured properly or not Go to Command Prompt(CMD) and Type mvn -version

What is Maven dependency how to handle?

Dependency is a feature available in the Maven Tool which is used to get all Download, Required jar files from official Maven Repository to local repository

(Local Repository: -The Repository which is inside our System. Path: -c://users>user>.m2)

How to handle Maven Dependency?

1. Create Maven project
2. Go to POM.xml and add Dependency tag
3. Go to Global repository that is Maven repository, search for required tool like Selenium Java, TestNG, Apache Poi etc.
4. Copy the dependency code and paste it inside the dependency tag.
5. Save the file as soon as you saved the POM will search for local repository, if that file is not found in local repository, it will automatically connect
6. to Global repository and it will add all those jars to the build path

Maven Commands or Maven Build Lifecycle

1. Maven Clean: -It is used to clean the report from the target file
2. Maven Validate: -Used to validate entire Framework
3. Maven Compile: -Used to check the compilation and Integration issues if there is any jar is missing it will start downloading the jar automatically
4. Mvn Test:-It uses to execute all the test script from Src java test
5. Mvn Package: -Used to Wrap all the source code into .jar or .exe file
6. Mvn Install: -Used to install jar files in different environment or platform
7. Mvn Deploy: -Used to Deploy the project into production environment

How to Execute Test Script from Command Line?

1. Open Eclipse > Right Click on Project which you want to Execute > Properties > Go to Project Location > Select the path and Type Cmd > command prompt

will open in the project directory > now start executing maven commands

Note: -We have to name out test case suffix with test, if not maven will not have considered class as a test case

How to execute TestNg.xml via POM.xml

1. To Execute TestNg.xml via POM.xml we should include Maven sure-fire plugin inside the POM.xml which have to be executed
2. If you Add sure-fire plugin to your POM class, it will execute only the xml file which is their inside xml files

What is Maven Profiling

1. When Framework Contains multiple xml files in order to execute only one particular xml file we will go for Profiling
2. Command: -mvn test -P Profiling Id name

How to Execute Only one Test Case using maven project

1. If we want to execute only one Test Case in the command line, we have to write the command shown below

mvn -D test= Test Case Name Test

Dependencies Required to develop a Framework

1. Selenium-java: -To add Selenium libraries into the project
2. Apache POI and Apache POI-ooxml: -To do Data Driven Testing where data is reading from Excel Sheet
3. TestNG: -To add TestNG plugin to the project
4. WebDriverManager: -Where we can launch the browser without system. Set: Property
5. Common's IO: -To take the Screenshot
6. Extend Reports: -To Generate Extend Reports in Selenium
7. Cucumber Java: -To create cucumber Framework

**Framework**

Framework is a collection of reusable components that, makes Automation Development and Execution and Modification easily

Framework is a set of instructions followed by every organization that makes Automation Test Engineers life easy.

**Framework Types**

There are 5 types

1. Data Driven Framework
2. Keyword Driven Framework
3. Behavioural Driven Development Framework
4. Modular Driven Framework
5. Hybrid Framework
6. Data Driven Framework
   1. Here in this Framework all the Test Case should have at least one @ Data Provider Annotation
   2. Usually this kind of Frameworks will be used in application such as Banking or CRM Because the interaction of Data will be more
   3. Actually this Framework is not widely used in IT Because of the Performance issues
7. Keyword Driven Framework
   1. This Framework will be used when there is a less Resources, by then Automation Test Engineers will give K.T to Manual Test Engineer and Manual Test

Engineers use some Excel Sheet to Write the Code or to Locate the Elements and Automation Test Engineers will write the scrip to get the Data from

the Excel and use them in Scripts.

* 1. It’s a Drastic failure in the industry since it has its own drawback such as Performance issues and Communication Issues Between Test Engineers

1. Modular Driven Framework
   1. In this type all the Test Scripts and Test Data and Reports segregated in Module Wise
   2. This Model Didn’t Workout in the industry since maintenance was the problem
2. Behavioural Driven Development
   1. In the Project Their will be non-Technical People such as Stack Holders, Customers, Product Owner's etc. in order to make them understand all Automation script we will go for Cucumber
   2. Its Basically developed to bridge the gap between technical and non-technical people
   3. Here we will write feature file to describe our business scenarios in Gherkin language, here we will use terminologies like feature's given when and then.
3. Hybrid Framework
   1. It’s the combination of any 2 Frameworks, usually the combination will be Data Driven and Modular Driven

Generally, Framework Contains 8 components

1. Generic Utility
2. Object Repository
3. Test Data
4. Test Script
5. Driver Files or xml Files
6. HTMl Reports
7. Screenshot
8. Maven
9. Jenkins

**Generic Utility: -** It is one of the common component in my Framework, it can be used in any project

It Contains Several usable classes such as Base Class or Base test, Excel Utility, I-Auto Constants or Path Constants,

Screenshot Utility or Utility Classes etc.

1. Base Class: - It contains common TestNG Configuration Annotation which is required for all the test script as per Automation Rule Everyday each Test Script should extend Base Class to use those Annotations.
   1. Before Suit:-used to configure data base connectivity and reports, it will be executed before executing test tag in xml file
   2. Before Test:-Its used to Launch the browser, it will execute before class Annotation
   3. Before Class:-Its used launch the browser in sequential execution it will get executed Before Class
   4. Before Method:-Its used to login to the application which is excited before executing every @Test Method
   5. After Method:-Its used to logout from the application which is executed after executing @Test Method
   6. After Class:-Its used to close the browser, it will get executed after executing last class
   7. After Suit:-Its used to close the database connectivity
2. Excel Utility:-Its developed using Apache poi which is used to read the data from Excel sheet. as per the rule of Automation, data should not be hardcoded within the Test Script so we will take the help of Excel Utility to Interact with Excel File
3. Utility Class or File Utility:-Its used to get the data from property file, which is developed using java
4. Path Constants or I Auto Constants:-It contains common variables which used across the Framework Example: -Excel path, Property File Path or Screenshot path etc.

2:-**Object Repository: -** It’s a collection of reusable WebElement and Business Library which can be used to specific or Business or Project, in order to develop we

have used POM design pattern. As Per the Automation Rule Elements should not be Hardcoded in Test Scripts because modification and maintenance of the Elements is Tedious

so in real time we are using POM repository which helps us to maintain and in modification.

3: -**Test Data**: - It’s one of the components of my Framework, it contains data which is required to run the Framework, there are 2 types of data

1. Common Data: -This is the data we have stored all the details which is required throughout the project ex: -URL, Username Password
2. Test Script data:-It’s a data which is required for all the Test Script, by using This Modification and Maintenance is easy

4: -**Test Script: -** It contains Collection of TestNG script which is Automated using @Test

During Test Script development make sure Generic Library and Object Repository is being using,

In real time Test Case is allocated we should create separate Packages for a Module and develop Test Script by using Utility classes

**5: -Reports:-**In Real Time Reporting component is very much important because it provides the quality of the application and some reports we should submit

to customer. in real time report is also used to help us to know the Route Cause of the issue, Whenever the script is getting failed

**6:-Screenshot:-**Whenever any Test Script is getting failed during execution, will have the Screenshot in Screenshot folder

Advantages of Framework

1. Test Script development is easy and faster since reusability of code.
2. Modification and Maintenance of data is easy Because Data is placed in External Resource
3. Modification and Maintenance of Elements is easy because of POM

Disadvantages of Framework

1. Programming Knowledge is required

Difference Between Annotation and Method

1. Method is subset of class, but Annotation is Java Type
2. A Method can be Developed independent of variable, but Annotation should be placed above the Method or a Variable or Class
3. For a Method has an Argument we can directly pass a data or a variable, but for Annotation we have to pass Data only it will not accept Variables
4. Execution is bit slower, But Annotation Execution is Faster

**GitHub**

1. It’s a Distribute or Decentralized or Cloud Repository where we can Maintain our Source code and Automation Framework or CRS or Builds in one place
2. It’s a Cloud based Repository which is used to maintain the source code. we have to create an Account in order to use it

Git

1. It’s a Software which should be installed in client machine which is used to communicate to GitHub

E-Git

1. It’s a default Plugin Available in Eclipse Latest versions

Gitbash

:-It’s a Software which is used to communicate to GitHub using command prompt

Advantages of GitHub

1. File Sharing between the team members is easy
2. Since its a Cloud Base it provides Remote Access that means contributors from anywhere can access and contribute their work
3. It provides an option to Review the code or Test Script

Procedure to Push a project to GitHub

1. Create a Local Repository, create a new Repository in GitHub and copy the URL of it inside your system by Navigating to Eclipse and search for Git
2. Repositories, their click on create New Repository it will create Repository inside your system, once it’s done Right click on your
3. project > Team > Commit here Drag and Drop all your changes need to push to GitHub and write a Commit Message and then click on
4. Push and Commit > Next > Next then it will ask for Username and Password for this. in the eclipse in the place of password paste created token
5. > Next >Next again it will ask for Username and Password, enter Username and Password click on Ok, it will be pushed to GitHub

To Create Tokens

Click on profile>click on settings>Search for Developer Settings > Personal Access Tokens > Generate New Token > Select the options > Click on generate New Token > Copy the token and

paste it somewhere in the system

Procedure to Pull the Project

1. Go to GitHub, Copy the URL of the project Which you want to Pull
2. Open Eclipse > File > Import > Git > Clone URL >Paste the URL > Next > Next > Finish

**Jenkins**

1. Jenkins is an open source continues Integration or Continuous Deployment and Delivery Tool (CI/CD Tool) Written in Java Programming Language its used to

implement CI/CD Works it’s also called Pipelines

How to install Jenkins

1. Open the Browser, Type Jenkins Download, click on first link i.e. Official link(Jenkins.io)
2. Once you go there, there will be download Jenkins tables, their click on generic java package. War file
3. Once you clicked on that, it will start downloading War file
4. Once its downloaded Copy the file of jar and open command prompt and write java-jar-Paste the path
5. If Jenkins. War is not included in that path add Jenkins. War
6. Click on enter for the first time when your running, it will give you Password, wait for the message Jenkins is Fully up and Running
7. By default, Jenkins will run in 8080 Port so to open Jenkins open the Browser > Type Local Host:8080 > Click on install suggested plugins
8. Once its Downloaded, it will ask for user creation, them enter valid Username and all the Details, click on Ok, it will open Jenkins Homepage

Jenkins executions:

There are 3 types if executions:

1. Force execution by using build now
2. Build Scheduling
3. Jenkins git integration

Build now:

1. In Jenkins Click on new Item, enter an Item Name > Select a Free Time Project > Ok, then it will ask for configuration
2. In the Configuration Enter Description and Click on First Advanced Button > Select Custom Workspace, It will ask for path of your project their copy paste your Project Path(without pom.xml), scroll down and click on add Build Step > invoke top level Maven Targets, here enter the goals like test, clean Etc. > click on Advance Under POM write POM.xml > apply > Save > Build Now, if you click on this your project will Triger the Test case and it will start Running, once the Test Case is Run Click on the Project to see the Console Output.

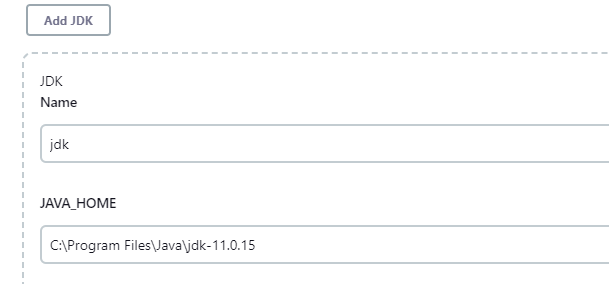
Build scheduling:

1. In Jenkins Click on new Item, enter an Item Name > Select a Free Time Project > Ok, then it will ask for configuration
2. In the Configuration Enter Description and Click on First Advanced Button > Select Custom Workspace, it will ask for path of your project their copy paste your Project Path, scroll down and click on build triggers> select periodically> schedule the execution by giving crone pattern (to understand better click on ‘?’ this symbol ) >build steps> execute windows batch command and in command text area enter mvn test>save> build> after the time which you have mentioned the build will start automatically to run.

Jenkins git integration:

1. In order to achieve this, first we should have some plugins that is git plugin and maven integration plugin. To install this in dashboard> manage Jenkins>manage plugins>available>search for the plugins.
2. We should have git exe file, Jdk, and maven in our system and we have to provide the same path for Jenkins.
   1. To download git exe open the browser> enter git download>click on first link from git,.scm.com> click on windows> in standalone installer click one window set up links based on your system configuration> it will start downloading.
   2. To add these to Jenkins.

Manage Jenkins>global tool configuration> click on Jdk and copy the path of Jdk in your system and paste. (without bin)



* 1. Add git path as shown below

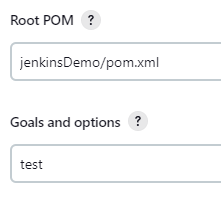


* 1. Add maven as shown below



* 1. Click on save and apply.

1. In order to achieve git Jenkins integration the pre-condition is to have project in GitHub.
2. Dashboard>new item>maven project> add description>source code management >git> provide git url> provide GitHub credentials (un, pwd>build triggers>poll scm (here provide the cron pattern to check your git changes >build> provide the path where your pom.xml file is present in git> in goals and options enter the command which you want to execute> save



1. If you configure this, for each and every commit Jenkins will execute the test cases automatically.

Advantages of Jenkins

1. Jenkins is Free and Open Source so that can be extended using Plugin's
2. Jenkins is Easy to install Manage and Troubleshoot issues
3. Since Jenkins is developed in Java so it is Platform Independent

Disadvantages of Jenkins

1. Jenkins is Easy to Install but Difficult to Implement
2. Jenkins Installation and Configuration is Time Consuming
3. Managing of Jenkins Dashboard when we have to Many Jobs or Projects is Hard

**Interview Question**

1. Explain your Framework
2. Where Collection concepts included in Selenium
3. Can you explain Java concepts Used in your Framework?
4. Can you tell me about Maven Build Lifecycle?
5. Can you Name the Dependencies used in your Framework?
6. Which CI/CD Tool you are using in your Project
7. Which Unit Testing Tool Your using in your Project
8. Which Build Management or Build Development Tool your using in your Project
9. Which Cloud Repository or Version Control Tool Your Using