```
Framework Interview Questions
==============================
1.What is mean by POM?
======================
*Page Object Model.
*POM is an object repository design pattern in selenium webdriver.
*POM creates our testing code maintainable and reusable.
*Page factory is an optimized way to create object repository.

2.What are the types of POM?
============================
*With page factory.
*WithOut page factory.

3.What is meant by Annotations?
================================
*In the Java computer programming language, an annotation is a form of
syntactic metadata that can be added to Java source code.
*Classes, methods, variables, parameters and packages may be annotated.
*Like Javadoc tags, Java annotations can be read from source files.

4.What are the page factory Annotations?
========================================
*@FindBy
*@FindBys
*@FindAll
*@CacheLookUp

5.What is mean by pojo class?
==============================
*POJO stands for Plain Old Java Object, and would be used to describe the
same things as a "Normal Class" whereas a JavaBean follows a set of rules.
*Most commonly Beans use getters and setters to protect their member
variables, which are typically set to private and have a no-argument
constructor.

6.How will you generate the pojo class?
======================================
*RightClick-->Source-->Generate-->Getter&Setter-->SelectAll.

7.Write the folder Structure used in POM?
=========================================
*Src/Main/java-->Used to store the locators.
*Src/Test/java-->Used to execute the business logic.
*Src/Resource/java-->Used to store the reusable code.

8.What is the difference between @FindBys and @FindAll?
=======================================================
@FindBys
--------
*List<WebElement> with both FindBy WebElement will contain any WebElement
which statifies both criteria.
@FindAll
--------
*List<WebElement> with either any one of the FindBy WebElement will contain
any WebElement which statifies any one of the criteria.

9.What is the use of POM?
==========================
*Object repository is independent of test cases.
```

*Code becomes less and optimized because of the reusable page methods in the POM classes.
*Methods get more realistic names which can be easily mapped with the operation happening in UI.

10.What are the use of @cacheLookup?
====================================
*If we don’t do it, then every time when we turn to our element,WebDriver will check if the element is present on the page.

11.What is JUnit?
=================
*It is an open-source testing framework for java programmers.
*The java programmer can create test cases and test his/her own code.
*It is one of the unit testing framework. Current version is junit 4.
*The org.junit package contains many interfaces and classes.

12.What is Unit Testing?
========================
*JUnit is a unit testing framework for the Java programming language.
*JUnit has been important in the development of test-driven development, and is one of a family of unit testing frameworks which is collectively known as xUnit that originated with SUnit.

13.What are all the annotations available in JUnit?
===
*@BeforeClass
*@Before
*@AfterClass
*@After
*@Test

14.Explain what is ignore test in JUnit?
==
*A test method annotated with @Ignore will not be executed.
*If a test class is annotated with @Ignore, then none of its test methods will be executed.

15.Explain the test execution order?
====================================
*@BeforeClass
*@Before
*@Test
*@After
*@AfterClass

16.What is the difference between @BeforeClass and @Before?
===
@BeforeClass
----
*This annotation specifies that method will be invoked only once, before starting all the tests.
@Before
----
*This annotation specifies that method will be invoked before each test.

17.What is the difference between @AfterClass and @After?
===
@AfterClass
----

*This annotation specifies that method will be invoked only once, after finishing all the tests.
@After
------
*This annotation specifies that method will be invoked after each test.

18.Can we able to use main method in JUnit?
==
*No, we can't able to use main method in JUnit.

19.Is it possible to generate report using JUnit?
==
*No,it is not possible to generate report using JUnit.

20.What is the use of annotations?
==================================
*In Selenium WebDriver if you want to run test script using JUnit framework we have to add few JUnit Annotations in Selenium WebDriver Test script.

21.What is the use of @Test annotation?
==
*When we Run the script as JUnit then all the methods specified below @Test are executed using JUnit Test.
*As @Test JUnit annotation is specified before the testUntitled() method, this method will be executed when we run the "GoogleSearch" class using JUnit.

22.What does Assert class?
==========================
*A set of assertion methods useful for writing tests.
*Only failed assertions are recorded.
*These methods can be used directly: Assert.assertEquals(...), they read better if they are referenced through static import.

23.What is the difference between Assert and verify?
===
Assert
----
*When an "assert" command fails, the test execution will be aborted.
*The assert command is used when the end result of the check value should pass to continue to the next step.
*If the assert condition is true then the program control will execute the next test step but if the condition is false, the execution will stop and further test step will not be executed.
Verify
------
*When a "verify" command fails, the test will continue executing and logging the failure.
*The Verify command is used to check non-critical things.
*There wont be any halt in the test execution even though the verify condition is true or false.

24.What are all the methods available in assert class?
==
*assertEquals()
*assertTrue()
*assertFalse()
*assertNull()
*assertNotNull()
*assertSame()
*assertNotSame()

*fail()

25.Is it possible to group the test cases?
==
*Yes,it is possible to group the test cases using @RunWith annotation.

26.Is it possible to re-run the failed test cases in JUnit? if yes, How will you do?
==
=========
*No,it is not possible to re-run the failed test case in JUnit.

27.Can we change return type of JUnit test method from void to some other type?
==
=====
*No,we can't change the return type of JUnit test method from void to some other type.

28.How will you handle exception in JUnit?
===
*try-catch idiom
*With JUnit rule
*With annotation

29.How will you handle exception @test annotation?
===
*@Test (expected = IllegalArgumentException.class)

30.Which Is the latest version of JUnit?
==
*4.12

31.How will you set the timeout in test cases?
===
*By using Timeout parameter on @Test Annotation (applies to test method).
*Otherwise we can use Timeout parameter on @Test Annotation (applies to test method).

32.What are different assertions supported by JUnit?
==
*assertEquals()
*assertTrue()
*assertFalse()
*assertNull()
*assertNotNull()
*assertSame()
*assertNotSame()
*fail()

33.How to create and run JUnit test suite for selenium WebDriver?
===
*To create test suite, Right click on junitpack package folder and Go to -> New -> Other -> Java -> Junit -> Select 'JUnit Test Suite'.

34.For what purpose, assertTrue and assertFalse assertions are used?
==
*If you want to test the boolean conditions (true or false), we can use assertTrue and assertFalse assertions.

35.Can you give me example of JUnit assertEquals assertion?

==
```
 *String string1="Junit";
  String string2="Junit";
  Assert.assertEquals(string1,string2);
```

36.What is cucumber?
====================
*Cucumber is a tool that supports Behavior Driven Development (BDD).
*It offers a way to write tests that anybody can understand, regardless of their technical knowledge.
*In BDD, users first write scenarios or acceptance tests that describes the behavior of the system from the customer's perspective, for review and sign-off by the product owners before developers write their codes.

37.What language is used by Cucumber?
=====================================
*Gherkin language is used by cucumber.
*Gherkin language uses several keywords to describe the behavior of application such as Feature, Scenario, Scenario Outline, Given, When, Then etc.

38.What is meant by a feature file?
===================================
*A Feature File is an entry point to the Cucumber tests.
*A feature file must provide a high-level description of an Application Under Test.
*The first line of the feature file must start with the keyword 'Feature'
*A feature file can contain a scenario or can contain many scenarios in a single feature file but it usually contains a list of scenarios.

39.What is the purpose of Scenario Outline in Cucumber?
===
*Scenario outline is a way of parameterization of scenarios.
*We can pass both Scenario level data as well as step level data.
*It uses the keyword Example
*This scenario outline does iteration in the Data table.

40.What is the purpose of Step Definition file in Cucumber?
===
*Steps definition file stores the mapping between each step of the scenario defined in the feature file with a code of function to be executed.
*when Cucumber executes a step of the scenario mentioned in the feature file, it scans the step definition file and figures out which function is to be called.

41.What are the major advantages of Cucumber framework?
===
*The client can easily understand because of gherkin languages.
*We can easily found the step which will be failed.
*Style of writing tests allow for easier reuse of code in the tests.
*Quick and easy set up and execution.
*Efficient tool for testing.

42.What is the limit for the maximum number of scenarios that can be included in the feature file?
===
*A feature file can contain a maximum of 10 scenarios, but the number can vary from project to project and from one organization to another.

43.What symbol is used for parameterization in Cucumber?

===
*Pipe symbol (|) is used to specify one or more parameter values in a feature file.

44.What is the purpose of Examples keyword in Cucumber?
==
*Examples keyword is used to specify values for each parameter used in the scenario.
*Scenario Outline keyword must always be followed by the keyword Examples.

45.What is the purpose of Cucumber Options tag?
===
*Cucumber Options tag is used to provide a link between the feature files and step definition files.
*Each step of the feature file is mapped to a corresponding method on the step definition file.
*Example:@CucumberOptions(features="Features",glue={"StepDefinition"})

46.What is the meaning of TestRunner class in Cucumber?
===
*TestRunner class is used to provide the link between feature file and step definition file.
*A TestRunner class is generally an empty class with no class definition.

47.What is the starting point of execution for feature files?
===
* When integrated with Selenium, the starting point of execution must be from TestRunner class.

48.What is the use of glue property under Cucumber Options tag?
===
*Glue property is used to let Cucumber framework identify the location of step definition files.

49.What is the maximum number of steps that are to be written within a scenario?
==
=====
*The maximum number of steps to be written in a scenario is 3-4 steps.

50.What are the difference between Jbehave and Cucumber?
==
*Although Cucumber and Jbehave are meant for the same purpose, acceptance tests are completely different frameworks
*Jbehave is Java based and Cucumber is Ruby based
*Jbehave are based on stories while Cucumber is based on features

51.What is the difference between BDD and TDD?
===
BDD
---
*Behavior centered development process.
*BDD tests are written in readable format using Given-When-Then steps.
*BDD tests are readable by non-programmers.
TDD
---
*Test centered development process.
*TDD tests are written using programming languages like Ruby, JAVA etc.
*TDD tests are difficult to read by non-programmers.

52.What are the two files required to run a cucumber test?

==
The 2 files required to execute a Cucumber test scenario are
*Features
*Step Definition

53.What is the two main purpose of using Gherkin?
===
*The test is written in plain English which is common to all the domains of your project team.
*This test is structured that makes it capable of being read in an automated way. There by creating automation tests at the same time while describing the scenario.

54.What are the keywords used in Feature file?
===
*Feature
*Background
*Scenario
*Scenario Outline
*Given
*When
*Then
*And
*But

55.What is the difference between Given, When, Then steps in feature file?
===
*Given defines the context of the scenario.
*When defines the actions of the scenario.
*Then defines the outcome of the scenario.

56.Explain background in feature file?
==
*Steps written under the Background keyword are executed before every scenario.
*If you want to execute the same steps for every scenario like login to the website, you just write those common steps under the background keyword.
*While executing every scenario, steps written under background will be executed first.

57.What Is Cucumber tag,plugin,glue,monochrome,strict,feature, Dry Run?
==
*Feature - path to feature file.
*Glue - path to step definition.
*dryRun - boolean value - check for missing step definition.
*tags - used to group cucumber scenarios in the feature file.
*strict - boolean value - fail the execution if there is a missing step.
*monochrome - boolean value - display console output in a readable way.

58.How To Generate Cucumber Execution Reports?
==
*format = {"pretty", "html:target/Destination"}
*format={"json:target/Destination/cucumber.json"}

59.How To Run A Particular Scenario From A Feature File ?
===
*@tag before the scenario but while in Test Runner file when given this tag it is running entire feature file.
*tags={"@Islamic_User_check"}

60.What are @before, @after hooks?
==
*Cucumber Hooks allows us to better manage the code workflow and helps us to reduce the code redundancy.
*@before hook gets executed well before any other test scenario, and @after hook gets executed after executing the scenario.

61.How to Run Cucumber tests in parallel?
==
*A common approach for running Cucumber features in parallel is to create a suite of Cucumber runners, one for each suite of tests you wish to run in parallel.
*For maximum parallelism, there should be a runner per feature file.
*This is a pain to maintain and not very DRY.

62.What is TestNG?
==================
*TestNG is a testing framework designed to simplify a broad range of testing needs, from unit testing to integration testing.

63.What is the difference between JUnit and TestNG?
==
JUnit:
----
*@BeforeClass and @AfterClass methods have to be declared as static. TestNG does not have this constraint.
*Group test is not possible.
*Dependencies test is not possible.
*Paramaeterized object we can't pass.
TestNG:
----
*It has provided four additional setup/teardown pairs for the suite, test and groups, i.e. @BeforeSuite, @AfterSuite, @BeforeTest, @AfterTest, @BeforeGroups and @AfterGroups, @BeforeMethod, @BeforeClass, @AfterClass and @AfterMethod.
*Group test is possible.
*Dependencies test is possible.
*Paramaeterized object we can pass.

64.What is the difference between Jbehave and Cucumber?
===
Jbehave
----
*JBehave is a pure Java framework.
*Only supports stories, not features.
Cucumber
----
*Cucumber is based on Ruby.
*Supports features.

65.What are all the annotations available in TestNG?
==
*@BeforeTest
*@AfterTest
*@BeforeClass
*@AfterClass
*@BeforeMethod
*@AfterMethod
*@BeforeSuite
*@AfterSuite
*@BeforeGroups

*@AfterGroups
*@Test

66.Explain the Execution order in TestNG?
==
@BeforeSuite
@BeforeTest
@BeforeClass
@BeforeMethod
@Test case 1
@AfterMethod
@BeforeMethod
@Test case 2
@AfterMethod
@AfterClass
@AfterTest
@AfterSuite

67.How will you prioritize the test case?
===
*We use priority attribute to the @Test annotations.
*In case priority is not set then the test scripts execute in alphabetical order.

68.If you set two methods same priority means, which method execute first?
==
*Testng considers the alphabetical order of the method names whose priority is same.

69.What is the default method priority?
=======================================
*If you don't mention the priority, it will take all the test cases as "priority=0" and execute.

70.What is the latest TestNG version?
=====================================
*6.14.3

71.What is TestNG suite?
===========================
*A test suite is a collection of test cases intended to test a behavior or a set of behaviors of software program.
*In TestNG, we cannot define a suite in testing source code, but it is represented by one XML file, as suite is the feature of execution.
*It also allows flexible configuration of the tests to be run.

72.Why we are using TestNG.xml file?
=======================================
*Testng.xml file allows to include or exclude the execution of test methods and test groups.
*It allows to pass parameters to the test cases.
*Allows to add group dependencies.
*Allows to add priorities to the test cases.
*Allows to configure parallel execution of test cases.
*Allows to parameterize the test cases.

73.How will you group the test cases in testng suite?
==
*Groups are specified in your testng.xml file using the <groups> tag.
*It can be found either under the <test> or <suite> tag.

*Groups specified in the <suite> tag apply to all the <test> tags underneath.

74.How will you pass the value using parameter?
==
*Name attribute represents the parameter name
*value represents the value of that parameter.
*Example:<parameter name="browser" value="firefox"/>

75.What is @optional annotation?
=================================
*If defined parameter is not found in your testng.xml file, The test method will receive the default value which is specified inside the @Optional annotation.

76.What is the use of @dependsOnMethod annotation?
==
*If one @Test method fails or skipped from execution then It's dependent @Test method must not be executed.

77.What is the use of @dependsOnGroup annotation?
===
*If one @Test group fails or skipped from execution then It's dependent @Test groups must not be executed.

78.What is TestNG Assert and list out common TestNG Assertions?
===
*TestNG Asserts help us to verify the condition of the test in the middle of the test run.
*Based on the TestNG Assertions, we will consider a successful test only if it is completed the test run without throwing any exception.
Some of the common assertions supported by TestNG are:
----
*assertEqual(String actual,String expected)
*assertEqual(String actual,String expected, String message)
*assertEquals(boolean actual,boolean expected)
*assertTrue(condition)
*assertTrue(condition, message)
*assertFalse(condition)
*assertFalse(condition, message)

79.What is Hard Assert in TestNG?
==================================
*Hard Assert throws an AssertException immediately when an assert statement fails and test suite continues with next @Test.

80.What is Soft Assert in TestNG?
==================================
*Soft Assert collects errors during @Test.
*Soft Assert does not throw an exception when an assert fails and would continue with the next step after the assert statement.

81.How to run test cases in parallel using TestNG?
===
*we can use “parallel” attribute in testng.xml to accomplish parallel test execution in TestNG.
*All the test cases inside <test> tag of testng.xml file will run parallel.

82.How to exclude a particular test method from a test case execution?
===
*By adding the exclude tag in the testng.xml

```
  <classes>
  <class name="TestCaseName">
     <methods>
       <exclude name="TestMethodNameToExclude"/>
     </methods>
  </class>
  </classes>
```

83.How to disable a test case in TestNG ?
==
*To disable the test case we use the parameter enabled = false to the @Test annotation.
*Example::@Test(enabled = false)

84.What are the different ways to produce customize reports for TestNG results?
==
There can be two ways we can customize TestNG report
*Using ITestListener Interface:
*Using IReporter Interface:

85.What is the use of @Listener annotation in TestNG?
===
*TestNG listeners are used to configure reports and logging.
*One of the most widely used listeners in testNG is ITestListener interface.
*It has methods like onTestStart, onTestSuccess, onTestFailure, onTestSkipped etc.
*We should implement this interface creating a listener class of our own.
*Next we should add the listeners annotation (@Listeners) in the Class which was created.

86.What is the use of @Test(invocationCount=x)?
==
*The invocationcount attribute tells how many times TestNG should run a test method
*Example:@Test(invocationCount = 10)

87.What is the use of @Test(threadPoolSize=x)?
==
*The threadPoolSize attribute tells to form a thread pool to run the test method through multiple threads.
*This attribute is ignored if invocationCount is not specified.
*Example:@Test(threadPoolSize = 3)

88.What are @Factory and @DataProvider annotation?
==
@Factory:
----
*A factory will execute all the test methods present inside a test class using a separate instance of the respective class with different set of data.
@DataProvider:
----
*A test method that uses DataProvider will be executed the specific methods multiple number of times based on the data provided by the DataProvider.
*The test method will be executed using the same instance of the test class to which the test method belongs.

89.How to re-run the failed test cases?

==
*After the first run of an automated test run. Right click on Project –
Click on Refresh.
*A folder will be generated named “test-output” folder. Inside “test-
output” folder, you could find “testng-failed.xml”
*Run “testng-failed.xml” to execute the failed test cases again.

90.What is mean by DataDriven?
==============================
*It is used to read /write the data in excel sheet.

91.What are the format available in Excel to read?
===
*POI
*jxl

92.What is mean by Workbook?Whether Workbook is an Interface or class?
===
*The Whole Excel sheet is called as workbook.
*Workbook is an Interface.

93.What is the purpose of getCellType()?
===
*It is used to get the particular cell type and show Whether it is a
integer or numerical.

94.What are the possible output given by getCellType()?
==
*0
*1

95.What method is used to replace the cell value?
==
*setCellType()

96.What is the difference between .xls and .xlsx?
==
.Xls
-----
*It is used before 2007.
*It will support only 512 (2^9) rows and 65536 (2^16) column.
*It will not support macros.
*colors are limited.
*It will support onlu xls format.
.xlsx
----
*It is used recently
*It will support up to 2^14 rows and 2^20column.
*It will support macros.
*colors are unlimited.
*It will support onlu xls and xlsx format.

97.What is the difference between jxl and apache POI jar file?
==
jxl:
----
*It support only .xls file.
POI:
----
*It support both .Xls and .Xlsx

98.How will you write data in Workbook?
==
*write() method is available.