1. Tell me about yourself

- First of all, I would like to appreciate you for giving me this opportunity.
- I am **XXXXXXX**, I have more than 2 years of experience in IT industry as a manual tester and test automation engineer.
- I've gained domain knowledge in e-commerce web applications in Agile environment.
- I have much experience in Front-end UI testing and Back-end testing like Database and API.
- I was proficient in various types of testing such as Functional testing, Smoke testing, Regression testing and Integration testing.
- I've experienced Performance testing using JMeter.
- I have experienced generating extend reports and logs using Log4j.
- I've experienced executing all test suites automatically in headless mode using Jenkins.
- I am good at several test automation tools including Java, Selenium WebDriver, TestNG, and Cucumber.
- I've gained strong knowledge in various types of management system including Confluence, Jira, and Bitbucket.
- I started my career as a manual tester, after few months became test automation engineer.
- I was the team lead in my last project. So I developed Automation framework from scratch.

In terms of soft skills:

- .....

2. Tell me about your last project?

In my previous project, I worked on eCommerce application manual testing and test automation.
The project is a very complex eCommerce application. The application has an:
admin dashboard module,
sales module,
catalog module,
customers module,
promotions module,
newsletters module,
content management module,
reporting module, and
system management module.
The front-End UI testing had 7 users. One of them is customer user, five of them are manager users such as customer manager, store manager, catalog manager, marketing manager, and sales manager. The last one is admin user. Our team lead was responsible for testing admin user.
I also did back-end database testing using MySQL and API testing using Postman.
In database testing we had reading database access, in the first step we prepared valid queries scripts for getting data from database. Then added, updated or deleted information from UI as requirements, at last tested if the contents were in the database correctly. I used Connection,

Statement and ResultSet objects in database testing framework.
In API testing at first I tried all requests in Postman manually for checking if all urls and body structures are valid. Then I started to automate in Java. In API testing I used JSONObject, RestAssured, RequestSpecification, and Response objects.

I have written more than 200 test cases and conducted tests before and after the feature and patch release.

In the project, I used Jira for bug tracking and automation development tasks, Confluence for requirement and design documentation, BitBucket and Git for code repository management, and Jenkins for build and nightly test execution.

In addition to doing functional testing, I have also done API performance testing using Apache JMeter. That is all about my project.

3. Can you describe your daily activity?

• When I start my work, first of all, I check result report of Smoke test from Jenkins server. If something goes wrong, I will send out an email to my team for letting them aware of the result.
• Then I check my email if there are any important tasks or notices, and check my schedule if there are any meetings for the day and also check Jira to review what needs to be done that day in which priority.

• Then I go to attend daily standup meeting with my scrum team to talk about what we did yesterday, what we will do today and are there any roadblock in our way. This meeting takes about 15 minutes.

• After that, I go back to my work. First, I clarify the problems in result report from Jenkins if there is any, and sort them out. If I found any bug or defect, I create a bug ticket in Jira, and assign it to developers team. Then I start automating test cases from regression suits according to their priorities in Jira.

4. Why should we hire you?

- First of all, I think my experience and technical expertise at test automation is the exact match to your job description.
- Second, I am a quick learner, and someone who is eager to learn. I always research the newest technical skills in my field, and very keen to learn and apply them in my work as soon as possible. In terms of this, I believe I can be the most productive and supportive team worker in your company.
- Finally, I am a creative problem solver, and I possess very strong logical and critical thinking skills. And I have much experience in managing teams over 10 years. So I can attribute many effective suggestions and solutions, they will add a lot of values and benefits to the company.

5. Describe yourself in 3 words

I would describe myself as Technical-astute, Analytical and Resourceful.

- My TECHNICAL abilities and experience are strong match for this job description, so I am confident that I have the skills and qualities that needed to make a positive difference quickly in the role.
- My ANALYTICAL qualities mean I can always use my knowledge to find, assess and resolve problems quickly.
- Finally, my RECOURSEFULNESS means I have an abundance of options that I can use to deal with any type of technical situation. So you don't have to worry about my duties.

6. How do you handle conflict?

- Nothing is personal at work. Everyone has the same goal and wants to get job done successfully. Of course, I can do the things which are most helpful to my company. So, I would try to communicate with him/her and let each other understand our concerns. I believe that effective communication is key to resolving conflicts.

7. What do you do in case you have too much work?

- I am a multitasker, but I prefer to handle one project at a time. In my long-term management experience in my life, I developed a strong ability to prioritise tasks. So If I have too much work, I prioritise them first, and then start to deal with them according to their priority.
  However, in work, the reality is that sometimes you need to be able to accomplish multiple things at a time. At that time, I can deal with several tasks at the same time

by focusing on one task while keeping track of others, and I have never feel nervous and crumble under pressure.

8. What do you do in case you cannot finish your work before deadline/you can not meet the deadline?

- You don't just suddenly miss a deadline unless you have an emergency. There are telltale signs that you won't be able to meet your deadline. In this case, I give an advance notice to everyone who will be affected by the delay, and I also give a brief explanation of why I am missing the deadline.
- Then I will explain what I am doing to get the work done and the amount of time that I need.
- If the time isn't really enough because of the emergency, I may also need some help from my team to reach the goal on time.

9. What's your weakness?

I am a very detail-oriented person. So if, occasionally, when my prioritisation is not well enough, I waste a lot of time on unnecessary details.

10. How do you handle stress?/Can you work under pressure?

I believe that pressure plays very important role to get job done successfully. It helps me to focus more on my tasks and be more productive.

In terms of managing stress, my first approach is calm down and try to react to situations, rather than stress. By

this way, the situation can be handled and I feel less stressed.
I also handle stress and pressure by keeping myself fit and healthy. I eat well, sleep well and visit gym three times a week. They help me to have the best concentration levels when I am at work.

11. Where do you see yourself in next 5 years?
I have a mid-term goal for 3 years. I hope I will do a very good job and can be a QA team-lead in 3 years.
In terms of technical skills, I am someone who is eager to learn. So I'll learn the newest technical skills in QA field as much as possible, and become a senior level QA engineer.
Also I have been learning Front-End developing for 6 months. This also helps me to reach my goals earlier.

12. What do you think a person need to be a good QA?
- Positive Attitude
- Technical knowledge and skills
- Analytical thinking
- Team player
- Strong communication skills
- Prioritising
- Quality-oriented
- Detail-Oriented
- Lifelong-learner
- End-user perspective

13. Tell me one mistake you did.

When I started-up my first job, I underestimated my effort for testing a new feature, so the overall workflow was longer than I expected. And I was just about behind the schedule. Then I talked to the manager, he helped me out and gave me some tips on how to better estimate. After that I became much better on estimation, never run into this kind of issue.

14. Tell me the most challenging part of your job?

Some pages of applications always or occasionally load so slowly. In this case, if we are not allowed to use Sleep method in our framework, the both Explicit and Implicit wait are not working for the page loading until the element we want presents on the page. And we always get “No such element exception” or “timeout exception.” We even face with the situation that many other Page Load functions are also not working at all.

15. Tell me about your framework?

…..

16. **Describe the last bug you found in your last project?**

……

SDLC&STLC

1. **What is software testing?**
   It's a process of verifying a software in order to check if it has any bugs, so that to build bug free and high quality application.
2. **What is Software Requirements Specification/SRS?**
   • A software requirements specification is a document/ detailed description of a software or application between the customer and the supplier.
   • It contains all the requirements that end-user expected.
   • SRS document is prepared by Business Analyst.
   • Developer and tester examine the application based on SRS document.
3. **What is SDLC/Software Development Life Cycle?**
   SDLC is a process of creating high-quality software or application. It includes the following phases:
   • Project Planning
   • Requirement Gathering
   • Design (How the application will be built)
   • Coding (by developers team)
   • Testing (by QA team)
   • Deployment (Releasing product)
4. **What is STLC/Software Testing Life Cycle?**
   STLC is a process of testing software or application (in

order to build bug free and high-quality application). It includes the following phases:

• Requirements analysis

• Test Planning (test plan, test estimation, test schedule)

• Test Designing (test case, test script, test data)

• Test Environment Setup

• Test Execution (test result, defect reports)

• Test Reporting (final test result, test metrics, test closure report)

Testing Metrics *are the quantitative measures used to estimate the progress, quality, productivity and health of the software testing process.*

5. **What is the difference between STLC and SDLC?**
   STLC is one of the part and a very important phase of SDLC. A software cannot be released without passing through the STLC process. STLC is also necessary for post-release and update cycle.
6. **What is requirement?**
   Requirement is the expectation of clients and users for the software or application.
7. **When should the testing start?**
   It should start right after the Requirement Gathering Phase of SDLC. QA engineers examine the SRS document carefully, and prepare their test data even before the Coding phase.
8. **How to tell if the requirement is good or bad?**
   Requirement must be SMART:

   • **S**pecific

   • **M**easurable

- **A**ttainable
- **R**ealistic
- **T**estable

9. **Why do we test?**
   - To build bug free application.
   - To satisfy end user and client.
   - To build great product to generate more efficiency.
10. **What is tester's main responsibility?**
    - To find bugs as much as possible, as early as possible.
    - To make sure most of the bugs get fixed. As a result, satisfy the end-user and client (by delivering bug free application).
11. **Is 100% testing possible?**
    The application is dynamic and developing step by step, so every new functionality that is added to the application will require testing.
    Also, there are unlimited scenarios that we can't even imagine. So we can't test the application 100%.
12. **What is the testing hierarchy/testing pyramid/testing phases/testing stages?**
    Unit testing: Developers test each module or functionality of the software during development.
    Integration Testing: Check the data flow from one module to others. When the modules integrate each other, make sure they all still work.
    System Testing: That's End-to-End testing, we test everything from beginning to end.
    Acceptance Testing: It's a test that done by customer or another QA team to confirm whether the software has met the required specifications or not.

13. **How Many Environments Do You Have?**
    1. Development environment;
    2. QA/Test environment;
    3. Pre-production/staging environment;
    4. Production environment;

    QA environment is setup by Developers for the test engineers.
14. **What is Black box testing?**
    Black box testing is a test that you perform when you have no idea how the software you are testing was programmed.
15. **What is White box testing?**
    White box testing is a test that you perform when you have a deep understanding or general knowledge about the architecture of the software.
16. **What is Functional Testing?**
    To verify that each function of the software/application works according to the requirement and specification. In short, Functional testing verifies what software does.

    Examples: Unit Testing, Smoke Testing, Regression Testing, Integration Testing.
17. **What is Non-Functional Testing?**
    To check non-functional aspects of a software/ application such as performance, usability, reliability, security, speed etc.

    In short, Non-functional testing verifies the way software works and how well it works.

    Examples: Performance Testing, Load Testing, Stress

Testing, Volume Testing, Security Testing, Upgrade & Installation Tests, Recovery Testing.

18. **What is Unit Testing?**
    - A unit is the smallest testable part of an application.
    - Unit test is done by developers for eliminating the bug before testers find it.
    - In unit test developers test each module or functionality of the software during development.
    - Normally Testers don't perform Unit Tests, every developer is responsible for their own Unit Test.
19. **What is Component Testing?**
    - Component testing refers to test each module or functionality separately after the whole software is developed.
    - It is performed by testing team.

20. **What is Integration Testing?**
    …
21. **What is Acceptance Testing?**
    …
22. **What is Regression Testing?**
    …
23. **What is Smoke Testing?**
    Smoke Testing refers to verify the critical functionalities of software are working fine.
24. **What's Requirement Analysis (in STLC)?**
    - To analyse and study the requirements and try to find out whether the requirements are testable or not.
    - To identify the scope of testing. If any feature is not

testable, team members plan a strategy for it.

• To assign user stories to testers by team lead.

25. **What is Test plan and what does it include**?

    Test plan is a detailed document of how the test will proceed, It is, usually, written by team lead.

    It includes:

    Project information like project name, project lead, application name, modules, QA lead, execution date etc.

    Also it includes:

    Purpose and Background, Test strategy, Requirements, Feature to be tested, Feature not to be tested, Entrance and Exist criteria, Testing type, Test environment, Traceability of the Test, Resources, Test execution (Date), Test artefacts, Defect report, Defect severity, Test schedule, Acceptance criteria, Test cases.

26. **What is Test case and what does it include**?

    Test Case is a documentation which describes the functionality and test steps. Test case is written by test engineers.

    In Jira Test case includes:

    Test description, Test priority, Test environment, Issue, Sprint, Department, Module, T-shirt Size.

    In Confluence Test case includes:

    Project information, Precondition, Postcondition, Dependencies, Resource, Test users, Bug tracking system, Test execution, Detailed test steps, Expected result, Actual result.

27. **What is a Test Scenario?**

    Test Scenario refers any functionality that can be tested. It is a collective set of test cases.

    Test Scenario gives a high-level idea of what we need to test.

    For an eCommerce Application, a few test scenarios would be:

    Test Scenario 1: Check the Search Functionality

    Test Scenario 2: Check the Payments Functionality

    Test Scenario 3: Check the Login Functionality

28. **What is Environment Setup?**

    QA/Testing environment setup refers software and hardware setup for the testing teams to execute test cases.

    Testing environment setup is done by developer, and developers should push their codes from dev environment to testing environment.

29. **What is Test Execution?**

    Test Execution refers to carry out the testing manually or automatically based on the test plans and the test cases.

30. **What is Test Closure Activity?**

    Test closure activity involves several activities like final test reporting, test matrices and test results.

    It performs at the end of the testing process. It is usually performed after the product is delivered.

31. **What is a defect?**

    When the expected result does not match the actual result, it is a defect.

32. **What is The Severity of the Bug?**

    Severity is degree of impact that a defect has on the software or application.

    The QA engineer usually determines the severity level of defects.

    Defect Severity Levels:

    Critical: A defect that blocks the functionality of an application completely. When it happens, the user or the tester is unable to proceed or test anything.

    Major (High): A highly severe defect that collapses the system. However, certain parts of the system still working.

    Minor: A defect that has an impact on the less important data and minor functions of the software.

    Low (Trivial): A defect that does not relate to the data or working of the software. It relates to the look and feel of the program.

33. **What is The Priority of the Bug?**

    Priority refers to in which order a defect should be fixed.

    Developers or Product Owner determines the priority level of defects.

    Defect Priority Levels:

    High: It is a defect that must be resolved as soon as possible. Because it affects the system severely and cannot be used until it is fixed.

    Medium: It is a defect that doesn’t require urgent resolution and can be fixed during the usual course of development, for example, during the next sprint.

Low: It is a defect that isn't serious, so it can be resolved after the high and medium defects are fixed.

34. **What does Bug Report include?**
    It should includes:
    ID/name
    Description/summary
    Environment (browser, operating system, zoom level and screen size)
    Source URL
    Visual proof
    Steps reproduce
    Expected Result
    Actual Result
    Attachments
35. **What do you do when you find a defect?**
    If I find a defect, before reporting it I reproduce the bug to make sure it is a valid defect.
    If I am not sure if it is a bug or not, I will talk to someone who is on the higher level.
    If it is a bug, I assign a bug ticket to developer.
36. **What did you do if developer said the bug you found is not a bug?**
    First of all, I recheck SRS documents, and make sure that I was right about the bug, also make sure that I provided all the documents including screenshot picture that why I think this is a bug. If the developer still insist it's not a bug, I will speak to someone who is on the higher level, it might be project manager, or business analyst.

37. **What will you do when the script fails/automation test fails?**
    First of all, I rerun the failed tests and look deep into the report and try to identify the failure wether it is an application error, script issue or environment problem.
    After that, I will resolve the problem according to its type.
38. **What is the difference between debugging and testing?**
    Debugging is the process of fixing a bug in the software. It refers the process of identifying, analysing and removing errors.
    Testing refers to find errors rather than to fix them. When a tester finds a bug, they usually report it, so that a developer can fix it.

Agile

1. **What is Waterfall Methodology?**
   The Waterfall methodology is a sequential development process that project development team only moves to next phase of development or testing if the previous step completed successfully.
2. **What is Agile Methodology?**
   Agile is a process that breaking the project into several stages and at every stage we need continuous cooperation and improvement.
3. **What are the core values of Agile?**
   1) Individuals and Interactions over Process and tools.
   2) Working Software over Comprehensive Documentation.
   3) Customer Collaboration over Contract Negotiation.
   4) Responding to change over Following a plan.
4. **What is Scrum?**
   Scrum is one of the implementations of agile methodology that allows us to rapidly and repeatedly inspect actual working software in every two weeks to one month.
5. **What does Ceremony include in Agile Framework?**
   It includes:
   Sprint Planning Meeting
   Sprint Review Meeting
   Sprint Retro Meeting
   Daily Scrum Meeting
   Grooming Session Meeting

6. **What is Sprint Planning Meeting?**
   Sprint planning meeting is attended by the product owner, Scrum Master and the entire Scrum team. During the sprint planning meeting, the product owner describes the highest priority features to the team.
   Sprint planning meeting is designed to answer the questions, What can be delivered in the sprint? And how will we accomplish that work?
7. **What is Sprint Review Meeting?**
   Usually it happens at the end of every one or two Sprints. Usually everyone will join this meeting inludes customer, stakeholders. It runs by the developer.
   During this meeting, the Scrum team shows what they accomplished during the sprint. Typically all new functionalities of the application will be shown.
   Automation engineer can DEMO their automation script as well.
8. **What is Daily Stand up (Daily Scrum) Meeting?**
   It's a meeting that holds everyday by scrum team.
   Usually it focuses on 3 Main Questions: What did you do yesterday? What will you do today? Do you have any blocker (impediment)?
   It takes around 15 mins.
9. **What is Sprint Retro Meeting?**
   It's a meeting that holds by scrum team after every sprint.
   In this meeting the scrum team discuss what went well during the previous sprint cycle and what can be improved for the next sprint.

It's an opportunity for the Scrum Team to inspect and improve itself.

10. **What is Epic?**
    Epic is the biggest user story which is almost everything about one project.
11. **What is Feature in Agile?**
    Feature is a collection of user stories.
12. **What is a User Story?**
    Use story is a short, simple description of a requirement.
13. **What is an Acceptance Criteria?**
    Acceptance Criteria is Conditions that a software must satisfy in order to be accepted by a user, customer or other stakeholder.
14. **Tell me about your Agile experience in your recent project:**
    ❖ Before each Sprint, we have our Sprint Planning Meeting:
    • We discuss the priority of features.
    • We learn the specific part of the application that we are going to work on.
    • We choose our stories based on **Velocity** and **Capacity**:
        ▪ **Velocity**: a measure of how much work an agile team can deliver on average in a sprint.
    It takes a view at measuring:
    - How much work an agile team has delivered in the past sprints.
    - How long it took the team to get the work done.
    For example: if team planned 30 story points (Business

value); worth of user stories in a sprint and able to deliver as planned then team's velocity is 30

- **Capacity**: Total number of available hours for a sprint.

Calculate holiday and PTO hours.

• This meeting is held every week and lasts for almost 1 hour. We get a general idea then we do a Sprint Grooming meeting for giving some estimates for the tasks.

- Team, SM, and PO get together to ensure work items are relevant and useful.
- Ask questions to P.O. of user stories
- Re-define acceptance criteria
- Writing Stories
- Breaking Epics Into User Stories
- Understand the story to give proper estimation/ to prevent under/over estimate.

❖ How Do You Estimate?

• Based on my experience and complexity of the story and it is something I worked on before.

• After sprint starts, we do Daily Stand up Meeting

• Every morning we discuss what we did yesterday, what we will do today and any blocker.

❖ End of the sprint, we usually do a Sprint Demo/ Review Meeting.

• It is just to show customer what we build sprint (PO can put feedback)

• As an SDET in my team, I have done presentations sometimes and go over the functionalities in the conference room.

• Clients or stakeholders or business people ask questions that they don't know.

❖ After the Sprint Demo, we do a Sprint Retrospective Meeting.

• In Sprint Retro, we talk about what was good in the last sprint, what kind of mistakes we made.

• We go over them and make sure that we don't make the same mistakes again.

• If we did something good and improvements, we would continue doing it.

• This meeting is held at the sprint review meeting or at the end of the sprint; it lasts for 2-3 hours.

15. **What is a Test Script?**
    Test script is code that can be run automatically to perform a test on a user interface.
16. **What is Requirement Traceability Matrix (RTM)?**
    RTM is a document that details the technical requirements for a given test scenario and its current state.
17. A

CI/CD

1. **What is CI/CD?**
   • CI/CD is a shortened term for Continuous Integration and Continuous Delivery.

   • CI/CD is the combination of principles, practices, and capabilities that allow for software changes of all kinds to get users in a quick, repeatable, and safe manner.

2. **What is continuous Integration?**
   CI is an automated process that requires developers to integrate code and its changes into a shared repository frequently.
3. **What is continuous Delivery?**
   To deploy code changes automatically into staging environment.
4. **What is continuous deployment?**
   To deploy code changes automatically into production environment safely, quickly and continuously.
5. **What is CI/CD Pipeline?**
   Pipeline is a set of processes that take the code from version control and deploy to production automatically.
6. **What is Jenkins?**
   Jenkins is Continuous Integration and Deployment tool.
   Jenkins has 3 components:
   1. Code changing;
   2. Testing;
   3. Deployment;
7. **What is Jenkins job?**
   Jenkins job is the heart of *Jenkins's build* process. It can be considered as a task that runs automatically in *Jenkins.*
8. **How to create a job in Jenkins?**
   Step 1) Login to Jenkins
   Step 2) Create New Item
   Step 3) Enter Item details
   Step 4) Enter Project details
   Step 5) Enter repository URL

Step 6) Setup Build section
Step 7) the project
Step 8) Build Source code
Step 9) Check the status
Step 10) See the console output

SELENIUM

1. **What is test automation?**
   • Test automation is automating the testing process using specific softwares.
   • It requires few sets of skills and tools like Selenium, Cypress, Appium, Cucumber, TestNG, Serenity, UFT, Protractor, etc.
2. **What are the advantages of selenium?**

1) Selenium is open source and free to use.
2) It supports multiple languages like Java, Ruby, Python, C#...
3) It supports many operating systems like Windows, Mac, Linux.
4) It supports multi-browser testing.
5) It has a good amount of resources and helping community .
6) It interacts with the web application.

3. **What are the advantages of selenium?**

   1) Selenium supports only web-based applications, does not support computer desktop applications, Mobile application.
   2) No built-in reporting tool, it needs third party tools

for report generation activity.

3) Cannot work with graphics, captchas, barcodes, shapes.

4) It does not support file upload facility.

5) Hard to synchronize.

4. **What functionalities cannot be automated using Selenium**?

   • Images, Videos, Captchas, QR or Bar Codes, Shapes, etc.

   • Broken functionalities: if the function is not working, you cannot automate it.

   • 3rd party applications in the webpage: For example, we will not be able to handle GoogleMaps in the webpage since it is a 3rd party application.

   • Test cases related to look and feeling of the webpage, because they are not about functionality.

5. **What types of testing can you automate with Selenium?**

   • Functional Tests (positive/negative, UI)
   • Smoke Tests
   • Regression Tests
   • Integration Tests
   • System Testing
   • End-to-end Testing

6. **What are the test types we don't automate with Selenium?**

   • Non-functinal testings like Performance, Stress, Load testing.
   • Unit testing based on the look and feel of webpage.
   • Static testing

7. **What is in the Selenium tool set?**

   • **Selenium IDE:** implemented as a Chrome and Firefox extension, and allows you to record, edit, and debug tests.

   • **Selenium WebDriver:** execute your tests against different browsers

   • **Selenium GRID:** run your tests on different machines against different browsers in parallel.

   • **Selenium RC:** to write automated web application UI tests in any programming language.

8. **What is Implicit Wait?**

   Implicit wait is used to tell the web driver to wait for a certain amount of time before it throws a "No Such Element Exception". The default setting is 0. Once we set the time, the web driver will wait for the element for that time before throwing an exception.

   Implicit Wait syntax:

   driver.manage().timeouts().implicitlyWait(TimeOut, TimeUnit.SECONDS);

9. **What is Explicit Wait?**

   Explicit wait is used to tell the web driver to wait for certain conditions or specified maximum time before throwing the `ElementNotVisibleException` .

   Explicit Wait syntax:

   WebDriverWait wait = new WebDriverWait (WebDriverRefrence, TimeOut);

10. **What is Fluent Wait?**

    Fluent wait refers to try to find element again and again until it find it or until the final timer runs out.

    Fluent Wait syntax:

    Wait<WebDriver>wait=new FluentWait<Webdriver>(driver)

.withTimeout(5,timeUnit.seconds).pollingEvery(100,timeunit.milliseconds)
.ignoring(NoSuchElementException.class)

11. **What is Xpath?**

    • Xpath is used to find the location of any element on a webpage using html structure based on specific syntax.

    • **Syntax**:

    //tagname[@attribute=`value`]

    • Absolute xpath starts with single slash (/), starting from root element and all the way to the element.

    • Relative xpath starts with double slash (//), starting selection matching anywhere in the document.

12. **How do you handle dynamic elements?**

13. • We can use advanced syntax and keywords for handling dynamic elements, such as **Parent, Ancestor, Sibling, Contains, Text, Startwith…**

14. **What is the difference between xpath and css selector?**

    • Xpath can search elements backward or forward, CSS works only in forward direction.

    • Xpath can work with text, CSS can not work.

    • Xpath has more combination and can search by index. CSS can not search by index, but CSS is working faster than xpath.

15. **What kind of Data Driven Frameworks are used in Selenium?**

    Data Driven Framework in Selenium is separate data sets that is fetched for test cases and suites. It comes from external and internal files.

    • External files are like Excel, .csv, .xml or some database tables.

• Internal files are like Property files, TestNG Data Provider.

16. **What do we do if we get NoSuchElementException?**
    • Check if locator is correct.
    • Check if timing out is enough.
    • Check if element is hidden inside an iframe.
17. **What is the difference between driver.get() and driver.navigateto() ?**
    • driver.get(): To open an URL and it will wait till the whole page gets loaded.
    • driver.navigate(): To navigate to an URL and it will not wait till the whole page gets loaded.
18. **What is JavascriptExecutor in Selenium?**
    JavascriptExecutor is an interface that is used to execute JavaScript with Selenium. It helps us to have more control over webpages in certain conditions.