👨‍💼 Interviewer – Question 1

Introduce yourself and explain your project.

✅ Ideal Answer (Speak like this):

"My name is **XYZ**, and I have around **x years of experience as a QA Engineer**.
I have worked on **manual and automation testing** for web applications using **Selenium with Java, TestNG**, and **API testing using Rest Assured**.
I follow **Agile Scrum methodology** and participate in all ceremonies.

My project was a **web-based enterprise application** used for customer onboarding, transactions, and reporting.
I was involved in **requirement analysis, test case design, automation framework development, regression execution, CI/CD runs, and defect tracking**."

👨‍💼 Interviewer – Question 2

Explain your automation framework.

✅ Ideal Answer:

"I worked on a **Hybrid Automation Framework** which is a combination of **Page Object Model, TestNG, and Data-Driven approach**.

📌 In my framework:

- Page classes contain **locators and actions**
- Test classes contain **test logic only**
- Utility classes handle **common methods like waits, clicks, screenshots**
- TestNG handles **execution, grouping, parallel runs, and reports**

This framework helped us maintain clean code, reuse logic, and easily scale automation."

👨‍💼 Interviewer – Question 3

Which framework do you prefer – Cucumber or TestNG Hybrid? Why?

✅ Ideal Answer:

"I prefer **TestNG with POM Hybrid framework**.

📌 Reasons:

- Faster execution compared to Cucumber
- Easier debugging (no step-definition overhead)
- Less maintenance
- Better control over execution and retries

In my project, business did not require Gherkin syntax, so TestNG Hybrid was more efficient and stable."

👨‍💼 Interviewer – Question 4

What are the limitations of Cucumber?

✅ Ideal Answer:

"Cucumber has some limitations:

- Slower execution
- High maintenance for feature files
- Step definition duplication
- Debugging is difficult due to extra layers

In large projects, managing feature files becomes complex, which is why we preferred TestNG Hybrid."

👨‍💼 Interviewer – Question 5

Explain WebDriver.

✅ Ideal Answer:

"WebDriver is a **Selenium interface** used to automate web browsers.

```
WebDriver driver = new ChromeDriver();
```

- WebDriver is an interface
- ChromeDriver is the implementation

WebDriver provides methods to open browsers, locate elements, perform actions, and validate results."

🕵️ Interviewer – Question 6

Explain Data Driven testing in TestNG.

✅ Ideal Answer:

"Data-driven testing means executing the same test case with **multiple sets of data**.

📌 In TestNG, we use `@DataProvider`.

```
@DataProvider
public Object[][] data() {
 return new Object[][] {
  {"user1","pass1"},
  {"user2","pass2"}
 };
}
```

I used this mainly for login, registration, and negative test scenarios."

🕵️ Interviewer – Question 7

Explain execution order: BeforeTest, BeforeMethod, AfterMethod, AfterTest.

✅ Ideal Answer:

"The execution order is:

1️⃣ @BeforeTest
2️⃣ @BeforeMethod
3️⃣ @Test
4️⃣ @AfterMethod
5️⃣ @AfterTest

📌 In real time:

- BeforeTest → Browser setup
- BeforeMethod → Login
- AfterMethod → Logout
- AfterTest → Close browser”

🧑‍💼 Interviewer – Question 8

Explain Parallel Execution.

✅ Ideal Answer:

“Parallel execution allows multiple test cases to run at the same time using multiple threads.

```
<suite parallel="tests" thread-count="3">
```

We used this in regression suites to reduce execution time in CI/CD pipelines.”

🧑‍💼 Interviewer – Question 9

How do you group test cases?

✅ Ideal Answer:

“Grouping allows us to execute specific sets of test cases like smoke, sanity, and regression.

```
@Test(groups="smoke")
```

We use smoke tests after deployment and regression before release.”

🧑‍💼 Interviewer – Question 10

What is the return type of getWindowHandles()?

✅ Ideal Answer:

“The return type is **Set<String>** because window handles are unique and duplicates are not allowed.”

🧑‍💼 Interviewer – Question 11

Explain Hashing (Correctly).

✅ Ideal Answer (Senior Level):

“Hashing is a technique used to **store and retrieve data quickly** using a key-value pair.

📌 Java uses hashing in **HashMap**.

```
HashMap<String,String> data = new HashMap<>();
data.put("username","admin");
```

Internally, HashMap uses a **hash function** to convert keys into hash codes, which decides where the data is stored in memory.

📌 In my framework, I used HashMap to store:

- Test data
- Environment details
- API response values

This avoids looping and improves performance.”

🧑‍💼 Interviewer – Question 12

Explain WebDriver Architecture (Internal Working).

✅ Ideal Answer (Clear Flow):

“WebDriver follows a **client–server architecture**.

📌 Flow:
1️⃣ Test script (Java code)
2️⃣ Selenium WebDriver API

3️⃣ Browser Driver (ChromeDriver)
4️⃣ Actual Browser

Browser driver acts as a bridge between Selenium and the browser.
This explains why driver version compatibility is important."

👨‍💼 Interviewer – Question 13

Which collections have you used in your framework?

✅ Ideal Answer:

"I have used:

- List → dropdown values
- Set → window handles
- Map → test data and API responses"

👨‍💼 Interviewer – Question 14

Which is better for modification: ArrayList or LinkedList?

✅ Ideal Answer:

"In most cases, **ArrayList** is preferred because it provides faster access and less memory overhead.
In real projects, ArrayList is more commonly used."

👨‍💼 Interviewer – Question 15

Explain iteration in collections.

✅ Ideal Answer:

"Iteration means looping through collection elements.

```
for(String value : list) {
```

```
 System.out.println(value);
}
```

Used to validate dropdowns and API response lists.”

👮 Interviewer – Question 16

Dynamic dropdown with 500 values – will you loop?

✅ Ideal Answer (VERY IMPORTANT):

“No, looping through all 500 values is not the best approach.

First, I analyze the dropdown behavior:

- If search box exists → type & select
- If auto-suggestion → loop filtered results
- If lazy loading → scroll & load
- Looping all values is my **last option only**”

(You already mastered this answer 👌)

👮 Interviewer – Question 17

Explain CI/CD pipeline.

✅ Ideal Answer:

“CI/CD automates build, test execution, and deployment.

📌 Flow:
Code Push → Build → Automation Tests → Report → Deployment

We used CI/CD for nightly regression and release validation.”

👮 Interviewer – Question 18

Which Selenium exception was difficult to handle?

✅ Ideal Answer:

“StaleElementReferenceException.

It occurs when DOM refreshes and element reference is lost.

📌 Handling:

- Re-locate element
- Use waits
- Retry logic”

🕵️ Interviewer – Question 19

Explain components of REST API / Rest Assured.

✅ Ideal Answer:

“REST API components are:

- Request
- Headers
- Body
- Response
- Status code

📌 Using Rest Assured:

```
given().when().get("/users").then().statusCode(200);
```