**Top 18 Appium Interview Questions & Answers**

**1) Explain what is Appium?**

Appium is a freely distributed open source mobile application UI[Testing](https://www.guru99.com/software-testing.html)framework.

**2) List out the Appium abilities?**

Appium abilities are

* Test Web
* Provides cross-platform for Native and Hybrid mobile automation
* Support JSON wire protocol
* It does not require recompilation of App
* Support automation test on physical device as well as similar or emulator both
* It has no dependency on mobile device

**3) List out the pre-requisite to use APPIUM?**

Pre-requisite to use APPIUM is

* ANDROID SDK
* JDK
* TestNG
* Eclipse
* Selenium Server JAR
* Webdriver Language Binding Library
* APPIUM for Windows
* APK App Info On Google Play
* js

**4) List out the limitations of using Appium?**

* Appium does not support testing of Android Version lower than 4.2 or lower than API level 17
* Limited support for hybrid app testing. E.g., not possible to test the switching action of application from the web app to native and vice-versa
* No support to run Appium Inspector on Microsoft Windows

**15.What is Appium’s most considerable limitation?**

It would surely be: **multiple session handling.**Since Appium is a server, it serves HTTP requests; you might have two different computers running a test each against the same Appium server: what happens? As for now, Appium does not support this scenario and the second test will be aborted. This is a considerable limitation, because no queuing system comes with Appium. If you need to support multiple sessions, you will need to implement this feature by yourself.

**5) Explain how to find DOM element or xPath in a mobile application?**

To find the DOM element use "UIAutomateviewer" to find DOM element for Android application.

**6) Explain the design concept of Appium?**

* Appium is an "HTTP Server" written using[Node.js](https://www.guru99.com/node-js-tutorial.html)platform and drives iOS and Android session using Webdriver JSON wire protocol. Hence, before initializing the Appium Server, Node.js must be pre-installed on the system
* When Appium is downloaded and installed, then a server is setup on our machine that exposes a REST API
* It receives connection and command request from the client and execute that command on mobile devices (Android / iOS)
* It responds back with HTTP responses. Again, to execute this request, it uses the mobile test automation frameworks to drive the user interface of the apps. Framework like
  + Apple Instruments for iOS (Instruments are available only in Xcode 3.0 or later with OS X v10.5 and later)
  + Google UIAutomator for Android API level 16 or higher
  + Selendroid for Android API level 15 or less

**7) What language does Appium support?**

Appium support any language that support HTTP request like Java,[JavaScript](https://www.guru99.com/interactive-javascript-tutorials.html)with Node.js, Python, Ruby, PHP, Perl, etc.

**8) Explain the pros and cons of Appium?**

**Pros**:

* For programmer irrespective of the platform, he is automating ( Android or iOS) all the complexities will remain under single Appium server
* It opens the door to cross-platform mobile testing which means the same test would work on multiple platforms
* Appium does not require extra components in your App to make it automation friendly
* It can automate Hybrid, Web and Native mobile applications

**Cons**:

* Running scripts on multiple iOS simulators at the same time is possible with Appium
* It uses UIAutomator for Android Automation which supports only Android SDK platform, API 16 or higher and to support the older API's they have used another open source library called Selendroid

**9) Explain what is APPIUM INSPECTOR?**

Similar to[Selenium](https://www.guru99.com/selenium-tutorial.html)IDE record and Playback tool, Appium has an "Inspector" to record and playback. It records and plays native application behavior by inspecting DOM and generates the test scripts in any desired language. However, Appium Inspector does not support Windows and use UIAutomator viewer in its option.

**10) Mention what are the basic requirement for writing Appium tests?**

For writing Appium tests you require,

* **Driver Client**: Appium drives mobile applications as though it were a user. Using a client library you write your Appium tests which wrap your test steps and sends to the Appium server over HTTP.
* **Appium Session**: You have to first initialize a session, as such Appium test takes place in the session. Once the Automation is done for one session, it can be ended and wait for another session
* **Desired Capabilities**: To initialize an Appium session you need to define certain parameters known as "desired capabilities" like PlatformName, PlatformVersion, Device Name and so on. It specifies the kind of automation one requires from the Appium server.
* **Driver Commands**: You can write your test steps using a large and expressive vocabulary of commands.

**11) Mention what are the possible errors one might encounter using Appium**?

The possible errors one might face in Appium includes

* Error 1: The following desired capabilities are needed but not provided: Device Name, platformName
* Error 2: Could not find adb. Please set the ANDROID\_HOME environment variable with the Android SDK root directory path
* Error 3: openqa.selenium.SessionNotCreatedException: A new session could not be created
* Error 4: How to find DOM element or[XPath](https://www.guru99.com/xpath-selenium.html)in a mobile application?

**12) Do you need a server machine to run tests on Appium**?

No, you don't need server machine to run tests on Appium. Appium facilitates a 2-tier architecture where a test machine connects to a test server running Appium and automating the whole thing. You can have Appium running on the same machine where your test runs.

**13) Is it possible to interact with my apps using Javascript while I am testing with Appium?**

Yes, it is possible to interact with App while using Javascript. When the commands run on Appium, the server will send the script to your app wrapped into an anonymous function to be executed.

**14) Mention what are the most difficult scenarios to test with Appium?**

The most difficult scenario to test with Appium is data exchange and multiple session handling.

**15) While using Appium can I run my tests in a multithreaded environment?**

Yes, you can run the test in a multithreaded environment, but you have to ensure that no more than one test runs at the same time against the same Appium server.

**16) In Android, do you need an app's .apk to automate using Appium or you also need app in my workspace?**

In Android, you only need .apk file to automate using Appium.

**17) Explain what is Appium package master? How to create package?**

Appium package master is a set of tools manage and create appium packages. For example to create package you can use the code

# using es7/babe1

Gulp create-package –n <package-name>

#regular es5

Gulp create-package ---nobabe1 –n <package-name>

The package will be generated in the out/<package-name>

**18) Explain how test frameworks are supported by Appium?**

Appium does not support test framework as such there is no need to support them. Appium can be used with any frameworks you want.

### **Q-3: How Many Types Of Apps Do You Know?**

### **Answer.**

Mostly there are three kinds of apps in use.

* **Native apps** are those which are built using the iOS/Android SDK.
* **Mobile web apps** are the ones which are getting accessed via the mobile browsers.
* **Hybrid apps** fall into the category which wraps around a <webview>. It’s a native control to enable interaction with the web elements.

### **Q-7: What Would You Prefer To Test On Real Devices Or Use Simulators/Emulators?**

### **Answer.**

It’s one of the most commonly asked Appium interview questions. You’ve to be little logical and practical while answering it. Don’t just simply answer “It would depend on what you need.”. Because it would be a layman answer which interview won’t expect from you. Rather you should explain it by example.

You can say something like that it’s always best to test on real devices. As it would allow you to catch errors that you may not detect otherwise. But you have to configure the device smartly with the Appium server so that it can detect the device. Sometimes the ADB, the Android debugger may disconnect from the device even if it remains plugged in. And it can cause your tests to fail. To handle such issues, you can write a module which resets the ADB after some time to re-connect the devices.

### **Q-8: How Do The Tests Run On Simulators/Emulators?**

### **Answer.**

The limitation of disconnecting from the devices would never occur with the emulators. They will remain attached to the ADB. You may even set orientation and other hardware-based features that it can emulate.

Running entirely in a virtual world wouldn’t come without some limitations. Your tests would run slow as the emulators consume a lot of memory. Even they might behave intermittently during the execution.

But you can start early testing with them till you acquire the real devices.

### **Q-11: What Are The Essentials For Writing The Appium Tests?**

### **Answer.**

There few things that you need before starting to write test cases in Appium.

* **Driver –**It provides a set of APIs to write your Appium tests. It then sends the commands to HTTP server for processing.
* **Session –**All Appium tests needs a session for execution. So, first of all, you have to create it. Close it after the automation ends.
* **Capabilities –**Like you have to set Chrome driver path to start executing tests in a Chrome browser. Similarly, here you need to define params like **PlatformName, PlatformVersion, DeviceName** and so on.
* **Commands –**These are action words which you use in test steps while you write test cases. Appium has a rich set of commands. Read their syntax and purpose from its online documentation.

### **Q-14: What Are The Probable Errors You Might See While Working With Appium?**

### **Answer.**

Following are the errors you might observe with Appium.

* **Error#1:**Missing desired capabilities e.g. Device Name, PlatformName.
* **Error#2:** Couldn’t locate **ADB**. You may have missed setting the **<ANDROID\_HOME>** environment variable.
* **Error#3:** Selenium exception **<openqa.selenium.SessionNotCreatedException>**. It indicates a failure in creating a new session.
* **Error#4:** Failure in locating a DOM element or determining the XPath.

### **Q-16: Can You Run Multithreaded Tests In Appium?**

### **Answer.**

Yes, you can. Appium doesn’t prevent you running the tests in a multithreaded environment. Just make sure that no more than one test runs at the same time against the same Appium server.

### **Q-20: Does It Require A Server Node To Execute Appium Tests?**

### **Answer.**

No, you don’t need a server node to run tests on Appium. It comprises of a 2-tier architecture where a test node connects to a server running Appium and carries out automation. So, you can have Appium tests running on the same node where your test runs.

### **Q-17: Does It Require An <.Apk> File For Automating Tests In Appium?**

### **Answer.**

Yes, you need the **<.apk>** package to automate using Appium.

### **Q-19: Which Are The Selenium Commands To Use With Appium?**

### **Answer.**

There are a no. of Selenium commands to operate with Appium tool.

* Locate commands using ID or class names.
* Raise events on elements **e.g. Click()**.
* Text commands like **type()**.
* **Get/Set** element properties.
* Commands to run JavaScript.
* Switch context between different web views like switching **<iFrames>** in Selenium Webdriver.
* Commands to manage alert boxes.

**7. What are limitations of**Appium ?

Answer:

Below are few limitations or challenges of Appium:

* It takes lot of initial efforts for configuring appium both on Android and iOS platforms
* Image comparison is not possible using Appium
* Unexpected errors are observed quite often which sometimes takes time for resolution
* Android API level < 17 is not supported by Appium (Selendroid needs to be used otherwise)
* Execution of scripts is slow on iOS platform
* Limited gesture support is available in Appium

**25. Can we do data exchange using Appium?**

* **Answer:** This is one of the weakest point of Appium. The data exchange part is quite a complex task to perform using Appium. Simple data exchange is possible easily using the available Appium and Selenium APIs like writing data in the fields etc. However for complex data exchange such as fetching the state of application or data exchange by complex objects in the web-view of applications, Appium is not much useful for such cases and it’s a better approach to go for manually writing the test cases specific to every platform.

### **What Is The Appium Philosophy?**

**Answer :**

R1. Test the same app you submit to the marketplace  
R2. Write your tests in any language, using any framework  
R3. Use a standard automation specification and API  
R4. Build a large and thriving open-source community effort

**What is Appium Inspector?**

**Answer:**

Appium Inspector is a GUI tool similar to Selenium IDE. It enables the user to check out the hierarchy of the mobile application. Appium Inspector is used for:

1. To understand the element hierarchy
2. To find the xpath, name and value of the objects
3. To record the manual actions

**6. What are the advantages of Appium?**

**Answer:**

Below are few of the main advantages of Appium Automation:

* Due to Json Wire Protocol usage, we can use any programming language
* Testing of Native, Hybrid and Web Apps is possible using appium
* Appium supports multiple frameworks
* It’s an open source and cross-platform tool
* It’s compatible with CI tool like Jenkins as well
* Uses selenium API thus no need to learn from scratch
* Can be used with Emulators as well as real devices

**7. What are limitations of**Appium ?

Answer:

Below are few limitations or challenges of Appium:

* It takes lot of initial efforts for configuring appium both on Android and iOS platforms
* Image comparison is not possible using Appium
* Unexpected errors are observed quite often which sometimes takes time for resolution
* Android API level < 17 is not supported by Appium (Selendroid needs to be used otherwise)
* Execution of scripts is slow on iOS platform
* Limited gesture support is available in Appium

**15. What is fruitstrap. What is its use?**

**Answer:**

Fruitstrap is a third-party tool to deploy the application build on a real iOS devices. For automation, Appium installs the testing application on the iOS device using Fruitstrap.

**17. What is the default port number used for Appium Server?**

**Answer:**

Default port number used for Appium Server is 4723(In my machine it is 4727). You need to mention the same while writing desired capabilities in the initial lines of code.