Software Testing Complete Guide

Custom Search	Search

Top 20 Automation Testing Interview Questions and Answers

Here are the most frequently asked Software Automation testing interview questions with answers.

Test automation plays a very important role in the entire life cycle. Most of the time when want to prepare for an automation testers interview, we focus only on the tool specific question.

We should also consider the fact that learning and knowing the tool is just a mean and it's not the ultimate goal.

So whenever we are **preparing for an automation tester interview**, we have to consider "Automation" as a whole and focus around the framework and the steps involved.

Here I have tried to cite down the questions very much specific to the automation as a whole and not specific to any "tool".

Top 20 Automation testing interview questions

Q #1) What is Automation?

Automation is any action which can reduce human efforts.

Q #2) What all things can you automate?

The right candidates for automation are:

- Regression test suite
- Smoke / Sanity test suite
- Build deployment
- Test data creation
- Automating behind the GUI like testing of APIs and methods

Q #3) How do you identify the test cases which are suitable for automation?

Identify the appropriate test cases for automation is the most important step towards automation.

Q #4) Can you achieve 100% automation?

100% automation would be difficult to achieve because there would be many edge test cases and some cases which are executed seldom. Automating these cases which are not executed that often will not add value to the automated suite.

Q #5) Currently I do not have any automation in place in my project, now I want to implement automation, what would be my steps?

- First identify which type of testing / test cases you want to automate
- Identify the tool
- Design the framework
- Create the utility files and environment files
- Start scripting
- Identify and work on the reporting
- Allocating time for enhancing and maintaining the scripts.

Q #6) How do you decide which tool you have to use?

Concluding which tool is best suitable for the project requires a lots of brainstorming and discussions.

Q #7) Once you identify the tool what would be your next steps?

Once we finalize the tool, our next step would be to design the framework.

Q #8) What is a framework?

A framework is a set of structure of the entire automation suit. It is also a guideline, if followed can result in a structure which is easy to maintain and enhance. These guidelines include:

- Coding standards
- Handling the test data
- Maintaining and handling the elements (object repository in QTP)
- Handling of environment files and properties file
- Reporting of data
- Handling logs

Q #9) What are the attributes of a good framework?

The characteristics are:

- Modular – The framework should be adaptable to change. Testers should be able to modify the scripts as per the environment or login information change
- Reusable – The commonly used methods or utilities should be written in a common file which is accessible to all the scripts.

- Consistent – The suite should be written in a consistent format by following all the accepted coding practices.
- Independent – The scripts should be written in such a way that they are independent of each other. In case one test fails, it should not holdback remaining test cases (unless it is a login page)
- Logger – It is good to have implemented the logging feature in the framework. This would help in case our scripts run for longer hours (say nightly mode), if the script fails at any point of time, having the log file will help us to detect the location and the type of error.
- Reporting – It is good to have reporting feature automatically embedded into the framework. Once the scripting is done, we can have the results and reports sent via an email.
- Integration – Automation framework should be such that it is easy to integrate it with other application like continuous integration or triggering the automated script as soon as the build is deployed.

Q #10) Can you do without a framework?

Frameworks are guidelines and not mandatory rules, so we can do without a framework, but if we create it and follow it, enhancing and maintaining would be easy to implement.

Q #11) What are the different types of automation tool you are aware of?

Open source tool   like Selenium, JMeter

Paid tools like QTP, Load Runner, Ranorex, RFT, and Rational Robot.

Q #12) What generally is the structure of a framework?

Normally the structure should have – (It would differ from project to project)

- A "src" (source) folder having the actual test scripts
- A"lib" (library) folder having all the libraries and common methods
- A "class" folder having all the class file (in case using java)
- A "log" folder having the log file(s)
- A file / folder having all the web element Ids
- A file containing the URL, environment and login information.

Q #13) Where you maintain information like URL, login, password?

This information should always be maintained in a separate file.

Q #14) Why do you want to keep this kind of information in separate file and not directly in code?

URL, Login and Password are the kind of fields which are used very often and these changes as per the environment and authorization. In case we hardcode it into our code, we have to change it in every file which has its reference. In case there are say more than 100 files, then it becomes very difficult to change in all the 100 files and hence can lead to errors. So this kind of information is maintained in a separate file so that updating becomes easy.

Q #15) What are the different types of frameworks?

Different types of [framework](#) available are:

- Keyword driven framework
- Data Driven framework
- Hybrid Framework
- Linear Scripting

Q #16) Can you tell some good coding practices while automation?

Some of the good coding practices include:

- Add appropriate comments
- Identify the reusable methods and write it in separate file
- Follow the language specific coding conventions
- Maintain the test data in a separate file
- Run your scripts regularly

Q #17) Any kind of test which you thing should not be automated?

- Test which are seldom executed
- Exploratory testing
- Usability testing
- Test which are executed fairly quickly when done manually

Q #18) Do you think that testing can be done only at the UI level?

Today as we are moving to Agile mode, testing is not limited to the UI layer. Early feedback is imperial for any agile project. If we concentrate only on the UI layer, we are actually waiting until the UI is developed and available to test. Rather we can test even before the UI is actually developed. We can directly test the APIs or the methods using tools like Cucumber and Fitnesse.

In this way we are giving the feedback much early and even are testing before the UI is developed. Following this approach will help us to test only the GUI aspect like small cosmetic changes or some validations on the UI and will help the developers by giving more time to fix the bugs.

Q #19) How do you select which automation tool is best suited for you?

Selecting the automation tool depends upon various factors like:

- Scope of the application which we want to automate
- Management overhead like cost and budget
- Time to learn and implement the tool
- Type of support available for the tool.
- Limitation of the tool

Q #20) What do you think holds testers back to do automation? Is there a way to overcome it?

The major hurdle for testers is to learn programming / coding when they want to automate. Since testers do not code, adapting to coding is a bit challenging for testers. We can overcome it by:

- Collaborating with developers when automating
- Considering that automation is the responsibility of the whole team and not only of the testers
- Giving a dedicated time and focus on automation.
- Getting proper management support.

You can save these automation testing interview questions as pdf and print for further reading.

Conclusion:

Most of the test automation interview questions are centered on the framework you develop, so it is recommended that you create and understand your test framework thoroughly. When I am interviewing, and the

candidate has answered my question on the framework, I also prefer asking language specific question (core java in my case).

The questions starts from basics of java to writing the logic of some basic scenario like –

- How would you extract a set of text from a given line?
- How would you extract URL?
- In any web page, at any frame, the number of links and its content change dynamically, how would you handle it?
- How do you handle images and flash objects?
- How do you find a word in a line?

Answers to all these automation interview questions are very much specific to the tool / language you are using for automating. So before you go for the interview, brush up your programming skills. In case you did not get a chance to create your framework and someone else have created it, make some time to understand it thoroughly before sitting for the interview.

- Share
- Tweet
- +1
- Pin
- Share

Useful Links!

- Home
- Subscribe for FREE Updates
- Testing RESOURCES
- QA Testing TRAINING
- Write and Earn Program
- FREE eBooks
- ALL Tutorials
- Suggest New Tutorials
- Contact Us

Help & Tutorials

- ISTQB Study Guide
- ISTQB Premium Study Guide
- Free QA Training
- Free Selenium Training
- Free QTP Training
- Free QC Training
- HP LoadRunner Tutorials
- JIRA Tutorials
- 101+ Interview Questions

Follow Us!

follow SoftwareTestingHelp