Top 20 Software Testing Interview Questions.

### 1. What Is Equivalence Partitioning? Explain By Example.

**Answer.**

It’s a famous black box testing technique which defines the effectiveness of manual test cases. You can apply it to a range of testing areas like unit testing, acceptance testing, integration testing and so on. It works by breaking the test data into different sets and selecting one input value from each range.

Since it’s not possible to validate all data points from the domain because any attempt to do so would result in huge no. of test cases. Hence, you should apply this technique in such situations. It classifies the data into different classes where each class lays the input criteria from the equivalence class.

#### Example.

Assuming an application which accepts dates from the calendar year. If we apply the above technique, then we can break the inputs into classes. For example, one for valid dates and another for invalid dates. We’ll now create test cases from each class.

**TC#1.** Valid date class would allow cases with dates from the current calendar year.

**TC#2.** Dates from years other than the current calendar year would belong to the invalid class.

### 2. What Is Boundary Value Analysis? Explain By Example.

**Answer.**

Most of the errors emanate either from the head or the tail ends of the test data. The values at these extreme ends are the boundary values. And the process to analyze them is Boundary value analysis. We may sometimes call it as <range checking>.

It’s one more black box testing approach which focuses on finding errors at the terminating ends of the input domain. You can use this technique at all levels of testing.

While designing test cases using BV, you should consider both valid and invalid boundary values for the edges. Usually, we pick one test case from each boundary.

Let’s see some examples to demonstrate the use of BV analysis.

**TC#1.** The first test case will analyze exact boundary values from the input data. Considering the [example in Q:1](http://www.techbeamers.com/software-testing-interview-questions-manual-testers-part1/#Q:1), it would be the dates from the January and the dates from the December.

**TC#2.** For invalid values, you may consider using the dates from months other than the Jan or Dec.

### 3. What Is State Transition Testing? Explain By Example.

**Answer.**

This approach is best suitable where there is a possibility to view the whole system as a <finite state machine>. It works on the notion that a system can be in a finite no. of distinct states. And it’s the rules of the machine that drive the transitions from one state to another.

It’s the model on which is the basis for the system and the test cases. Any system which produces a different output for the constant input, depending on what has happened before, is a finite state system.

#### Example.

For example, you can consider a bank account. If you use it regularly, it’ll stay active. If you don’t make any transaction for over a period of 12 months, then your account will become inactive. You can though choose to close your account. Once it’s closed, you can not use it until you reopen. So, a bank account can have the following states.

**1.** Open,  
**2.** Inactive,  
**3.** Close,  
**4.** Reopen.

Hence, you can design cases to test all transitions around the states mentioned above. In this model, you measure coverage in terms of switches, see below.

**1.** <0-switch> => You are testing every valid transition.  
**2.** <1-switch> => You’ve covered the pair of two valid transitions.  
**3.** <2-switch> => You need to test the sets of 3 transitions for this coverage.

**Recommended –**[**Top QA Interview Questions and Answers.**](http://www.techbeamers.com/qa-interview-questions-and-answers-top-20/)

### 4. What Is The Difference Between A White Box, Black Box, And Gray Box Testing?

**Answer.**

**4.1. Black box** testing is a testing approach which depends completely on the product requirements and specifications. The knowledge of internal paths, structures, or implementation of the software isn’t a prereq for it.  
**4.2. White box** testing is a testing mechanism which covers the internal paths, code structures, and implementation of the software under test. It generally demands that a tester should possess serious programming skills.  
**4.3.**Last but not the least is the **gray box testing**. It allows looking into the “box” being tested to understand the implementation. Finally, you close the box and apply knowledge to choose more effective black box tests.

### 5. What Are The Categories Of Defects?

**Answer.**

There are three main categories of defects:

**5.1. Wrong:** It indicates a mismatch in the requirement and the implementation. It implies a variance from the given specification.  
**5.2. Missing:** The end product doesn’t have a feature matching the requirement. It’s a variance from the specifications and represents that you didn’t document the requirement properly.  
**5.3. Extra:** You added a feature which the customer didn’t ask for. It’s again a variance from the specification. And the users of the product may like this feature. But it’s still a defect because it’s not the part of the specs.

### 6. What Is The Basis To Prepare For An Acceptance Plan?

**Answer.**

First of all, you need inputs from the following areas to prepare for the acceptance document. These may vary from company to company and from project to project.

**6.1.** **Requirement document:** This document specifies what exactly is needed in the project from the customer’s perspective.  
**6.2.** **Customer Input:** It could be discussions, informal chats, email addresses, etc.  
**6.3.** **Project Plan:** The project plan from the project manager (PM) also serves as a good input to conclude your acceptance test.

### 7. What Is Requirement Traceability Matrix?

**Answer.**

It’s a common Software testing interview question which you might get asked during an interview. The Requirements Traceability Matrix (RTM) is a requirement tracking tool that ensures the requirements are same for the whole development process. There are following reasons to use it:

**7.1.** To determine whether the developed project is meet the requirements of the user.  
**7.2.** To determine all the requirements given by the user.  
**7.3.** To make sure the application requirement can be fulfilled in the verification process.

### 8. Describe How To Perform Risk Analysis During Software Testing?

**Answer.**

Risk analysis is the process of foreseeing the risk in an application and prioritizing them to test. Following are some of the risks:

**8.1-** New Hardware.  
**8.2.** New Technology.  
**8.3.** New Automation Tool.  
**8.4.**The sequence of code delivery.  
**8.5.** Availability of application and test resources.

You can prioritize them into three categories in the following manner:

**i.** **High magnitude:** Impact of the bug on the other functionality of the application.  
**ii.** **Medium:** It can be tolerable in the application but not desirable.  
**iii.** **Low:** It can be bearable. This type of risk has no impact on the company business.

### 9. How To Deal With A Bug Which Is Intermittent And Not Reproducible?

**Answer.**

Again, it’s one of the practical software testing interview questions. A bug might not be reproducible for a variety of reason. Some of these are as follows.

**9.1.** Low memory.  
**9.2.** Addressing an unavailable memory location.  
**9.3.** Things happening in a particular sequence.

### 10. When Do You Perform Decision Table Testing?

**Answer.**

We may use the <decision table testing> for testing systems for which the specification takes the form of rules or cause-effect sequence. In a decision table, the input gets listed in a column, with the outputs in the same column but below the inputs. The remainder of the table examines the different sets of inputs to define the outputs produced.

### 11. What Is Boundary Testing?

**Answer.**

Boundary testing proposes to focus on the limit or edge conditions of the software under test.

### 12. What Is Branch Testing?

**Answer.**

Branch testing requires testing of all the branches in the program source code at least once.

### 13. What Is Breadth Testing?

**Answer.**

It is also known as overview testing. It covers the main functionality of the product but does not test the features in detail.

### 14. What Is Alpha Testing?

**Answer.**

The Alpha Testing takes place at the developer end but in a controlled environment. And it is the end user of the software who does it.

### 15. What Is Beta Testing?

**Answer.**

The purpose of Beta testing is to gather the feedback from the real-time usage of the product. It happens after the installation at the client-end.

### 16. What Is Component Testing?

**Answer.**

This testing focuses on validating the individual components rather than the whole application.

### 17. What Is End-To-End Testing?

**Answer.**

End-to-end testing verifies the complete flow of an application right from start to finish. Its purpose is to identify the system dependencies and ensuring that the data integrity is intact.

### 18. What Is Monkey Testing?

**Answer.**

Monkey testing is a black-box testing technique. It proposes to enter data in any format and verify the software is not crashing. It uses the concept of smart monkey and dumb monkey.

* **Smart Monkey –** Used for load and stress testing and lead to high development cost.
* **Dumb Monkey –** Used for basic testing and helps in locating quality bugs.

### 19. What Is The Difference Between Baseline And Benchmark Testing?

**Answer.**

Following are the primary differences between the baseline and benchmark testing.

* Baseline testing intends to collect the performance of an application. Benchmark testing compares the app performance with the industry standard.
* Baseline testing utilizes the data collected to improve the performance. Benchmark testing returns the information of target application vis a vis other applications.
* Baseline testing compares the current performance with application’s previous performance whereas benchmark testing compares our app performance with competitors performance.

### 20. What Is The Role Of QA In Project Development?

**Answer.**

* QA team is responsible for ensuring the quality of the software product.
* They are involved in planning, testing, and execution.
* QA manager prepares an estimate and agrees on a Quality Assurance plan for the product.
* He explains the QA process to the team members.
* The test engineers ensure the traceability of test cases to requirements.