



*Up-to-date Questions and Answers from authentic resources to improve knowledge and pass the exam at very first attempt. ----- Guaranteed.*



CT-TAE MCQs  
CT-TAE TestPrep  
CT-TAE Study Guide  
CT-TAE Practice Test  
CT-TAE Exam Questions



[killexams.com](http://killexams.com)

**ISTQB**

**CT-TAE**

*Certified Tester Test Automation Engineer*

ORDER FULL VERSION

<https://killexams.com/pass4sure/exam-detail/CT-TAE>



### Question: 11

The GUI of a Customer Relationship Management (CRM) application has been delivered through internet Explorer with proprietary Active X and Java controls. This implementation enables rich client capabilities, but specific commercial automation tools are necessary to automate test cases at GUI of functional test cases. This is to demonstrate whether a small set of the commercial are able to properly recognize actions taken by a tester when interacting with GUI of the CRM application.

Which of the following scripting techniques would be MOST suitable in this scenario?

- A. Data-driven scripting
- B. Keyword-driven scripting
- C. Linear scripting
- D. Structure scripting

**Answer: C**

### Question: 12

What are the four horizontal layers of the gTAA?

- A. Test adaptation, test execution, test design, test definition
- B. Test generation, test execution, test definition, test APIs
- C. Test generation, test definition, test execution, test adaptation
- D. Test definition, test execution, test reporting, test adaptation

**Answer: C**

Explanation:

The gTAA is a reference architecture for test automation systems, and it is organized into four layers to separate different concerns. Each layer has its own responsibilities and interacts with the others to form a complete test automation system. The Test Generation layer is for creating test cases, the Test Definition layer is for defining those cases and their data, the Test Execution layer is for running the test cases, and the Test Adaptation layer handles interactions between the test automation system and the system being tested.

### Question: 13

You have implemented a keyword-driven scripting framework, which uses a test execution tool to run the tests. This

has been in use for the past year and all of the teams now use this framework as the standard approach for test execution.

The teams all work on different aspects of the SUT and they have all experienced significant benefits in the use of this scripting framework. However, on closer examination, you have discovered that there are numerous instances where the teams have the same functionality to test but are using different keywords.

One of your objectives for improvement is to create consistency among the teams.

What is the BEST way to handle this situation?

- A. Move to a model-based approach to scripting where the models include the keywords.
- B. Do nothing, each team are working in isolation and they are all experiencing significant benefits in the way they are currently working.
- C. Provide each team with a set of guidelines and naming conventions for keywords.
- D. Create a central library of keywords and associated definitions for each team to use.

**Answer: D**

Explanation:

By having a shared library, all teams can refer to the same keywords, creating a more consistent and unified scripting approach. This can make it easier to manage and maintain the scripts, especially when different teams need to work together or share resources. It can also reduce the risk of misinterpretations or misunderstandings due to different teams using different keywords for the same functionalities. The other options, while might be valid in some other contexts, do not directly address the issue of consistency among the teams.

**Question: 14**

As a TAE you are evaluating a functional test automation tool that will be for several projects within your organization. The projects require that tool to work effectively and efficiently with SUTs in distributed environments. The test automated tool also needs to interface with other existing test tools (test management tool and defect tracking tool.) The existing test tools subject to planned updates and their interface to the test automated tool may not work properly after these updates.

Which of the following are the two LEAST important concerns related to the evaluation of the test automation in this scenario?

- â Is the test automation tool able to launch processors and execute test cases on multiple machines in different environments?
- â Does the test automation tool support a licensing scheme that allows accessing different sets?
- â Does the test automation tool have a large feature set, but only part of the features will be sets?
- â Do the release notes for the planned updates on existing specify the impacts on their interfaces to other tools?

Does the test automation tool need to install specific libraries that could impact the SUT?

- A. A and C
- B. A and E
- C. B and E

D. C and D

**Answer: D**

### **Question: 15**

You have inherited a TAS that is working well it uses keyword-driven scripting and was well architected. The automation architect who built the system has now moved on to another company. The TAS is working across several projects and has a multiple library of keywords, categorised by project. The individual project teams maintain these keyword scripts.

Based only on the given information, what is the MOST significant risk for the TAS?

- A. The keyword driven scripts may become out of date if not maintained
- B. The level of abstraction, coupled with the departure of the architect may make the system hard to maintain
- C. New projects may not work as well with the TAS as the current projects
- D. Because the keyword scripts are maintained by different teams, there is a likelihood that good coding standards are not followed

**Answer: B**

### **Question: 16**

The Test Automation Manager has asked you to provide a solution for collecting metrics from the TAS that measures code coverage every time the automated regression test pack is run. The metrics must be trend based to ensure that the scope of the regression test pack continues to reflect enhancements made to the SUT - coverage must not drop and should ideally increase. The solution must be as automated as possible to avoid unnecessary manual overheads and errors.

Which of the following approaches would BEST meet these requirements?

- A. Test automation cannot measure code coverage for the SUT, only the code for the automation tools and scripts. The automated test cases would need to be run manually with a code coverage and reporting tool running in the background.
- B. The automated testware would record overall code coverage for each run and add the figure to a new row in a pre-formatted Excel spreadsheet. You would then present the spreadsheet to stakeholders so they could look for changes in coverage.
- C. The automated testware would record overall code coverage for each run, export the data to a pre-formatted Excel spreadsheet that automatically updates a trend analysis bar chart for you to distribute to stakeholders.
- D. The automated testware would record the pass/fail rate of each regression test case, export the data to a pre-formatted Excel spreadsheet that automatically updates a trend analysis success rate bar chart and emails it to stakeholders.

**Answer: C**

### **Question: 17**

You are working as a TAE for a company who have been using a web test execution tool for a number of years. The tool has been used successfully on ten web applications in the past.

The company are developing a new web application which has a friendly User Interface, but the developers have used an object throughout the application which the tool is unable to recognise. As a result, you have no way of capturing the object or verifying the contents using the automation tool.

What is the first thing you should do about this problem?

- A. See if the application can be run on a desktop and if the object can be recognised on the desktop by the tool.
- B. Investigate whether the object can be recognised by other test execution tools in the market
- C. Ask the developers to remove the object and replace it with some text fields
- D. Ask the developers if they can change the object to something that can be recognised by the tool

**Answer: D**

Explanation:

It is common in software testing to encounter situations where test automation tools are unable to recognize certain objects or elements within an application. In such cases, the first step is usually to collaborate with the development team to identify whether the problematic object can be replaced or modified to become recognizable by the automation tool.

This is generally the most efficient and cost-effective approach because it allows you to continue using the same test automation tool without the need for additional training or license costs that may come with switching tools. It also keeps the test environment consistent and doesn't require the redesign of the test cases that are already automated. The other options either involve possibly unnecessary tool or platform switching, or suggesting changes that may not suit the application's requirements.

Of course, if changes to the object or application aren't feasible, then other options such as switching tools or finding workarounds may need to be explored. However, the first step should be to communicate with the developers and investigate a potential quick fix to the issue.

**Question: 18**

Consider a TAS deployed into production. The SUT is a web application and the test suite consists of a set of automated regression tests developed via GUI. A keyword-driven framework has been adopted for automating the regression tests. The tests are based on identification at low-levels of the web page components (e.g class indexes, tab sequence indexes and coordinates) in the next planned release the SUT will be subject to significant corrective maintenance (bug-fixes) and evolution (new features) Maintenance costs to update the test scripts should be as low as possible and the scripts must be highly reusable.

Which of the following statements is most likely to be TRUE?

- A. The keyword-driven framework is not suitable, it would be better to adopt a structured-scripting approach
- B. False positive errors are likely to occur when running the automated tests on the new releases without modifying the test
- C. The total execution time of the automated regression test suite will decrease for each planned release.
- D. The keyword-driven framework introduces a level abstraction that is too high and makes it difficult what really happens

**Answer: B**

**Question: 19**

You are currently designing the TAA of a TAS. You have been asked to adopt an approach for automatically generating and executing test cases from a model that defines the SUT. The SUT is a state-based and event-driven that

is described by a finite-state machine and exposes its functionality via an API. The behavior of the SUT depends on hardware and communication links that can be unreliable.

Which of the following aspects is MOST important when designing the TAA in this scenario?

- A. Looking for tools that allows direct denoting of exceptions and actions depending on the SUT events.
- B. Adopting a test definition strategy based on classification tree coverage for the test definition layer.
- C. Looking for tools that allow performing setup and teardown of the test suites and the SU
- D. Adopting a test definition strategy based on use case/exception case coverage for the definition layer.

**Answer: A**

## Question: 20

You are working on a government system called "Making Tax Digital" or MTD for short. This system is being implemented to stop manual human input error and also to reduce fraudulent behaviour from companies when submitting their tax and VAT returns.

The key concept is that registered companies will need to use government recommended 3rd party software for their accounts and book keeping. These 3rd party applications will have a direct interface into the government's main system for transactions and submissions.

You have been using a test execution tool successfully on the project so far, and have implemented a basic "capture/replay" approach to scripting.

The management have been encouraged with the automation so far, but want the following objectives to be met:

- \* Test cases added easily
- \* Reduction in the amount of scripts and script duplication
- \* Reduction in maintenance costs

Which scripting technique would be MOST suitable in this scenario in order to meet the objectives?

- A. Linear scripting
- B. Structured scripting
- C. Data-driven scripting
- D. Keyword-driven scripting

**Answer: D**

Explanation:

Keyword-driven scripting allows for easier addition of test cases, less script duplication, and reduced maintenance costs which are the objectives outlined by management. This is because in keyword-driven scripting, you design test scripts using a set of predefined keywords, which represent actions that will be executed on the application under test. The keywords and the data associated with them are usually stored in tables (like a spreadsheet), making it easier to add, remove, or modify test cases.

Linear scripting (A) doesn't provide flexibility in terms of reducing script duplication and maintenance costs as it involves writing a linear set of instructions for each test case.

Structured scripting (B) is useful when there is a need to create modular and reusable scripts, but it requires more complexity to implement than keyword-driven scripting and isn't as easily adaptable for adding new test cases.

Data-driven scripting (C) is efficient when you need to test the same functionality with varying inputs and expected results, but it doesn't inherently reduce script duplication or maintenance costs as much as keyword-driven scripting can. It also doesn't directly aid in adding new test cases in an easy manner.

### Question: 21

Consider a SUT that small run on multiple platform during the execution of automated test runs. In each test run an automated test suite needs to be executed, with the same version of the TAF, against the same version of the SUT of each platform. Each platform shall have its own dedicated test environment. Your goal is to implement a process as automated as possible ( i.e with minimal manual intervention) that allows implementing a consistent setup of the TAS across the multiple test environments.

Which two of the following aspects are MOST relevant for achieving your goal in this scenario?

â The configuration of the TAS uses automated installation scripts

â The TAF saves the logs needed to debug errors in XML format

A) Features of the TAF not used by the automated tests have been tested

B) All the automated test cases contain the expected results

C) The TAS components are under configuration management

A. A and e

B. B and c

C. B and d

D. A and d

### Answer: A

### Question: 22

If model-based testing has been selected for the overall test automation approach for a project, how does that influence the layers of the TAA?

A. All layers are used, but the test generation layer will be automated based on the defined model

B. There will be no need for the execution layer

C. No adaptation will be needed because the interfaces will be defined by the model

D. There will be no need to design the tests for the API because those will be covered by the model

### Answer: A

Explanation:

In a model-based testing approach, all layers of the Test Automation Architecture (TAA) are typically used, but the test generation layer will be automated based on the defined model. The model describes the behavior of the system under test (SUT) and is used to automatically generate test cases.

Option B is incorrect because the execution layer is still needed to execute the generated test cases.

Option C is incorrect because the adaptation layer is typically needed to handle interface-related issues between the Test Automation System (TAS) and the System Under Test (SUT), regardless of the model used.

Option D is incorrect because even with a model-based testing approach, designing tests for the API (if APIs are a part of the system) would still be necessary. The model can aid in this, but it does not eliminate the need for test design.

### Question: 23

You are implementing test automation for a project and you want to be able to generate test cases automatically using a series of test design tools which use a variety of test design techniques such as decision tables, pairwise testing and boundary value analysis.

You also want to generate test data automatically which can then be used by the tests.

Initially these tests will be run manually to verify their correctness and ultimately you want to include them in the test execution tool so that they can run unattended.

Which layer of the gTAA will be used to support the specification of the test cases and preparation of the test data?

- A. The generation layer
- B. The definition layer
- C. The execution layer
- D. The adaptation layer

### Answer: B

Explanation:

The definition layer of the Generic Test Automation Architecture (gTAA) is responsible for defining test cases and preparing test data. This layer uses test design tools and techniques, like decision tables, pairwise testing, and boundary value analysis that you mentioned, to define what needs to be tested. This includes the preparation of test data that will be used by the tests.

The generation layer, on the other hand, is responsible for generating the actual test scripts from the defined test cases.

The execution layer is responsible for executing the generated test scripts.

And finally, the adaptation layer is responsible for managing the interaction between the Test Automation System (TAS) and the System Under Test (SUT), which involves handling interface-related issues.

### Question: 24

Designing the System Under Test (SUT) for testability is important for a good test automation approach and can also benefit manual test execution.

Which of the following is NOT a consideration when designing for testability?

- A. Observability: The SUT needs to provide interface that give insight into the system.

- B. Re-useability: The code written for the SUT must be re-useable for other similar system.
- C. Clearly defined architecture: The SUT Architecture needs to provide clear and understandable interfaces giving control and visibility on all test levels.
- D. Control: the SUT needs to provide interfaces that can be used to perform actions on SU

**Answer: B**

### **Question: 25**

You are working on a web-based application called Book Vault that allows people to upload books and order books. This application must be available on all major browsers.

You have been testing the application manually and management have asked you to consider automating some of the tests.

You have investigated a number of commercial and free tools which can automate tests at a web browser level and one tool in particular meets your requirements and you have implemented a trial version.

You have basic programming skills and the main goal is to automate a few functional tests to see if the tool is compatible with the application and can recognise the objects and controls.

Which scripting technique would be MOST suitable in this scenario in order to meet the objectives?

- A. Structured scripting
- B. Capture-replay scripting
- C. Data-driven scripting
- D. Model-based scripting

**Answer: B**

Explanation:

Given that you have basic programming skills and you are in the exploratory phase of the tool, the Capture-replay scripting approach would be the most suitable for this scenario. Capture-replay scripting involves recording a series of user actions as they navigate through the application, and then replaying those actions in an automated way. This technique is typically easy to use, requires minimal programming skills, and can quickly provide a basic level of automation for functional tests. It will also allow you to see if the tool is compatible with your application and can recognize the objects and controls. The other scripting techniques like Structured, Data-driven, and Model-based scripting require a more advanced level of programming skills and a better understanding of the application.

Killexams.com is a leading online platform specializing in high-quality certification exam preparation. Offering a robust suite of tools, including MCQs, practice tests, and advanced test engines, Killexams.com empowers candidates to excel in their certification exams. Discover the key features that make Killexams.com the go-to choice for exam success.



## Exam Questions:

Killexams.com provides exam questions that are experienced in test centers. These questions are updated regularly to ensure they are up-to-date and relevant to the latest exam syllabus. By studying these questions, candidates can familiarize themselves with the content and format of the real exam.

## Exam MCQs:

Killexams.com offers exam MCQs in PDF format. These questions contain a comprehensive collection of questions and answers that cover the exam topics. By using these MCQs, candidate can enhance their knowledge and improve their chances of success in the certification exam.

## Practice Test:

Killexams.com provides practice test through their desktop test engine and online test engine. These practice tests simulate the real exam environment and help candidates assess their readiness for the actual exam. The practice test cover a wide range of questions and enable candidates to identify their strengths and weaknesses.

## Guaranteed Success:

Killexams.com offers a success guarantee with the exam MCQs. Killexams claim that by using this materials, candidates will pass their exams on the first attempt or they will get refund for the purchase price. This guarantee provides assurance and confidence to individuals preparing for certification exam.

## Updated Contents:

Killexams.com regularly updates its question bank of MCQs to ensure that they are current and reflect the latest changes in the exam syllabus. This helps candidates stay up-to-date with the exam content and increases their chances of success.