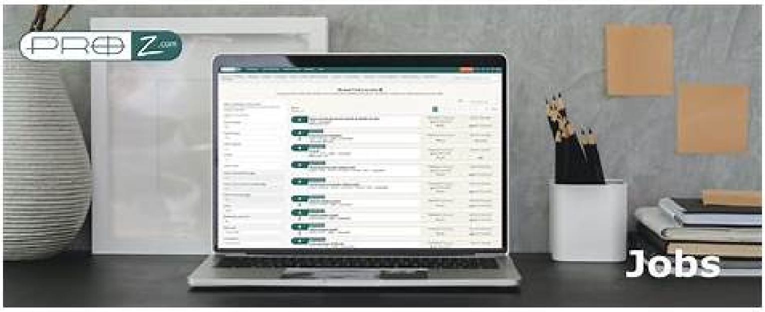
New ISTQB-CTFL Test Pattern | Visual ISTQB-CTFL Cert Test



P.S. Free & New ISTQB-CTFL dumps are available on Google Drive shared by It-Tests: https://drive.google.com/open?id=1bSqJRw-bfxa0vEOVa1uTE4qX2-R-UVDc

The study materials from our company can help you get your certification easily, we believe that you have been unable to hold yourself back to understand our ISTQB Certified Tester Foundation Level (CTFL v4.0) guide torrent, if you use our study materials, it will be very easy for you to save a lot of time. In order to meet the needs of all customers, Our ISTQB-CTFL study torrent has a long-distance aid function. If you feel confused about our ISTQB-CTFL test torrent when you use our products, do not hesitate and send a remote assistance invitation to us for help, we are willing to provide remote assistance for you in the shortest time.

Do you want to pass ISTQB-CTFL exam in one time? It-Tests exists for the purpose of fulfilling your will, and it will be your best choice because it can meet your needs. After you buy our ISTQB-CTFL Dumps, we promise you that we will offer free update service in one year. If you fail the exam, we also promise full refund.

>> New ISTQB-CTFL Test Pattern <<

Visual ISTQB ISTQB-CTFL Cert Test, Book ISTQB-CTFL Free

If you do not receive our ISTQB-CTFL exam questions after purchase, please contact our staff and we will deal with your problem immediately. The download process of ISTQB-CTFL practice engine does not take you a long time. We have some of the best engineers in the industry, and the system they build will guarantee you a smooth download of our ISTQB-CTFL Guide questions. After that, please arrange your own study time. Together with our ISTQB-CTFL practice engine, start your own learning journey.

ISTQB Certified Tester Foundation Level (CTFL v4.0) Sample Questions (Q312-Q317):

NEW QUESTION # 312
For withdrawing money tram an Automated Teller Machine (ATM), the following conditions are required:
- The bank card is valid
- The PIN code is correct
- Money is available in the user's account
The following are some possible interactions between the user and the ATM:
- The entered card is invalid The card is rejected
- The PIN code is wrong The ATM asks for another PIN code
- The requested amount is more than available in the user's account: The ATM asks for another amount
- The requested amount is available in the user's account The ATM dispenses the money Which test design technique should be used to cover all possible combinations of the in put conditions?

- A. Decision table
- B. Equivalence class partitioning
- C. Boundary value analysis
- D. Use case based testing

Answer: A

Explanation:
A decision table is a technique that should be used to cover all possible combinations of input conditions for withdrawing money from an Automated Teller Machine (ATM). A decision table shows combinations of inputs and/or stimuli (causes) with their associated outputs and/or actions (effects). A decision table consists of four quadrants: conditions (inputs), actions (outputs), condition entries (values) and action entries (results). A decision table can be used to test components that have multiple inputs and outputs that depend on logical combinations of conditions. For example, for testing the ATM, we can identify three input conditions: the bank card is valid, the PIN code is correct, and money is available in the user's account. We can also identify four output actions: the card is rejected, the ATM asks for another PIN code, the ATM asks for another amount, and the ATM dispenses the money. A decision table can show all possible combinations of these conditions and actions in a systematic way.
Use case based testing is not a technique that can cover all possible combinations of input conditions for withdrawing money from an ATM. Use case based testing is a technique that verifies that a software product or system meets its specified requirements or user stories by executing realistic scenarios or workflows. Use case based testing can be used to test components that have complex or dynamic interactions with users or other systems. For example, for testing the ATM, we can identify several use cases, such as withdraw money, check balance, transfer money, etc.Each use case can have one or more scenarios that describe the steps and outcomes of the interaction. However, use case based testing may not cover all possible combinations of input conditions, as some scenarios may be omitted or overlooked.
Boundary value analysis is not a technique that can cover all possible combinations of input conditions for withdrawing money from an ATM. Boundary value analysis is a technique that tests boundary values between partitions of equivalent data. Boundary values are values at the edge of an equivalence partition or at the smallest incremental distance on either side of an edge. Boundary value analysis can be used to test components that have input values that can be divided into partitions of equivalent data. For example, for testing the ATM, we can identify boundary values for the input amount, such as the minimum and maximum amount allowed by the system or the user's account. However, boundary value analysis may not cover all possible combinations of input conditions, as some conditions may not have boundary values or may not be related to input values.
Equivalence class partitioning is not a technique that can cover all possible combinations of input conditions for withdrawing money from an ATM. Equivalence class partitioning is a technique that divides the input data and output results of a software component into partitions of equivalent data. Each partition should contain data that is treated in the same way by the component. Equivalence class partitioning can be used to test components that have input values that can be divided into partitions of equivalent data. For example, for testing the ATM, we can identify equivalence partitions for the input amount, such as valid amount (within the range allowed by the system and the user's account) and invalid amount (outside the range allowed by the system or the user's account). However, equivalence class partitioning may not cover all possible combinations of input conditions, as some conditions may not be related to input values or may have more than two partitions. Verified References: [A Study Guide to the ISTQB Foundation Level 2018 Syllabus - Springer], Chapter 4, page 34-46.

NEW QUESTION # 313
Out of the following. what is not needed to specify in defect report?

- A. How to fix the defect
- B. Test environment details
- C. Seventy and priority
- D. How to reproduce the defect

Answer: A

Explanation:
A defect report is a document that records the details of a defect found during testing. A defect report typically contains the following items:
* Identifier: A unique identifier for the defect report
* Summary: A concise summary of the defect
* Description: A detailed description of the defect, including the steps to reproduce it, the expected and actual results, and any relevant screenshots or logs
* Severity: The degree of impact that the defect has on the system
* Priority: The level of urgency for resolving the defect
* Status: The current state of the defect, such as new, open, resolved, closed, etc.
* Resolution: The action taken to resolve the defect, such as fix, workaround, reject, etc. Out of these items, the one that is not needed to specify in a defect report is how to fix the defect. How to fix the defect is a technical solution that is usually determined by the developer who is assigned to resolve the defect. How to fix the defect is not part of the defect report, but rather part of the code change or patch that is delivered to fix the defect. The other items are needed to specify in a defect report, as they provide essential information for identifying, tracking and resolving defects. Verified References: [A Study Guide to the ISTQB Foundation Level 2018 Syllabus - Springer], Chapter 3, page 32-33.

NEW QUESTION # 314

Which ONE of the following is a CORRECT example of the purpose of a test plan?

- A. A test plan is a good document to have for the agile projects because it helps the test manager assign tasks to different people
- B. A test manager decides to create a one page test plan for an agile project for communicating the broad activities and explaining why detailed test cases will not be written as mandated by the test policy.
- C. A test lead decides to write a detailed test plan so that in future, in case of project failure responsibilities could be assigned to the right person
- D. A test manager should always create a very simple test plan because the purpose of test plan is to ensure that there is documentation for the purpose of audits.

Answer: B

NEW QUESTION # 315

For each test case to be executed, the following table specifies its dependencies and the required configuration of the test environment for running such test case:

Assume that C0NF1 is the initial configuration of the test environment Based on this assumption, which of the following is a test execution schedule that is compatible with the specified dependencies and allows minimising the number of switches between the different configurations of the test environment