

## ISTQB-CTFL Dumps

### ISTQB-Foundation Level Exam

<https://www.certleader.com/ISTQB-CTFL-dumps.html>



**NEW QUESTION 1**

Consider the following testing levels:

- 1) Component Testing
- 2) Integration Testing
- 3) System Testing
- 4) Acceptance Testing

Which of the following statements is true?

- A. Integration and system testing are applicable when V-model is used. Component and acceptance testing are applicable when iterative development models are used.
- B. All the testing levels are applicable to V-model for software development.
- C. Only acceptance testing is applicable for iterative models.
- D. Acceptance testing is applicable for all software development model.
- E. Component and system testing are applicable only for the V-model.
- F. All testing levels are applicable, independent of which software development life-cycle process (V-model, iterative, incremental) is used.
- G. iterative, incremental) is used.

**Answer: D**

**Explanation:**

All testing levels are applicable, independent of which software development life-cycle process (V-model, iterative, incremental) is used. Testing levels are defined based on the scope and objectives of testing, not on the software development model. Component testing, integration testing, system testing and acceptance testing are common testing levels that can be applied to any software development model, as long as they are planned and executed properly. The V-model is a software development model that emphasizes the relationship between each development phase and its corresponding testing phase. Iterative and incremental models are software development models that divide the development process into smaller cycles or iterations, where each iteration produces a working version of the software that can be tested and evaluated. Verified References: A Study Guide to the ISTQB® Foundation Level 2018 Syllabus - Springer, page 18.

**NEW QUESTION 2**

Which of the following statements regarding inspection is NOT true?

- A. An inspection may be led by a trained moderator who shall not be the author.
- B. The main purpose of an inspection is to find solutions to the problems.
- C. An inspection can be performed by peers.
- D. An inspection shall follow a formal process based on rules and checklists with entry and exit criteria.

**Answer: B**

**Explanation:**

An inspection is a type of review that follows a defined process with formal entry and exit criteria and roles and responsibilities for participants. An inspection can be performed by peers with different roles, such as moderator, author, reviewer and scribe. The following statement about inspection is not true:

? B) The main purpose of an inspection is to find solutions to the problems. This statement is not true, as the main purpose of an inspection is to find defects or issues in a work product, not to find solutions to the problems. Finding solutions to the problems is a debugging or problem-solving activity that is usually performed by the author or developer after receiving the inspection report. The following statements about inspection are true:

? A) An inspection may be led by a trained moderator who shall not be the author.

This statement is true, as an inspection requires a moderator role who leads the inspection process and ensures that it follows the rules and standards. The moderator should be trained in inspection techniques and should not be the author of the work product under inspection, in order to avoid bias or conflict of interest.

? C) An inspection can be performed by peers. This statement is true, as an inspection involves peer review, which means that the work product under inspection is evaluated by people who have similar roles or expertise as the author, but who are not directly involved in creating or modifying the work product.

? D) An inspection shall follow a formal process based on rules and checklists with entry and exit criteria. This statement is true, as an inspection follows a formal process that consists of six main steps: planning, kick-off meeting, individual preparation, review meeting, rework and follow-up. Each step has defined rules and checklists to guide the participants and ensure consistency and quality. Each step also has entry and exit criteria to ensure that the prerequisites and objectives are met before moving to the next step. Verified References: A Study Guide to the ISTQB® Foundation Level 2018 Syllabus - Springer, Chapter 3, page 28-29.

**NEW QUESTION 3**

Who of the following has the best knowledge to decide what tests in a test project should be automated?

- A. The developer
- B. The customer
- C. The development manager
- D. The test leader

**Answer: D**

**Explanation:**

The test leader is the person who is responsible for planning, monitoring, and controlling the test activities and resources in a test project. The test leader should have the best knowledge of the test objectives, scope, risks, resources, schedule, and quality criteria. The test leader should also be aware of the test automation criteria, such as the execution frequency, the test support, the team education, the roles and responsibilities, and the devs and testers collaboration<sup>1</sup>. Based on these factors, the test leader can decide which tests are suitable for automation and which are not, and prioritize them accordingly. The test leader can also coordinate with the test automation engineers, the developers, and the stakeholders to ensure the alignment of the test automation strategy with the test project goals and expectations. References = ISTQB Certified Tester Foundation Level (CTFL) v4.0 Syllabus, Chapter 2, Section 2.3.1, Page 152; ISTQB Glossary of Testing Terms v4.0, Page 403; ISTQB Certified Tester Foundation Level (CTFL) v4.0 Syllabus, Chapter 6, Section 6.1.1, Page 514; Top 8 Test Automation Criteria You Need To Fulfill - QAMIND<sup>1</sup>

**NEW QUESTION 4**

Which of the following is a correct reason to apply test automation?

- A. When a new test automation tool is launched

- B. When there are a lot of repetitive testing tasks
- C. When it is easy to automate
- D. When it is cheap to buy test automation tools

**Answer: B**

**Explanation:**

A correct reason to apply test automation is when there are a lot of repetitive testing tasks. Test automation is the use of software tools or scripts to perform or support testing activities, such as test case execution, test result comparison, test data generation, etc. Test automation can be beneficial when there are a lot of repetitive testing tasks that need to be performed frequently or consistently, such as regression testing, performance testing, load testing, etc. Test automation can help to save time and effort, increase reliability and accuracy, and improve coverage and efficiency of testing. The other options are not correct reasons to apply test automation. When a new test automation tool is launched is not a reason to apply test automation, but rather a factor for choosing a test automation tool. When it is easy to automate is not a reason to apply test automation, but rather a factor for evaluating the feasibility of test automation. When it is cheap to buy test automation tools is not a reason to apply test automation, but rather a factor for estimating the cost and benefit of test automation. Verified References: A Study Guide to the ISTQB® Foundation Level 2018 Syllabus - Springer, page 10.

**NEW QUESTION 5**

A software system checks age in order to determine which welcome screen to display. Age groups are:

Group I: 0-12

Group II: 13-18 Group III: over 18

Which of the below represent boundary values?

- A. (-1,0,12,13,18,19)
- B. (-1,0,11,12,13,14,18,19)
- C. (0,12,13,18,19)
- D. (4,5,15,20)

**Answer: A**

**Explanation:**

A correct list of boundary values for the age input should include the minimum and maximum values of each age group (0, 12, 13, 18), as well as the values just below and above each boundary (-1, 19). Boundary value analysis is a test design technique that involves testing the values at or near the boundaries of an input domain or output range, as these values are more likely to cause errors than values in the middle. Option A satisfies this condition, as it has all six boundary values (-1, 0, 12, 13, 18, 19). Option B has two values from the same equivalence class (12 and 13), option C has only four boundary values (0, 12, 18, 19), and option D has no boundary values at all. Verified References: A Study Guide to the ISTQB® Foundation Level 2018 Syllabus - Springer, page 34.

**NEW QUESTION 6**

For withdrawing money from 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 input conditions?

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

**Answer: B**

**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 7**

Which of the following would be LEAST appropriate as part of an incident report covering the observation of a failure during testing?

- A. SQL injection into the username entry field allowed a variety of SQL commands to be executed by the application without the appropriate authority.
- B. The user interface was complicated and confusing and I found it quite difficult to follow the test script.
- C. The updates made as part of the add new member' function did not reflect the expected change as the name was written into the address field.
- D. The expected result for the 'list friends' response time was less than 10 seconds, whereas the average response time obtained was 13 seconds.

**Answer: B**

**Explanation:**

An incident report during testing should focus on factual observations of failures or defects in the system, including their impacts and how they deviate from expected results. Options A, C, and D describe specific issues that are directly related to the system's behavior or performance and are suitable for inclusion in an incident report. Option B, which describes the user interface as "complicated and confusing" and relates to the tester's personal difficulty in following the test script, is more subjective and relates to the tester's experience rather than an objective observation of a system failure. Therefore, option B is the least appropriate for an incident report.

**NEW QUESTION 8**

Which of the following is a function of a dynamic analysis tool?

- A. Provide support for traceability of tests, test results and incidents to source documents
- B. Monitor the allocation, use and de-allocation of memory during run-time of a program
- C. Execute programs step-by-step in order to reproduce failures and find corresponding defects
- D. Provide support for release of baselines consisting of configuration items

**Answer: B**

**Explanation:**

A dynamic analysis tool is a tool that performs analysis of a software product based on its behavior during execution. A dynamic analysis tool can monitor various aspects of a program's run-time performance, such as memory usage, CPU load, response time, or resource leaks. A dynamic analysis tool can monitor the allocation, use and de-allocation of memory during run-time of a program, which can help detect defects such as memory leaks, buffer overflows, or memory corruption. A dynamic analysis tool cannot provide support for traceability of tests, test results and incidents to source documents, as this is a function of a test management tool. A dynamic analysis tool cannot execute programs step-by-step in order to reproduce failures and find corresponding defects, as this is a function of a debugging tool. A dynamic analysis tool cannot provide support for release of baselines consisting of configuration items, as this is a function of a configuration management tool. Verified References: [A Study Guide to the ISTQB® Foundation Level 2018 Syllabus - Springer], Chapter 6, page 56-57.

**NEW QUESTION 9**

A test engineer finds a defect while testing. After the developer has fixed the defect, the test engineer decides to re-run a complete section of the tests. Which of the following is correct?

- A. The test engineer should not re-run the tests, as they have already been run, and results recorded.
- B. The test engineer should not re-run the tests, they should be part of the developer tests.
- C. The test engineer should re-run the tests, in order to ensure that new defects have not been introduced by the fix.
- D. The test engineer should re-run the tests, because the defect shows that the test cases need to be updated.

**Answer: C**

**Explanation:**

The test engineer should re-run the tests, in order to ensure that new defects have not been introduced by the fix. This is also known as regression testing, which is a type of testing that verifies that previously tested software still performs correctly after a change. Regression testing helps to detect any side effects or unintended consequences of a fix or a modification. The other options are incorrect reasons for re-running the tests. The test engineer should not re-run the tests, as they have already been run, and results recorded, because this ignores the possibility of new defects caused by the fix. The test engineer should not re-run the tests, they should be part of the developer tests, because this assumes that developer tests are sufficient and reliable, which may not be true. The test engineer should not re-run the tests, because the defect shows that the test cases need to be updated, because this does not address the impact of the fix on other test cases or functionalities. Verified References: A Study Guide to the ISTQB® Foundation Level 2018 Syllabus - Springer, page 41.

**NEW QUESTION 10**

Given the following review process main activities and specific review activities:

- \* a. Planning
- \* b. Initiate review
- \* c. Issue communication and analysis d. Fixing and reporting
- \* 1. Creating defect reports
- \* 2. Estimating effort and timeframe
- \* 3. Recording updated status of defects
- \* 4. Selecting the people to participate
- \* 5. Distributing the work product and other material
- \* 6. Evaluating the review findings

Which of the following BEST matches the review process main activities with the appropriate specific review activities?

- A. 2-a, 4-a, 5-b, 6-c, 1-d, 3-d
- B. 2-a, 5-a, 1-b, 4-b, 3-c, 6-d
- C. 1-a, 4-b, 5-b, 6-c, 2-d, 3-d
- D. 2-a, 4-b, 5-c, 1-
- E. 3-d, 6-d

**Answer: A**



**Explanation:**

Matching the main review process activities with the specific review activities, we see that planning includes estimating effort and timeframe (2) and selecting people to participate (4). Initiating a review involves distributing work products and other material (5). Issue communication and analysis includes evaluating the review findings (6). Fixing and reporting would entail creating defect reports (1) and recording the updated status of defects (3). References: ISTQB Certified Tester Foundation Level Syllabus v4.0, Section 3.2 "Review Process".

**NEW QUESTION 10**

Which of the following software development models BEST exemplifies a model that does NOT support the principle of early testing?

- A. The iterative development model
- B. The V-model
- C. The Waterfall model
- D. The incremental development model

**Answer: C**

**Explanation:**

The Waterfall model exemplifies a software development model that does not support the principle of early testing. In the Waterfall model, each phase must be completed before the next begins, which delays testing until after the completion of the earlier phases like requirements gathering and design. This can often result in finding defects later in the development cycle, making them more expensive and time-consuming to fix (ISTQB not-for-profit association) (ISTQB not-for-profit association). References:

? ISTQB® Certified Tester Foundation Level Syllabus v4.0: [https://istqb-main-web-prod.s3.amazonaws.com/media/documents/ISTQB\\_CTFL\\_Syllabus-v4.0.pdf](https://istqb-main-web-prod.s3.amazonaws.com/media/documents/ISTQB_CTFL_Syllabus-v4.0.pdf)

? ISTQB News Release on CTFL v4.0: <https://www.istqb.org/news/posts/istqb-releases-certified-tester-foundation-level-v40-ctfl/>

**NEW QUESTION 14**

Which of the following definitions is NOT true?

- A. Test data preparation tools fill databases, create files or data transmissions to set up test data to be used during the execution of tests.
- B. Test execution tools execute test objects using automated test scripts.
- C. Test Management tools monitor and report on how a system behaves during the testing activities.
- D. Test comparators determine differences between files, databases or test results.

**Answer: C**

**Explanation:**

Test Management tools are designed to support the planning, execution, and monitoring of the testing process. They provide features for managing test cases, test runs, tracking defects, and reporting on testing activities. However, the statement in option C describes Test Management tools as monitoring and reporting on the system's behavior during testing activities, which is not accurate. Test Management tools focus on the testing process itself rather than on the behavior of the system under test.

? Test data preparation tools (A) indeed create and manage test data for use during test execution.

? Test execution tools (B) automate the execution of test cases and the comparison of actual outcomes against expected results.

? Test comparators (D) are tools that compare actual outcomes with expected outcomes, highlighting discrepancies.

Therefore, option C is the correct answer as it inaccurately describes the function of Test Management tools.

**NEW QUESTION 17**

Out of the following, what is not needed to specify in defect report?

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

**Answer: C**

**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 22**

Which of the following is a typical product risk?

- A. Poor usability of the software
- B. A problem in the code developed by a 3rd party
- C. Low quality of the configuration data, test data and tests
- D. Problem in defining the right requirements

**Answer: A**

**Explanation:**

A typical product risk involves issues directly related to the software product's functionality, performance, usability, reliability, etc. Option A, "Poor usability of the software," directly impacts the end-user's interaction with the software and is a quality attribute of the product itself, making it a product risk. Options B, "A problem in the code developed by a 3rd party," C, "Low quality of the configuration data, test data and tests," and D, "Problem in defining the right requirements," can be considered either product or project risks depending on the context, but option A is the most directly associated with a typical product risk concerning the quality and usability of the software.

**NEW QUESTION 25**

In the newest version of payroll system number of changes were made. As a tester you got a task to perform regression and confirmation tests. Which of the following project activities are related to confirmation testing?

- A. Testing due to the application of a new version of the interface
- B. Testing that fixes resolved the defects in the search function
- C. Testing if a system still works after update of an operating system
- D. Testing to ensure the adding of a new functionalities haven't broken existing functions

**Answer:** B

**Explanation:**

Confirmation testing, also known as re-testing, is performed to verify that specific defects have been successfully fixed.

Option A: "Testing due to the application of a new version of the interface" would typically involve regression testing, not confirmation testing.

Option B: "Testing that fixes resolved the defects in the search function" fits the description of confirmation testing as it focuses on ensuring that specific issues have been addressed. Option C: "Testing if a system still works after update of an operating system" is an example of regression testing, as it checks the overall system behavior after an update. Option D: "Testing to ensure the adding of new functionalities haven't broken existing functions" is another example of re (ISTQB not-for-profit association) (Udemy) it checks for unintended consequences of new changes.

Therefore, the correct answer is B†source9†source. References:

? Certified Tester Foundation Level v4.0

? ISTQB Foundation Level Syllabus 4.0 (2023)

**NEW QUESTION 28**

ST is a Software Testing organization which utilizes a testing knowledge base. Access to ST knowledge base can be either full or limited. Access level is determined based on ST certification and testing experience as follows:

- \* 1. If ST certified, with less than 5 years testing experience - allow limited access
- \* 2. If ST certified, 5-10 years of testing experience - allow full access
- \* 3. If not ST certified with 5-10 years of testing experience - allow limited access.

What would be the results for:

- A - ST certified. 12 years of testing experience
- B - Not ST certified. 7 years of testing experience
- C - Not ST certified. 3 years of testing experience

- A. A - unknown B - limned access C- unknown
- B. A - full access B - limited access C - unknown
- C. A - full access B - limited access C - limited access
- D. A - unknown B - full access C - unknown

**Answer:** B

**Explanation:**

The correct answer can be derived by applying the given rules to each case:

? A is ST certified and has 12 years of testing experience, which is more than 10 years. Therefore, A does not match any of the rules and the result is unknown.

? B is not ST certified and has 7 years of testing experience, which is between 5 and 10 years. Therefore, B matches rule 3 and the result is limited access.

? C is not ST certified and has 3 years of testing experience, which is less than 5 years. Therefore, C does not match any of the rules and the result is unknown.

Verified References: This question does not require any external references, as it is based on logical reasoning.

**NEW QUESTION 33**

A test manager defined the following test levels in her test plan; Component, System and Acceptance.

Which Software Development lifecycle is the Test Manager most likely following?

- A. V-Model
- B. Agile
- C. Waterfall
- D. Prototyping

**Answer:** A

**Explanation:**

The test manager is most likely following the V-model for software development. The V-model is a software development model that defines four testing levels that correspond to four development phases: component testing corresponds to component design, integration testing corresponds to architectural design, system testing corresponds to system requirements specification, and acceptance testing corresponds to user requirements specification. The V-model also defines the test planning and test execution activities for each testing level. Agile is a software development model that follows an iterative and incremental approach, where testing is integrated into each iteration and adapts to changing requirements and feedback. Waterfall is a software development model that follows a sequential and linear approach, where testing is performed after the development phase is completed. Prototyping is a software development model that involves creating a simplified version of the software to elicit user feedback and validate requirements before developing the final product. Verified References: A Study Guide to the ISTQB® Foundation Level 2018 Syllabus - Springer, page 18.

**NEW QUESTION 36**

Why is it important to select a test technique?

- A. There are usually loo many test cases that may be run on a syste
- B. Test techniques help reduce the number of tests.

- C. The only way to test a software application is by using well proven test techniques.  
D. Selecting the right test technique in a given situation Increases the effectiveness of the test process Oy creating tests with higher chance of finding bugs.  
E. Test techniques define the number of regression cycles, which in turn impact the project schedule.

**Answer: C**

**Explanation:**

Selecting the right test technique is crucial because different techniques are suited to different types of testing and can significantly increase the effectiveness of the testing process by creating tests that are more likely to find defects. While reducing the number of tests (A) and defining the number of regression cycles (D) are considerations in the testing process, they are not the primary reasons for selecting a test technique. The assertion that the only way to test a software application is by using well-proven test techniques (B) is too restrictive and does not acknowledge the adaptability required in testing to suit different contexts and objectives. Therefore, option C is the most comprehensive reason, as it focuses on the effectiveness and efficiency of testing, leading to the creation of high-quality tests that have a higher chance of finding bugs.

**NEW QUESTION 41**

Which are the MAIN goals of risk management in a software project?

- A. To increase the success probability for the project regardless of costs.  
B. To increase focus on preventative processes and to increase satisfaction for the testers.  
C. To control contractual problems and minimize the impacts of company policies.  
D. To reduce the probability of undesired situations and to reduce the effect of potential impact.

**Answer: D**

**Explanation:**

Risk management is a process that identifies, analyzes, evaluates and mitigates risks in a software project. Risks are factors that may negatively affect the quality, schedule, budget or scope of a software project. The main goals of risk management in a software project are to reduce the probability of undesired situations and to reduce the effect of potential impact. This can be achieved by applying various strategies, such as avoidance, transfer, reduction or acceptance. Risk management does not aim to increase the success probability for the project regardless of costs, as this may not be feasible or realistic. Risk management does not aim to increase focus on preventative processes and to increase satisfaction for the testers, as these are secondary or indirect outcomes. Risk management does not aim to control contractual problems and minimize the impacts of company policies, as these are specific types of risks that may not apply to all projects. Verified References: A Study Guide to the ISTQB® Foundation Level 2018 Syllabus - Springer, Chapter 2, page 14-15.

**NEW QUESTION 42**

Why should you choose a test technique?

- A. Because you need to match the way you test to the content of the product under test  
B. Because of the time constraints that usually accompany a test project  
C. Because this way you cover the full scope of the product's functionality  
D. Because choosing a test technique is a common practice in software testing

**Answer: A**

**Explanation:**

You should choose a test technique because you need to match the way you test to the content of the product under test. A test technique is a method or process for deriving and selecting test cases based on some criteria or rules. Different test techniques are suitable for different types of software products, depending on their characteristics, functionalities, requirements, specifications, risks, etc. Choosing a test technique helps to ensure that the test cases are relevant, effective, and efficient for the product under test. The other options are not correct reasons to choose a test technique. Time constraints are not a factor for choosing a test technique, but rather for prioritizing or optimizing testing activities. Covering the full scope of the product's functionality is not a guarantee of choosing a test technique, but rather a goal of testing. Choosing a test technique is not a common practice in software testing, but rather a professional skill and responsibility. Verified References: A Study Guide to the ISTQB® Foundation Level 2018 Syllabus - Springer, page 31.

**NEW QUESTION 44**

Which of the following options cover the test types performed during typical system testing phase:

- A. UsabilityII Requirements based scenariosIII Testing parts of the code in isolationIV Correct order of parameters in API calls  
B. I, III  
C. II  
D. I  
E. IV  
F. II  
G. IV

**Answer: B**

**Explanation:**

System testing is a level of testing performed to evaluate the behavior and quality of a whole software product or system. System testing can include various types of testing, such as:

- ? I) Usability testing: A type of testing that evaluates how easy, efficient and satisfying it is to use the software product or system from the user's perspective.  
? II) Requirements based scenarios testing: A type of testing that verifies that the software product or system meets its specified requirements or user stories by executing realistic scenarios or workflows. System testing does not include the following types of testing, as they are more suitable for lower levels of testing, such as unit testing or integration testing:  
? III) Testing parts of the code in isolation: A type of testing that verifies the functionality and quality of individual software components or units by isolating them from other components or units.  
? IV) Correct order of parameters in API calls: A type of testing that verifies the functionality and quality of software components or units that communicate with each other through application programming interfaces (APIs) by checking the correct order and format of parameters in API calls. Verified References: [A Study Guide to the ISTQB® Foundation Level 2018 Syllabus - Springer], Chapter 2, page 20-21; Chapter 4, page 34-35.

**NEW QUESTION 46**

While repotting a defect, which attribute indicates the degree of impact that the defect has on the system?

- A. Priority
- B. Severity
- C. Status
- D. Description

**Answer: B**

**Explanation:**

In defect reporting, the attribute that indicates the degree of impact that the defect has on the system is the severity. Severity reflects the seriousness of the defect in terms of its impact on the operation of the system, ranging from minor issues that do not significantly affect the system's functionality to critical defects that can cause system failure. Therefore, option B is the correct answer.

**NEW QUESTION 49**

Which of the following can be considered a VALID exit criterion? I Estimates of defect density or reliability measures.

II. The completion and publication of an exhaustive Test Report.

III. Accuracy measures, such as code, functionality or risk coverage. IV Residual risks such as lack of code coverage in certain areas.

- A. I, III, IV
- B. I, II, III
- C. III, IV
- D. II, III, IV

**Answer: A**

**Explanation:**

An exit criterion is a condition that defines when a test activity has been completed or when a test phase can be concluded. An exit criterion can be based on various factors, such as:

? I) Estimates of defect density or reliability measures. These are quantitative measures that indicate how many defects are present in the software product or how likely it is to fail under certain conditions. These can be used as exit criteria to ensure that the software product meets a certain level of quality or performance before moving to the next phase or releasing it to the customer.

? III) Accuracy measures, such as code coverage, functionality coverage or risk coverage. These are quantitative measures that indicate how much of the software product has been tested in terms of its code, functionality or risk. These can be used as exit criteria to ensure that the test suite is adequate or complete before moving to the next phase or releasing it to the customer.

? IV) Residual risks, such as lack of code coverage in certain areas, unresolved defects or unknown factors. These are qualitative measures that indicate the remaining risks or uncertainties associated with the software product after testing. These can be used as exit criteria to ensure that the residual risks are acceptable or manageable before moving to the next phase or releasing it to the customer. The following factor is not a valid exit criterion:

? II) The completion and publication of an exhaustive Test Report. This is not a valid exit criterion, as it does not reflect the quality or completeness of the testing process or product. A test report is a document that summarizes the results and outcomes of a test activity or phase. A test report can be used as an input for deciding whether to exit a test activity or phase, but it is not a condition that defines when to exit. Verified References: A Study Guide to the ISTQB® Foundation Level 2018 Syllabus - Springer, Chapter 2, page 13; Chapter 6, page 58-59.

**NEW QUESTION 52**

Which test approach will best fit a new project, with little documentation and high probability for bugs?

- A. Exploratory testing
- B. Requirements based testing
- C. Metric based approach
- D. Regression testing

**Answer: A**

**Explanation:**

Exploratory testing is an approach to testing that emphasizes learning, test design and test execution at the same time. Exploratory testing relies on the tester's skills, creativity and intuition to explore the software under test and discover defects. Exploratory testing is suitable for a new project with little documentation and high probability for bugs, as it can help uncover unknown requirements, assumptions and risks. Exploratory testing is not requirements based testing, which is an approach to testing that derives test cases from documented requirements or specifications. Requirements based testing is not feasible for a new project with little documentation, as it requires clear and complete requirements to be available. Exploratory testing is not metric based approach, which is an approach to testing that uses quantitative measures to monitor and control the testing process and evaluate the quality of the software product. Metric based approach is not effective for a new project with high probability for bugs, as it may not capture all aspects of quality and may lead to false confidence or unrealistic expectations. Exploratory testing is not regression testing, which is an approach to testing that verifies that previously tested software still performs correctly after changes. Regression testing is not relevant for a new project with no previous versions or baselines. Verified References: [A Study Guide to the ISTQB® Foundation Level 2018 Syllabus - Springer], Chapter 5, page 47-48.

**NEW QUESTION 56**

Given the following state model of sales order software: SEE ATTACHMENT

Which of the following sequences of transitions provides the highest level of transition coverage for the model (assuming you can start in any state)?

- A. IN PRODUCTION -> CANCELLED -> PLACED -> IN PRODUCTION -> CANCELLED -> PLACED
- B. IN PRODUCTION -> SHIPPED -> INVOICED -> CANCELLED -> PLACED -> IN PRODUCTION
- C. PLACED -> IN PRODUCTION -> SHIPPED -> CANCELLED -> PLACED
- D. PLACED -> CANCELLED -> PLACED -> CANCELLED -> PLACED -> IN PRODUCTION -> CANCELLED

**Answer: B**

**Explanation:**

State transition testing is a black-box testing technique where test cases are designed to cover states and transitions of a state machine. Given the state model with the following transitions:



? PLACED -> IN PRODUCTION  
? IN PRODUCTION -> CANCELLED  
? IN PRODUCTION -> SHIPPED  
? SHIPPED -> INVOICED  
? INVOICED -> CANCELLED  
? CANCELLED -> PLACED

To cover all transitions at least once, we need to create a sequence that covers all six transitions.

Option A: IN PRODUCTION -> CANCELLED -> PLACED -> IN PRODUCTION -> CANCELLED -> PLACED- Misses SHIPPED -> INVOICED and INVOICED -> CANCELLED transitions.

Option B: IN PRODUCTION -> SHIPPED -> INVOICED -> CANCELLED -> PLACED -> IN PRODUCTION- Covers all transitions.

Option C: PLACED -> IN PRODUCTION -> SHIPPED -> CANCELLED (ISTQB not-for-profit association) (Udemy)sses INVOICED -> CANCELLED transition.

Option D: PLACED -> CANCELLED -> PLACED -> CANCELLED -> PLACED -> IN PRODUCTION -> CANCELLED- Misses SHIPPED -> INVOICED and INVOICED -> CANCELLED transitions.

Given these, Option B covers all the transitions6†source9†source. References:

? Certified Tester Foundation Level v4.0

? 10 Sample Exams ISTQB Foundation Level (CTFL) v4.0

### NEW QUESTION 57

In maintenance testing, what is the relationship between impact analysis and regression testing?

- A. Impact analysis requires a regression testing for only the tests that have detected faults in previous SW release
- B. There is no relationship between impact analysis and regression testing.
- C. Impact analysis requires a regression testing for all program elements which were newly integrated (new functionalities).
- D. The impact analysis is used to evaluate the amount of regression testing to be performed.

**Answer: D**

#### Explanation:

In maintenance testing, the relationship between impact analysis and regression testing is that the impact analysis is used to evaluate the amount of regression testing to be performed. Maintenance testing is a type of testing that is performed on an existing software product after it has been delivered or deployed, in order to ensure that it still meets its requirements and functions correctly after a change or a modification. Maintenance testing can be triggered by various reasons, such as corrective maintenance (fixing defects), adaptive maintenance (adapting to new environments), perfective maintenance (improving performance), preventive maintenance (avoiding future problems), etc. Impact analysis is a technique that is used to assess the extent and nature of changes introduced by maintenance activities on the software product or project. Impact analysis helps to identify which parts of the software product are affected by the changes, which parts need to be modified or updated accordingly, which parts need to be retested or verified for correctness or compatibility, etc. Regression testing is a type of testing that verifies that previously tested software still performs correctly after a change or a modification. Regression testing helps to detect any side effects or unintended consequences of maintenance activities on the software product's functionality or quality. Regression testing can be performed at various levels and scopes depending on the impact analysis results. Therefore, in maintenance testing, impact analysis is used to evaluate the amount of regression testing to be performed. Verified References: A Study Guide to the ISTQB® Foundation Level 2018 Syllabus - Springer, page 20.

### NEW QUESTION 58

As the last stage of a test cycle of an embedded device, you are performing exploratory testing. You observed that some character. (A, X and Z) sent via a serial port to the device do not get registered on the device whereas they should be. You suspect that this could be due to a wrong configuration of the "bit parity" parameter.

Which of the following items of an incident report would you be UNABLE to write down based on this information?

- A. Expected result
- B. Test case identifier
- C. Test setup details
- D. Actual result

**Answer: B**

#### Explanation:

An incident report is a document that records the details of an incident. An incident report typically contains the following items:

? Identifier: A unique identifier for the incident report

? Summary: A concise summary of the incident

? Description: A detailed description of the incident, including the steps to reproduce it, the expected and actual results, and any relevant screenshots or logs

? Severity: The degree of impact that the incident has on the system

? Priority: The level of urgency for resolving the incident

? Status: The current state of the incident, such as new, open, resolved, closed, etc.

? Resolution: The action taken to resolve the incident, such as fix, workaround, reject, etc. Based on the information given in the question, the tester would be able to write down all of these items except for the test case identifier. A test case identifier is a unique identifier for a test case that is used to link it to other test artifacts, such as test plans, test scripts, test results or incident reports. However, since the tester is performing exploratory testing, there is no predefined test case that can be associated with the incident. Exploratory testing is an approach to testing that emphasizes learning, test design and test execution at the same time. Exploratory testing relies on the tester's skills, creativity and intuition to explore the software under test and discover defects. Exploratory testing does not use formal test cases or scripts, but rather uses test charters or missions that guide the tester's actions and objectives. Verified References: A Study Guide to the ISTQB® Foundation Level 2018 Syllabus - Springer, Chapter 3, page 32-33; Chapter 5, page 47-48.

### NEW QUESTION 60

Which of the following statements about test reports are TRUE?

- II. Test reports shall give stakeholders information as basis for decisions.
- III Test reports shall summarize what happened through a period of testing.
- IV. Test reports shall be approved by the development team, the test team and the customer

- A. Test reports shall include information about remaining risks.
- B. II, III, V

- C. I, II, IV
- D. I, III, v
- E. II, III, IV

**Answer:** A

**Explanation:**

Statements II, III and V are true about test reports. Test reports are documents that provide information on the results and status of testing activities for a given period or phase. Test reports should give stakeholders information as basis for decisions, such as whether to release the software product, whether to continue testing, whether to change the scope or priorities of testing, etc. Test reports should summarize what happened through a period of testing, such as what test cases were executed, what defects were found, what risks were identified, what issues were encountered, what achievements were made, etc. Test reports should include information about remaining risks, such as what defects are still open, what test cases are still pending, what functionalities are still untested, what uncertainties are still unresolved, etc. Statements I and IV are not true about test reports. Test reports do not need to be approved by the test team, the development team, or the customer, unless it is specified by the test policy or the test plan. Test reports only need to be reviewed and verified by the test leader or the test manager before being distributed to the intended recipients. Verified References: A Study Guide to the ISTQB® Foundation Level 2018 Syllabus - Springer, page 141.

**NEW QUESTION 63**

Which of the following BEST distinguishes the terms "validation" and "verification"?

- A. Verification is confirmation through the provision of objective evidence that the specified requirements have been met while validation is confirmation through the provision of objective evidence that the requirements for a specific intended use have been met
- B. Verification is confirmation through the provision of subjective evidence that the specified requirements have been met while validation is confirmation through the provision of subjective evidence that the designs for a specific intended use have been met
- C. Validation is confirmation through the provision of objective evidence that the specified requirements have been met while verification is confirmation through the provision of objective evidence that the requirements for a specific intended use have been met
- D. Validation is confirmation through the provision of subjective evidence that the specified requirements have been met while verification is confirmation through the provision of subjective evidence that the designs for a specific intended use have been met

**Answer:** A

**Explanation:**

In the context of software testing, the ISTQB Certified Tester Foundation Level (CTFL) v4.0 differentiates between "validation" and "verification" based on their respective focuses in the software development lifecycle. Verification is the process of evaluating a system or component to determine whether the products of a given development phase satisfy the conditions imposed at the start of that phase. In simpler terms, verification is about checking the product against the specified requirements to ensure it was built correctly. Validation, on the other hand, involves evaluating a system or component during or at the end of the development process to determine whether it meets specified requirements for its intended use. This means validation is about ensuring the product fulfills its intended use and meets the needs of the user.

References:

? ISTQB CTFL Syllabus v4.0: ISTQB Official Website

? ISTQB Foundation Level Resources v4.0: ASTQB Resources

**NEW QUESTION 67**

"Statement Testing" is part of;

- A. Specification Based testing
- B. Decision Testing
- C. Experience based testing
- D. Structured based testing

**Answer:** D

**Explanation:**

Statement Testing is a type of white-box testing technique where the test cases are designed based on the implementation of the software, specifically aiming to execute every statement in the code at least once. This falls under the category of structure-based testing (also known as white-box testing), where the internal structure of the system is used to design test cases. Therefore, option D is correct.

**NEW QUESTION 72**

Which of the following is NOT an objective of testing?

- A. Finding defects
- B. Providing information for decision-making
- C. Gaining confidence about the level of quality of the software
- D. Analyzing and removing the cause of failures

**Answer:** D

**Explanation:**

Analyzing and removing the cause of failures is not an objective of testing, but rather a task of development or maintenance. A failure is an event or behavior that deviates from the expected or specified result of a system under test. A failure is caused by an error (also known as a mistake or a fault) in the software code, design, or specification. Analyzing and removing the cause of failures is a process of locating and fixing errors in the software code, design, or specification, which is also known as debugging or defect resolution. Analyzing and removing the cause of failures does not aim to find or report defects, but rather to correct or prevent them. The other options are objectives of testing. Finding defects is one of the main objectives of testing, as it helps to improve the quality and reliability of the software product. Providing information for decision-making is another objective of testing, as it helps to support decision making and risk management. Gaining confidence about the level of quality of the software is another objective of testing, as it helps to assure that the software product meets its requirements and customer or user needs and expectations. Verified References: A Study Guide to the ISTQB® Foundation Level 2018 Syllabus - Springer, page 3.

**NEW QUESTION 74**

- I When closing the test activities, all the testware resources can be uninstalled and released
- II All the testware should be subject to Configuration Management
- III. The testware. at the end of the project, should be transferred to the organization responsible for maintenance
- IV The developers are responsible for the correct installation of the testware

- A. II, III
- B. I, III
- C. I, IV
- D. II, IV

**Answer: A**

**Explanation:**

Testware is a term that refers to all artifacts produced during the testing process, such as test plans, test cases, test scripts, test data, test results, defect reports, etc. The following statements about testware are correct:

? II) All the testware should be subject to Configuration Management. Configuration management is a process that establishes and maintains consistency among work products throughout their life cycle. Configuration management applies to all testware, as it helps ensure their quality and consistency, track their changes and defects, control their versions and access rights, and link them to other artifacts.

? III) The testware at the end of the project should be transferred to the organization responsible for maintenance. Maintenance testing is testing performed on a software product after delivery to correct defects or improve performance or other attributes. Maintenance testing requires testware from previous testing activities or phases, such as test cases, test data, test results, etc. Therefore, the testware at the end of the project should be transferred to the organization responsible for maintenance testing, such as support team or maintenance team. The following statements about testware are incorrect:

? I) When closing the test activities, all the testware resources can be uninstalled and released. This statement is incorrect, as some testware resources may still be needed for future testing activities or phases, such as maintenance testing or regression testing. Therefore, when closing the test activities, some testware resources should be archived and stored for future use, while others can be uninstalled and released.

? IV) The developers are responsible for the correct installation of the testware. This statement is incorrect, as the testers are responsible for the correct installation of the testware. The testers should ensure that they have access to all necessary testware resources and that they are installed and configured properly before starting the test execution. Verified References: A Study Guide to the ISTQB® Foundation Level 2018 Syllabus - Springer, Chapter 6, page 58-61.

**NEW QUESTION 77**

Which of the following is the main benefit of a configuration management of testware?

- A. All testware is backed up with restore option, including incident reports and change request
- B. The testware can be traced to information in requirements tools and to the bug tracking system.
- C. All testware items are identified, version controlled, tracked for changes with relation to each other
- D. There is an easy way to assess the level to test coverage provided by the existing tests

**Answer: C**

**Explanation:**

Configuration management of testware is a critical aspect of maintaining the integrity and traceability of test assets throughout the testing lifecycle. The main benefit of configuration management is to ensure that all testware items, such as test cases, test scripts, test data, and test results, are systematically identified, version controlled, and tracked for changes in relation to each other.

Option C accurately describes this benefit. By applying configuration management principles to testware, teams can manage changes to test assets efficiently, ensuring that the testware remains consistent, up-to-date, and aligned with the version of the software under test. This control mechanism facilitates the reproducibility of tests, enhances the reliability of testing activities, and supports traceability from requirements through to defects.

Options A, B, and D describe other aspects of test management and testing processes but do not capture the core benefit of configuration management of testware, which is centered on the systematic control and tracking of testware items.

**NEW QUESTION 80**

A Test Manager conducts risk assessment for a project. One of the identified risks is: "The sub-contractor may fail to meet his commitment". If this risk materializes, it will lead to delay in completion of testing required for the current cycle.

Which of the following sentences correctly describes the risk?

- A. It is a product risk since any risk associated with development timeline is a product risk.
- B. It is no longer a risk for the Test Manager since an independent party (the sub- contractor) is now managing it
- C. It is a object risk since successful completion of the object depends on successful and timely completion of the tests
- D. It is a product risk since default on part of the sub-contractor may lead to delay in release of the product

**Answer: D**

**Explanation:**

? A product risk is a risk that affects the quality or timeliness of the software product being developed or tested<sup>1</sup>. Product risks are related to the requirements, design, implementation, verification, and maintenance of the software product<sup>2</sup>.

? The risk of the sub-contractor failing to meet his commitment is a product risk, as it could cause a delay in the completion of the testing required for the current cycle, which in turn could affect the release date of the product. The release date is an important aspect of the product quality, as it reflects the customer satisfaction and the market competitiveness of the product<sup>3</sup>.

? The other options are not correct because: References =

? 1 ISTQB® Certified Tester Foundation Level Syllabus v4.0, 2023, p. 97

? 2 ISTQB® Certified Tester Foundation Level Syllabus v4.0, 2023, p. 98

? 3 ISTQB® Certified Tester Foundation Level Syllabus v4.0, 2023, p. 99

? 4 ISTQB® Certified Tester Foundation Level Syllabus v4.0, 2023, p. 100

? 5 ISTQB® Certified Tester Foundation Level Syllabus v4.0, 2023, p. 101

? 6 ISTQB® Certified Tester Foundation Level Syllabus v4.0, 2023, p. 102

**NEW QUESTION 83**

Which of the following tasks is MOST LIKELY to be performed by the tester?

- A. Develop a test strategy and test policy for the organization
- B. Promote and advocate the test team within the organization

- C. Create the detailed test execution schedule
- D. Introduce suitable metrics for measuring test progress

**Answer:** C

**Explanation:**

Testers are typically involved in creating detailed test execution schedules, among other tasks such as designing tests, executing tests, and logging defects. Creating a test strategy and test policy, promoting and advocating the test team, and introducing metrics are typically responsibilities of test managers or senior roles.

In the ISTQB Certified Tester Foundation Level (CTFL) v4.0 syllabus, the responsibilities of testers include creating test cases, setting up test (ISTQB not-for-profit association)nts, executing tests, and reporting defects, which align with creating detailed test execution schedules6†source.

References:

? Certified Tester Foundation Level v4.0

? ISTQB Foundation Level Syllabus 4.0 (2023)

**NEW QUESTION 87**

Which of the following BEST describes checklist-based testing?

- A. Checklist-based testing includes formal tests from detailed lists of test conditions, allowing much repeatability
- B. Checklist-based testing may involve a list of tests based on what is important to the user as well as an understanding of why and how software fails
- C. Checklist-based testing, while popular, provides little consistency and few guidelines concerning test case development
- D. Checklist-based testing is restricted to non-functional testing, including usability, performance, and security test

**Answer:** B

**Explanation:**

Checklist-based testing involves using checklists that contain items, such as potential test conditions, that should be tested. These checklists are often based on insights into what is important to the user, potential areas where software might fail, and specific aspects that need to be tested. It provides a structured yet flexible approach to testing, ensuring key areas are covered while allowing testers to use their experience and understanding of the system. Checklist-based testing is not limited to non-functional testing but can be applied to various types of testing, including functional testing.References:

? ISTQB Certified Tester Foundation Level Syllabus v4.0, Section 4.4.5.

**NEW QUESTION 90**

Which of the following statements BEST describes how test cases are derived from a use case?

- A. Test cases are derived based on non-functional requirements such as usability
- B. Test cases are created using white-box test techniques to execute scenarios of use cases
- C. Test cases are derived based on pair testing between a user and a tester to find defects
- D. Test cases are designed to cover various user behaviors, including basic, exceptional oralternative and error behaviors associated with human users or systems

**Answer:** D

**Explanation:**

Use cases describe a system's behavior as it responds to a request from a user. They typically consist of various scenarios, such as basic flow, alternative flow, and exceptional flow, which represent possible behaviors when a user interacts with the system. When deriving test cases from use cases, it is important to cover these different types of user behaviors.Test cases should be designed to verify how the system behaves during each of these scenarios. This ensures that the system operates correctly for normal and error conditions encountered by human users or systems interacting with the application. Thus, test cases derived from use cases aim to cover basic, exceptional, and alternative flows, ensuring comprehensive coverage.References:

? ISTQB Certified Tester Foundation Level Syllabus v4.0, Section 4.2.4.

**NEW QUESTION 93**

Mark the correct sentences:

- \* Defects are a result of environmental conditions and are also referred to as "Failures"
- \* A human mistake may produce a defect
- \* A system mil totally fail to operate correctly when a failure exists in it
- \* When a defect exists in a system it may result in a failure
- \* Defects occur only as a result of technology changes

- A. II, IV
- B. I, II
- C. IV, V
- D. II, III, IV

**Answer:** A

**Explanation:**

? The question is about marking the correct sentences among the given statements related to defects, failures, and mistakes. According to the ISTQB glossary, the definitions of these terms are1:

? Therefore, out of the five given statements, only two are correct, namely:

? The other three statements are incorrect, namely: References:

? 1: ISTQB Glossary of Testing Terms 4.0, 2023, available at ISTQB) and ASTQB).

**NEW QUESTION 97**

4 equivalence classes are given for integer values:

0 < x <100

100<= x <= 200

200 < x < 500

x >= 500



Which of the following options represent correct set of data for valid equivalence class partitions?

- A. 50; 100; 200. 1000
- B. 0. 1.99, 100.200,201.499, 500;
- C. 0.50; 100; 150.200.350.500;
- D. 50; 100; 250; 1000

**Answer: C**

**Explanation:**

The correct set of data for valid equivalence class partitions should include one value from each equivalence class, and no value from outside the range. Option C satisfies this condition, as it has one value from each of the four equivalence classes (50, 100, 250, 500). Option A has two values from the same equivalence class (100 and 200), option B has values outside the range (0 and 0.99), and option D has two values from the same equivalence class (1000 and 500). Verified References: A Study Guide to the ISTQB® Foundation Level 2018 Syllabus - Springer, page 35.

**NEW QUESTION 101**

A bank software has a feature of locking down accounts that have an overdraft higher than \$10,000.

Due to a bug in the code, the system locks down accounts once they reached \$1000 overdraft.

Based on this context, which of the following is correct statement

- A. It is an example of a defect that it goes undetected, will cause harm to the end customer and to the company (the bank)
- B. It is an example of a defect, that if goes undetected, will cause harm to the end customer but not to the company (the bank)
- C. It is an example of a defect that if goes undetected, will not cause harm to the end customer, but will cause harm to the company (the bank)
- D. It is an example of a defect that if goes undetected, will not cause any real harm to either the end customer or to the company (the bank)

**Answer: A**

**Explanation:**

A defect is a flaw in a component or system that can cause the component or system to fail to perform its required function. A defect can cause harm to the end customer, the company, or both, depending on the impact and severity of the failure. In this case, the defect in the code causes the system to lock down accounts that have an overdraft higher than \$1000, instead of \$10,000. This can cause harm to the end customer, as they may lose access to their funds and face inconvenience or financial difficulties. This can also cause harm to the company, as they may lose customer trust and satisfaction, face legal issues or complaints, or incur additional costs for fixing the defect and restoring the accounts. Verified References: [A Study Guide to the ISTQB® Foundation Level 2018 Syllabus - Springer], Chapter 1, page 3-4.

**NEW QUESTION 105**

Which of the following is an example of black-box dynamic testing?

- A. Functional Testing
- B. Code inspection
- C. Checking memory leaks for a program by executing it
- D. Coverage analysis

**Answer: A**

**Explanation:**

Functional testing is an example of black-box dynamic testing. Black-box testing (also known as specification-based testing) is a type of testing that does not consider the internal structure or implementation of the system under test, but rather its external behavior or functionality. Dynamic testing is a type of testing that involves executing the system under test with various inputs and observing its outputs. Functional testing is a type of black-box dynamic testing that verifies that the system under test performs its intended functions according to its requirements or specifications. Functional testing can be performed at various levels and scopes depending on the objectives and criteria of testing. The other options are not examples of black-box dynamic testing. Code inspection is an example of white-box static testing. White-box testing (also known as structure-based testing) is a type of testing that considers the internal structure or implementation of the system under test. Static testing is a type of testing that does not involve executing the system under test, but rather analyzing it for defects, errors, or violations of standards. Code inspection is a type of white-box static testing that involves examining the source code of the system under test for quality, readability, maintainability, etc. Checking memory leaks for a program by executing it is an example of white-box dynamic testing. Memory leaks are defects that occur when a program fails to release memory that it has allocated but no longer needs. Checking memory leaks for a program by executing it requires knowledge and access to the internal structure or implementation of the program, such as memory allocation and deallocation mechanisms, pointers, references, etc. Coverage analysis is an example of white-box static testing. Coverage analysis is a technique that measures how much of the code or structure of the system under test has been exercised by a test suite. Coverage analysis requires knowledge and access to the internal structure or implementation of the system under test, such as statements, branches, paths, conditions, etc. Verified References: A Study Guide to the ISTQB® Foundation Level 2018 Syllabus - Springer, page 7.

**NEW QUESTION 107**

Which of the following BEST explains a drawback of independent testing?

- A. Having the business organization participate as an independent test team can hurt the overall testing effort since business participants are often not trained nor experienced in testing
- B. Due to their differing backgrounds and perspectives, an independent test team may discover defects which the developers did not uncover
- C. An independent test team may be isolated from the rest of the development and project team
- D. An independent test team may possess specializations in specific test types such as usability or security which detract from the overall effectiveness of the test team

**Answer: C**

**Explanation:**

Independent testing offers several advantages, such as unbiased testing and detection of different defects. However, a drawback is that an independent test team may be isolated from the development team and project team. This can lead to communication gaps, reduced collaboration, and a lack of understanding of the project context.

According to the ISTQB Certified Tester Foundation Level (CTFL) syllabus v4.0, an independent test team may not have the same level of understanding of the system as the development team, leading to potential issues in communication and integration (ISTQB not-for-profit association).

References:

? Certified Tester Foundation Level v4.0  
? ISTQB Foundation Level Syllabus 4.0 (2023)

**NEW QUESTION 109**

Which of the following statements about test estimation approaches is CORRECT?

- A. The Wideband Delphi estimation technique is an example of the risk-based approach
- B. The Wideband Delphi estimation technique is an example of the expert-based approach
- C. Burndown charts used in Agile development is an example of the risk-based approach
- D. Burndown charts used in Agile development is an example of the expert-based approach

**Answer: B**

**Explanation:**

There are two main approaches to test estimation:

? Expert-based approach:

? Metrics-based approach:

According to the ISTQB Certified Tester Foundation Level (CTFL) v4.0 syllabus, the expert-based approach relies on experts' experience and knowledge, which aligns with the Wideband Delphi technique<sup>6†source</sup>.

References:

? Certified Tester Foundation Level v4.0

? ISTQB Foundation Level Syllabus 4.0 (2023)

**NEW QUESTION 111**

Which of the following statements is not correct?

- A. Looking for defects in a system may require Ignoring system details
- B. Identifying defects may be perceived as criticism against product
- C. Looking for defects in system requires professional pessimism and curiosity
- D. Testing is often seen as a destructive activity instead of constructive activity

**Answer: A**

**Explanation:**

? Looking for defects in a system does not require ignoring system details, but rather paying attention to them and understanding how they affect the system's quality, functionality, and usability. Ignoring system details could lead to missing important defects or testing irrelevant aspects of the system.

? Identifying defects may be perceived as criticism against product, especially by the developers or stakeholders who are invested in the product's success.

However, identifying defects is not meant to be a personal attack, but rather a constructive feedback that helps to improve the product and ensure its alignment with the requirements and expectations of the users and clients.

? Looking for defects in system requires professional pessimism and curiosity, as testers need to anticipate and explore the possible ways that the system could fail, malfunction, or behave unexpectedly. Professional pessimism means being skeptical and critical of the system's quality and reliability, while curiosity means being eager and interested in finding out the root causes and consequences of the defects.

? Testing is often seen as a destructive activity instead of constructive activity, as it involves finding and reporting the flaws and weaknesses of the system, rather than creating or enhancing it. However, testing is actually a constructive activity, as it contributes to the system's improvement, verification, validation, and optimization, and ultimately to the delivery of a high-quality product that meets the needs and expectations of the users and clients.

**NEW QUESTION 116**

Which statement about use case testing is true?

- A. The test cases are designed to find defects in the data flow.
- B. The test cases are designed to be used by real users, not by professional testers
- C. The test cases are always designed by customers or end users.
- D. The test cases are designed to find defects in the process flow.

**Answer: D**

**Explanation:**

Use case testing is a technique that helps identify test cases that exercise the whole system on a transaction by transaction basis from start to finish. Use cases are descriptions of how users interact with the system to achieve a specific goal. Use case testing is not focused on data flow, but rather on process flow. Use case testing can be performed by professional testers, customers or end users, depending on the context. Use case testing does not require the test cases to be designed by customers or end users, but rather by anyone who has access to the use case specifications. Verified References: A Study Guide to the ISTQB® Foundation Level 2018 Syllabus - Springer, Chapter 4, page 36.

**NEW QUESTION 121**

Which of the following statements about static analysis are FALSE?

- A. Static analysis can be used Instead of dynamic testing.I
- B. Static analysis can uncover defects like security vulnerabilities.II
- C. Static analysis can be used to check conformance to specifications and standard
- D. Static analysis typically detects failures prior to component testing.
- E. II
- F. I
- G. III
- H. II
- I. IV
- J. I, IV

**Answer: D**

**Explanation:**

Static analysis involves analyzing the software's code, design, and structure without executing the program. It can uncover various types of defects, including security vulnerabilities (II) and non-conformance to specifications and standards (III). However, static analysis cannot replace dynamic testing (I), which involves executing the software to observe its behavior under various conditions. Dynamic testing can identify failures that static analysis cannot, such as those related to runtime issues and interaction between different parts of the software. Statement IV is false because static analysis does not detect failures; it detects defects. Failures are observed when the software is executed, which is beyond the scope of static analysis.

**NEW QUESTION 126**

A test manager decided to skip static testing since he believes bugs can be found easily by doing dynamic testing. Was this decision right or wrong?

- A. The decision was wrong
- B. Ensuring quality mandates that static testing is performed after performing the dynamic testing.
- C. The decision was right
- D. Static testing is usually redundant if a product is planned to go through a full-cycle of dynamic testing.
- E. The decision was right
- F. Most of the bugs are easier to identify during the dynamic testing.
- G. The decision was wrong
- H. Static testing can find defects early in the development process, reducing the overall cost of testing and development

**Answer: D**

**Explanation:**

Static testing is a form of testing that does not involve executing the software or system under test. It includes activities such as reviews, inspections, walkthroughs, and analysis of documents, code, and models. Static testing can find defects early in the development process, before they become more expensive and difficult to fix in later stages. Static testing can also improve the quality of the software or system by preventing defects from being introduced in the first place. Static testing can complement dynamic testing, which involves executing the software or system under test and checking the results against expected outcomes. Dynamic testing can find defects that static testing may miss, such as performance, usability, or integration issues. However, dynamic testing alone is not sufficient to ensure quality, as it may not cover all possible scenarios, inputs, or paths. Therefore, a test manager who decides to skip static testing is making a wrong decision, as he or she is ignoring the benefits of static testing and relying solely on dynamic testing, which may not be effective or efficient enough to find and prevent defects. References = ISTQB Certified Tester Foundation Level Syllabus, Version 4.0, 2018, Section 2.1.1, page 14; ISTQB Glossary of Testing Terms, Version 4.0, 2018, page 36; ISTQB CTFL 4.0 - Sample Exam - Answers, Version 1.1, 2023, Question 3, page 9.

**NEW QUESTION 130**

Which of the following is a key characteristic of informal reviews?

- A. Kick-off meeting
- B. Low cost
- C. Individual preparation
- D. Metrics analysis

**Answer: B**

**Explanation:**

A key characteristic of informal reviews is low cost. Informal reviews are a type of review that does not follow a formal process or have any formal documentation. Informal reviews are usually performed by individuals or small groups of peers or colleagues who have some knowledge or interest in the product under review. Informal reviews can be done at any time and for any purpose, such as checking for errors, clarifying doubts, sharing ideas, etc. Informal reviews have low cost, as they do not require much time, effort, or resources to conduct. The other options are not key characteristics of informal reviews. Kick-off meeting is a characteristic of formal reviews, such as inspections or walkthroughs. Kick-off meeting is a meeting that is held before the review process starts, where the roles and responsibilities of the participants are defined, the objectives and scope of the review are agreed, and the logistics and schedule of the review are planned. Individual preparation is a characteristic of formal reviews, such as inspections or walkthroughs. Individual preparation is an activity that is performed by the reviewers before the review meeting, where they examine the product under review and identify any issues or questions that need to be discussed or resolved during the review meeting. Metrics analysis is a characteristic of formal reviews, such as inspections or walkthroughs. Metrics analysis is an activity that is performed after the review process is completed, where the data and results of the review are collected and analyzed to measure the effectiveness and efficiency of the review, as well as to identify any improvement actions or lessons learned for future reviews. Verified References: A Study Guide to the ISTQB® Foundation Level 2018 Syllabus - Springer, page 9.

**NEW QUESTION 132**

A software application incorrectly provided customers discounts of 50% off their total purchases if the purchases exceeded \$100. It was discovered through an audit that the discount should have been only 5% off these purchases. A root cause analysis uncovered that the requirements incorrectly stated 50% instead of 5% in this scenario.

Which of the following MOST accurately reflects this scenario?

- A. The audit finding is the root cause, the incorrect calculation of 50% is the defect, and the incorrect requirement is the effect
- B. The incorrect customer discount is the effect and the reason for the requirement error is the root cause
- C. The incorrect discount is the root cause, requiring a root cause analysis which led to investigating the software code, design, and requirements
- D. A defect in the code is determined to be the root cause of the incorrect calculation

**Answer: B**

**Explanation:**

According to the ISTQB CTFL syllabus, a defect is a deviation from the expected result which in this scenario is the incorrect discount applied to the customers. The root cause, as per the ISTQB definition, is the originating cause of a defect, which in this case is the incorrect requirement stating 50% instead of 5%. Therefore, the incorrect requirement is the root cause and the customer receiving the wrong discount is the effect of this root cause. References: ISTQB Certified Tester Foundation Level Syllabus v4.0, Section 1.4.3 "Defects, Root Causes, and Effects".

**NEW QUESTION 136**

A mid-size software product development company has analyzed data related to defects detected in its product and found out that defects fixed in earlier builds are getting re-opened after a few months.



The company management now seeks your advice in order to reverse this trend and prevent re-opening of defects fixed earlier. What would be your FIRST recommendation to the company?

- A. Automate existing test suits so that lesser time is spent on execution of each test, and more tests can be executed in the available time thus leading to a lower probability of defects slipping by
- B. Verify existing regression test suite are adequate, and augment it, if required, in order to ensure that defects fixed earlier get re-tested in each subsequent build
- C. Analyze the product modules containing maximum defects, and get them thoroughly tested and defects fixed as a one-time activity
- D. If required, train the teams responsible for development and testing of the modules containing maximum number of defects, and if this does not help, replace them with more knowledgeable people

**Answer: B**

**Explanation:**

Regression testing is a type of testing that verifies that previously tested software still performs correctly after changes. Regression testing can help prevent re-opening of defects fixed earlier by ensuring that they do not cause any new failures or side effects. The first recommendation to the company is to verify existing regression test suite are adequate, and augment it, if required, in order to ensure that defects fixed earlier get re-tested in each subsequent build. This can help improve the coverage and effectiveness of regression testing and detect any regression defects as soon as possible. Automating existing test suites may also help reduce the time and effort required for regression testing, but this is not the first recommendation, as automation may not be feasible or cost-effective for all test cases. Analyzing the product modules containing maximum defects and getting them thoroughly tested and defects fixed as a one-time activity may also help reduce the defect density and improve the quality of those modules, but this is not the first recommendation, as it does not address the root cause of re-opening defects fixed earlier. Training or replacing the teams responsible for development and testing of the modules containing maximum number of defects may also help improve their skills or performance, but this is not the first recommendation, as it may not be necessary or appropriate for all teams. Verified References: [A Study Guide to the ISTQB® Foundation Level 2018 Syllabus - Springer], Chapter 2, page 19; Chapter 4, page 45.

**NEW QUESTION 141**

A system has valid input numbers ranging between 1000 and 99999 (both inclusive). Which of the following inputs are a result of designing tests for all valid equivalence classes and their boundaries?

- A. 999.1000.23232.99999.100000
- B. 999.1000.50000.100000.100001
- C. 999.100000
- D. 1000,50000,99999

**Answer: B**

**Explanation:**

A correct list of boundary values for the P input should include the minimum and maximum values of the valid range (15 and 350), as well as the values just below and above the boundaries (14 and 351). Boundary value analysis is a test design technique that involves testing the values at or near the boundaries of an input domain or output range, as these values are more likely to cause errors than values in the middle. Option B satisfies this condition, as it has all four boundary values (14, 15, 350, 351). Option A has two values from the same equivalence class (1000 and 99999), option C has two values outside the range (999 and 100000), and option D has no boundary values at all. Verified References: A Study Guide to the ISTQB® Foundation Level 2018 Syllabus - Springer, page 34.

**NEW QUESTION 142**

Which of the following BEST describes a test summary report for executive-level employees

- A. The report is detailed and includes a status summary of defects by priority or budget
- B. The report is detailed and includes specific information on defects and trends
- C. The report is high-level and includes a status summary of defects by priority or budget
- D. The report is high-level and includes specific information on defects and trends

**Answer: C**

**Explanation:**

For executive-level employees, a test summary report should be concise and focus on high-level information. It typically includes a summary of defects categorized by priority or budget. Executives are generally interested in the overall status and the impact on critical business objectives rather than detailed technical information. The report should provide an overview of the most important aspects of testing, such as key issues, test progress, and any risks or concerns that could affect project outcomes. References:  
? ISTQB Certified Tester Foundation Level Syllabus v4.0, Section 5.3.1.

**NEW QUESTION 145**

Which of the following exemplifies how a software bug can cause harm to a company?

- A. "Print" prints the last page twice for a file with 1000 pages
- B. The timeout on the login page of a web site is 9 minutes, while the requirement was for 10 minutes
- C. When uninstalling the application, the uninstall dialog has a spelling mistake
- D. When calculating the final price in a shopping list, the price of the last item is not added

**Answer: D**

**Explanation:**

A software bug can cause harm to a company by directly affecting its operations, reputation, user satisfaction, and financials. Option D, "When calculating the final price in a shopping list, the price of the last item is not added," describes a defect that directly impacts the core functionality of a financial transaction, potentially leading to financial loss and customer dissatisfaction. This can have severe implications for the company's credibility and revenue. Options A, B, and C describe bugs that, while potentially annoying, do not have the same direct impact on the company's core operations and financial integrity as option D.

**NEW QUESTION 149**

What is 'Component Testing'?

- A. Integration Testing



- B. Functional testing
- C. Experience-based testing
- D. A test level

**Answer:** D

**Explanation:**

Component testing is a test level. A test level is a group of test activities that are organized and managed together based on some common characteristics or objectives. A test level can be defined based on various factors, such as the scope and target of testing, the phase and model of development, the stakeholders and roles involved in testing, etc. Component testing (also known as unit testing or module testing) is a test level that focuses on verifying the functionality and quality of individual software components (such as modules, classes, functions, methods, etc.). Component testing can be performed by developers or testers using various techniques and tools depending on the type and complexity of the components. The other options are not test levels. Integration testing is another test level that focuses on verifying the functionality and quality of groups of software components that interact with each other or with external systems. Functional testing is a type of black-box dynamic testing that verifies that the system under test performs its intended functions according to its requirements or specifications. Experience-based testing is a category of test design techniques that rely on the tester's knowledge and intuition to derive and select test cases based on their experience with similar systems, technologies, domains, risks, etc. Verified References: A Study Guide to the ISTQB® Foundation Level 2018 Syllabus - Springer, page 19.

**NEW QUESTION 152**

You are testing the download process of a mobile phone application.  
For which to the following capabilities to the system you need to design a nonfunctional test?

- A. It was easy to locate, download and install the application
- B. The application was correctly downloaded
- C. The application created an installation log file in a given folder
- D. The application installed only after the user's approval

**Answer:** A

**Explanation:**

This question asks for a non-functional aspect of testing the download process of a mobile application. Option A, "It was easy to locate, download and install the application," refers to usability, which is a non-functional quality attribute. Non-functional testing involves testing the system's attributes, such as usability, performance, reliability, etc., rather than specific behaviors or functions. Options B, "The application was correctly downloaded," C, "The application created an installation log file in a given folder," and D, "The application installed only after the user's approval," describe functional aspects, focusing on what the software does rather than how it performs or is experienced by the user.

**NEW QUESTION 153**

Which of the following is MOST likely to be an example of a PROJECT risk?

- A. A computation is not always performed correctly in some situations
- B. A system architecture may not support some non-functional requirements
- C. Team members' skills may not be sufficient for the assigned work
- D. Specific modules do not adequately meet their intended functions according to the user

**Answer:** C

**Explanation:**

A project risk relates to potential issues that could affect the overall success of the project. Among the options provided, the risk that "Team members' skills may not be sufficient for the assigned work" is clearly a project risk because it pertains to the potential failure of the project due to inadequate skillsets among the team. This risk affects the entire project's ability to meet its objectives. References:  
? ISTQB Certified Tester Foundation Level Syllabus v4.0, Section 1.4.2.

**NEW QUESTION 154**

Which of the following is an INCORRECT statement about the benefit of traceability between the test basis and test work products?

- A. Traceability may be required by IT governance rules
- B. Traceability may help evaluate the extent of test coverage
- C. Traceability may allow testing to be auditable
- D. Traceability may make it harder to understand the impact of changes

**Answer:** D

**Explanation:**

The statement "Traceability may make it harder to understand the impact of changes" is incorrect. Traceability in testing actually facilitates understanding the impact of changes by linking test cases to requirements. This linkage helps ensure that any changes in the requirements are adequately reflected and verified in the test cases, thus supporting effective management of changes and maintaining the integrity of the system or product being developed (ISTQB not-for-profit association) (ISTQB Main Web). References:  
? ISTQB® Certified Tester Foundation Level Syllabus v4.0: [https://istqb-main-web-prod.s3.amazonaws.com/media/documents/ISTQB\\_CTFL\\_Syllabus-v4.0.pdf](https://istqb-main-web-prod.s3.amazonaws.com/media/documents/ISTQB_CTFL_Syllabus-v4.0.pdf)  
? ISTQB Official Website - CTFL Certification: <https://www.istqb.org/certifications/certified-tester-foundation-level/>

**NEW QUESTION 158**

A company runs a pilot project for evaluation of a test automation tool. Which of the following is NOT a valid object of this pilot project?

- A. Get familiar with the functionality and options of the tool
- B. Check how the tool fits to the existing test processes
- C. Train all testers on using the tool
- D. Decide upon standards for tool implementation

**Answer:** C

**Explanation:**

? A pilot project is a small-scale experiment or trial that is conducted to evaluate the feasibility, effectiveness, and suitability of a test automation tool before implementing it on a larger scale<sup>1</sup>.

? The objectives of a pilot project may vary depending on the context and scope of the test automation initiative, but some common ones are<sup>2</sup>:

? Therefore, option C is not a valid objective of a pilot project, as it is not necessary to train all testers on using the tool at this stage. Training all testers on using the tool would be more appropriate after the tool has been selected and approved for full-scale implementation, and after the standards and guidelines have been established. Training all testers on using the tool during the pilot project would be inefficient, costly, and premature, as the tool may not be suitable or effective for the intended purpose, or may be replaced by another tool later.

References:

? 1: ISTQB Certified Tester Foundation Level Syllabus 2018, Version 4.0, p. 82

? 2: ISTQB Certified Tester Foundation Level Syllabus 2018, Version 4.0, p. 83

? : ISTQB Certified Tester Foundation Level Syllabus 2018, Version 4.0, p. 84

? : ISTQB Certified Tester Foundation Level Syllabus 2018, Version 4.0, p. 85

**NEW QUESTION 161**

In which of the following cases you would NOT execute maintenance testing?

- A. Retirement of the software or system
- B. Modifications to a released software or system
- C. Migration of the system data to a replacement system
- D. Update to the Maintainability requirements during the development phase

**Answer: D**

**Explanation:**

Maintenance testing is testing performed on a software product after delivery to correct defects or improve performance or other attributes. Maintenance testing can be triggered by various situations, such as modifications to a released software or system, migration of the system data to a replacement system, or retirement of the software or system. Maintenance testing is not executed when there is an update to the maintainability requirements during the development phase, as this is not a maintenance situation but rather a change request that should be handled by the development process. Verified References: [A Study Guide to the ISTQB® Foundation Level 2018 Syllabus - Springer], Chapter 2, page 18-19.

**NEW QUESTION 166**

Given the following state model of sales order software: SEE ATTACHMENT

Which of the following sequences of transitions provides the highest level of transition coverage for the model (assuming you can start in any state)?

- A. IN PRODUCTION -> CANCELLED -> PLACED -> IN PRODUCTION -> CANCELLED -> PLACED
- B. IN PRODUCTION -> SHIPPED -> INVOICED -> CANCELLED -> PLACED -> IN PRODUCTION
- C. PLACED -> IN PRODUCTION -> SHIPPED -> CANCELLED -> PLACED
- D. PLACED -> CANCELLED -> PLACED -> CANCELLED -> PLACED -> IN PRODUCTION -> CANCELLED

**Answer: B**

**Explanation:**

To achieve the highest level of transition coverage, one must consider all the possible transitions between the states in the given state model of the sales order software. The transitions in the sequence provided in Option B - "IN PRODUCTION -> SHIPPED -> INVOICED -> CANCELLED -> PLACED -> IN PRODUCTION" cover all the states and transitions effectively. This covers the transitions from IN PRODUCTION to SHIPPED, SHIPPED to INVOICED, INVOICED to CANCELLED, CANCELLED to PLACED, and

PLACED to IN PRODUCTION, thereby maximizing the transition coverage. References:

? ISTQB Certified Tester Foundation Level Syllabus v4.0, Section 4.3.5.

**NEW QUESTION 169**

Which of the following activities does NOT belong to a typical technical review?

- A. Pre-meeting preparation by reviewers
- B. Using checklists during the meeting
- C. Inviting end-users to the meeting
- D. Preparation of a review report

**Answer: C**

**Explanation:**

Technical reviews are structured meetings that aim to examine various aspects of a product or project to identify any defects or improvements. Options A (Pre-meeting preparation by reviewers), B (Using checklists during the meeting), and D (Preparation of a review report) are typical activities in a technical review process. Inviting end-users to the meeting (C), however, is generally not part of a typical technical review, as these reviews are usually more focused on the technical aspects and are conducted by peers or experts within the development or testing teams rather than end-users.

**NEW QUESTION 170**

Which of the following CORRECTLY matches the roles and responsibilities in a formal review?

- A. Facilitator - Fixes defects in the work product under review
- B. Scribe - Collates potential defects found during the individual review activity
- C. Review Leader - Creates the work product under review
- D. Author - Identify potential defects in the work product under review

**Answer: B**

**Explanation:**

In formal reviews, the scribe's role is to collate potential defects and other findings during the review process. This position is crucial as it ensures all observations and defects are recorded accurately, facilitating efficient analysis and resolution of issues identified during the review. References: ISTQB Certified Tester

Foundation Level Syllabus v4.0, Section 3.2.4 "Roles and Responsibilities in a Formal Review".

**NEW QUESTION 172**

Which of the following activities is NOT a part of the fundamental testing process?

- A. Archiving automation code
- B. Test status reporting
- C. Test process improvement
- D. Build release and maintenance

**Answer: D**

**Explanation:**

The fundamental testing process includes activities that are directly related to the planning, preparation, execution, and evaluation of tests, as well as the closure activities of the testing phase. Option D, "Build release and maintenance," falls outside the scope of the fundamental testing process as it relates more to software development and operations rather than specific testing activities. Options A, "Archiving automation code," B, "Test status reporting," and C, "Test process improvement," are all activities that can be part of or associated with the fundamental testing process. Archiving automation code is part of test closure, test status reporting is part of test monitoring and control, and test process improvement can be an outcome of test closure activities.

**NEW QUESTION 173**

Which of the following statements about decision tables are TRUE?

- A. Generally, decision tables are generated for low risk test items.I
- B. Test cases derived from decision tables can be used for component tests.II
- C. Several test cases can be selected for each column of the decision table.I
- D. The conditions in the decision table represent negative tests generally.
- E. III
- F. I, IV
- G. I
- H. IV
- I. I
- J. III

**Answer: D**

**Explanation:**

A decision table is a technique that 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). The following statements about decision tables are true:

? II. Test cases derived from decision tables can be used for component tests.

Decision tables can be used to test components that have multiple inputs and outputs that depend on logical combinations of conditions. Decision tables can help cover all possible combinations or scenarios in a systematic way.

? III. Several test cases can be selected for each column of the decision table. A column of a decision table represents a unique combination of condition entries and action entries. Several test cases can be selected for each column by varying other input values or expected results that are not part of the decision table. The following statements about decision tables are false:

? I. Generally, decision tables are generated for low risk test items. Decision tables are not related to risk level, but rather to complexity level. Decision tables are generated for test items that have complex logic or multiple conditions and actions that need to be tested.

? IV. The conditions in the decision table represent negative tests generally. The conditions in the decision table represent both positive and negative tests, depending on whether they are valid or invalid inputs for the test item. Verified References: A Study Guide to the ISTQB® Foundation Level 2018 Syllabus - Springer, Chapter 4, page 42-43.

**NEW QUESTION 176**

Which of the following BEST describes exploratory testing?

- A. Exploratory testing is a suitable test technique which may replace both black-box and white-box test techniques
- B. Exploratory testing is a valid and useful black-box test technique since it focuses on test cases related to the architecture and design of a system
- C. Exploratory testing requires both solid specifications and much project time available for test execution
- D. Exploratory testing may be used within defined time periods, during which the tester may follow a test charter as a guide

**Answer: D**

**Explanation:**

Exploratory testing involves simultaneous test design and execution and is guided by a test charter, which outlines what needs to be tested, how it should be tested, and what to look for. This technique is typically conducted within predefined time periods, known as time-boxing, which allows testers to explore a system, understand its functionalities, and identify potential issues without detailed documentation or prior test case planning. The key aspects of exploratory testing include flexibility, adaptability, and the ability to respond to system behavior during testing. References:

? ISTQB Certified Tester Foundation Level Syllabus v4.0, Section 4.4.2.

**NEW QUESTION 181**

Which of the following statements describes regression testing?

- A. Retesting of a fixed defectI
- B. Testing of an already tested programII
- C. Testing of new functionality in a programI
- D. Regression testing applies only to functional testingV Tests that do not have to be repeatable, because They are only used once
- E. II, IV, V
- F. I, III, IV
- G. II
- H. I, IV

**Answer:** C

**Explanation:**

Regression testing is the re-running of functional and non-functional tests to ensure that previously developed and tested software still performs as expected after a change<sup>1</sup> It does not involve retesting of a fixed defect, testing of new functionality, or applying only to functional testing. Tests that are used for regression testing should be repeatable, because they are used to verify the stability of the software after each change<sup>2</sup> References = ISTQB Certified Tester Foundation Level (CTFL) v4.0 Syllabus, Chapter 4, Section 4.2.2, Page 291; ISTQB Glossary of Testing Terms v4.0, Page 292

**NEW QUESTION 185**

Which of the following statements about independent testing is WRONG?

- A. Independent testing is necessary because developers don't know any testing.
- B. Independent testing is best suited for the system test level.
- C. A certain degree of independence makes the tester more effective at finding defects.
- D. Independent test teams may find other types of defects than developers who are familiar with the system's structure.

**Answer:** A

**Explanation:**

Independent testing is testing performed by a person or group that is independent of the development team. Independent testing can have various degrees of independence, ranging from testers who are part of the same organization as developers to testers who are external contractors or consultants. Independent testing can have various benefits, such as reducing bias, increasing objectivity, improving quality, or providing different perspectives. Independent testing is not necessary because developers don't know any testing, as this is a wrong and disrespectful statement. Developers can perform various types of testing, such as unit testing, component testing, or integration testing. However, independent testing can complement developer testing by providing additional levels of verification and validation, such as system testing, acceptance testing, or non-functional testing. Verified References: [A Study Guide to the ISTQB® Foundation Level 2018 Syllabus - Springer], Chapter 2, page 16-17.

**NEW QUESTION 190**

Which of the following would be the LEAST likely to be used as the basis for a test exit criteria?

- A. Test schedules
- B. Cost of testing performed so far
- C. Confidence of testers in tested code
- D. Number of unfixed defects

**Answer:** A

**Explanation:**

Test exit criteria are the conditions or requirements that must be met before testing can be concluded. Test exit criteria are usually defined in the test plan and agreed by the stakeholders. Test exit criteria can be based on various factors, such as test coverage, defect status, quality level, risk level, etc. Test schedules would be the least likely to be used as the basis for test exit criteria, because test schedules are not directly related to the quality or performance of the software product, but rather to the time or resources allocated for testing. Test schedules can be used as the basis for test entry criteria, which are the conditions or requirements that must be met before testing can start. The other options are more likely to be used as the basis for test exit criteria. Cost of testing performed so far can be used as a basis for test exit criteria, because it can indicate the return on investment or the cost-benefit ratio of testing. Confidence of testers in tested code can be used as a basis for test exit criteria, because it can reflect the level of satisfaction or assurance of the testers about the quality or reliability of the software product. Number of unfixed defects can be used as a basis for test exit criteria, because it can indicate the level of risk or impact of the remaining defects on the software product. Verified References: A Study Guide to the ISTQB® Foundation Level 2018 Syllabus - Springer, page 13.

**NEW QUESTION 195**

You need to test the login page of a web site. The page contains fields for user name and password. Which test design techniques are most appropriate for this case?

- A. Decision table testing, state transition testing.
- B. Equivalence partitioning, Boundary value analysis.
- C. Exploratory testing, statement coverage.
- D. Decision coverage, fault attack.

**Answer:** B

**Explanation:**

Equivalence partitioning and boundary value analysis are test design techniques that are most appropriate for testing the login page of a web site. The page contains fields for user name and password, which are input values that can be divided into partitions of equivalent data. Equivalence 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 partitioning can be used to reduce the number of test cases by selecting one representative value from each partition. 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 detect defects caused by incorrect handling of boundary conditions. For example, for testing the user name field, we can identify two equivalence partitions: valid user name (existing and correct) and invalid user name (non-existing or incorrect). The boundary values for these partitions are the minimum and maximum length of user name allowed by the system.

Decision table testing and state transition testing are not suitable for testing the login page of a web site, as they are more applicable for testing components that have multiple inputs and outputs that depend on logical combinations of conditions or events. Decision table testing is a technique that shows combinations of inputs and/or stimuli (causes) with their associated outputs and/or actions (effects). State transition testing is a technique that models how a system transitions from one state to another depending on events or conditions.

Exploratory testing and statement coverage are not suitable for testing the login page of a web site, as they are more applicable for testing components that require learning, creativity and intuition or structural analysis. Exploratory testing is an approach to testing that emphasizes learning, test design and test execution at the same time. Exploratory testing relies on the tester's skills, creativity and intuition to explore the software under test and discover defects. Statement coverage is a type of structural testing that measures how many statements in a program have been executed by a test suite. Statement coverage can be used to assess the adequacy or completeness of a test suite.

Decision coverage and fault attack are not suitable for testing the login page of a web site, as they are more applicable for testing components that have complex logic or potential errors. Decision coverage is a type of structural testing that measures how many decision outcomes in a program have been executed by a test



suite. Decision coverage can be used to assess the adequacy or completeness of a test suite. Fault attack is a type of functional testing that deliberately introduces faults into a system in order to provoke failures or errors. Verified References: [A Study Guide to the ISTQB® Foundation Level 2018 Syllabus - Springer], Chapter 4, page 34-46; Chapter 5, page 47-48.

**NEW QUESTION 199**

You are responsible for applying the correct technique for a review of the requirements document for a project to develop a new software application. You identify the reviewers and the required roles, including the meeting leader, who is the requirements document author, and a separate role for a scribe. Additionally, you decide to take a relatively informal approach to the requirements review. The goal of the review is to find defects in the requirements document, such as omissions, Inconsistencies, and duplications. Another goal of the review is to improve the software application's usability and accessibility by considering the various stakeholders' viewpoints.

Which of the following statements BEST describes this scenario?

- A. This scenario is using a pair review type and a perspective-based review technique
- B. This scenario is using a walkthrough review type and a checklist-based review technique
- C. This scenario is using a walkthrough review type and a perspective-based review technique
- D. This scenario is using a pair review type and a checklist-based review technique

**Answer: C**

**Explanation:**

This scenario is using a walkthrough review type and a perspective-based review technique. In a walkthrough, the author of the document leads the meeting and it typically includes a meeting leader and a scribe, as described. This type of review is informal, focuses on discussion, and often involves scenario-based reading of the document to understand different user perspectives (ISTQB Main Web). References:

? ISTQB® Certified Tester Foundation Level Syllabus v4.0: ISTQB CTFL Syllabus v4.0 PDF

**NEW QUESTION 200**

A software calculates the annual car tax using three inputs:

- E; the emission level of the vehicle
- P: the power of the vehicle
- T the type of the vehicle

The input value for P can be integer positive values between 15 and 350.

Which of the following answers contains a correct list of a boundary values for the P input?

- A. 14,351
- B. 14,15,350,351
- C. 15,350
- D. 5.175.500

**Answer: B**

**Explanation:**

A correct list of boundary values for the P input should include the minimum and maximum values of the valid range (15 and 350), as well as the values just below and above the boundaries (14 and 351). Boundary value analysis is a test design technique that involves testing the values at or near the boundaries of an input domain or output range, as these values are more likely to cause errors than values in the middle. Option B satisfies this condition, as it has all four boundary values (14, 15, 350, 351). Option A has only two boundary values (14 and 351), option C has only two boundary values (15 and 350), and option D has no boundary values at all. Verified References: A Study Guide to the ISTQB® Foundation Level 2018 Syllabus - Springer, page 34.

**NEW QUESTION 203**

In what way do Configuration Management effects testing?

- A. Without proper configuration management, test planning cannot proceed.
- B. Proper configuration management ensures that testers can uniquely identify the tested item
- C. Configuration management is important for developers, not for testers
- D. There is very little influence of configuration management practices on the test project.

**Answer: B**

**Explanation:**

Configuration management is a process that establishes and maintains consistency among work products throughout their life cycle. Configuration management affects testing in various ways, such as:

? Proper configuration management ensures that testers can uniquely identify the tested item, which can help traceability, reproducibility and accountability.

? Proper configuration management ensures that testers have access to consistent versions of software components and testware, which can help reliability, compatibility and efficiency.

? Proper configuration management ensures that testers can track changes and defects in software components and testware, which can help verification, validation and reporting.

? Proper configuration management ensures that testers can control the configuration of the test environment, which can help stability, security and performance.

Configuration management is not a prerequisite for test planning, as test planning can proceed without configuration management, although it may be less effective or accurate. Configuration management is not important for developers only, but for testers as well, as it affects the quality and consistency of the testing process and products. Configuration management has a significant influence on the test project, as it affects various aspects of testing, such as traceability, reproducibility, reliability, compatibility, efficiency, verification, validation, reporting, stability, security and performance. Verified References: A Study Guide to the ISTQB® Foundation Level 2018 Syllabus - Springer, Chapter 6, page 60-61.

**NEW QUESTION 205**

Which of the following is a CORRECT statement about how a tester should communicate about defects, test results, and other test information?

- A. Testers should include personal opinions and judgements in defect reports and review findings
- B. Testers should emphasize the benefits of testing, such as increased quality and reduced risk
- C. Testers should reject all questions about their test findings and information
- D. Testers should take a command-and-control approach with the project team

**Answer:** B

**Explanation:**

Communication from testers about defects, test results, and other test information should emphasize the benefits of testing such as increased quality and reduced risk. This positive framing helps in reinforcing the value of testing and ensuring stakeholders understand the contribution of testing to the overall project success (ISTQB not-for-profit association).References:

? ISTQB® Certified Tester Foundation Level Syllabus v4.0: [https://istqb-main-web-prod.s3.amazonaws.com/media/documents/ISTQB\\_CTFL\\_Syllabus-v4.0.pdf](https://istqb-main-web-prod.s3.amazonaws.com/media/documents/ISTQB_CTFL_Syllabus-v4.0.pdf)

**NEW QUESTION 209**

Which of the following does MOT describe a reason why testing is necessary?

- A. The customer decided that 100% branch coverage shall be achieved
- B. The acquisition of test automation tools was based on the assumption that it will be used in all projects
- C. For avionics and pharmaceutical systems software testing is mandated by standards
- D. The risks associated with delivering the system are far higher than the cost of testing

**Answer:** B

**Explanation:**

Testing is necessary for various reasons, such as:

? To detect defects and failures that may affect the quality, performance, reliability or security of a software product or system

? To verify that a software product or system meets its specified requirements, expectations and standards

? To validate that a software product or system fulfills its intended purpose and satisfies its stakeholders' needs

? To provide information and feedback about the status and risks of a software product or system

? To comply with regulations or contractual obligations that mandate testing for certain types of software products or systems The following statements describe some reasons why testing is necessary:

? A) The customer decided that 100% branch coverage shall be achieved. This is a reason why testing is necessary, as it reflects a contractual obligation or a quality standard that requires testing to measure and achieve a certain level of code coverage.

? C) For avionics and pharmaceutical systems software testing is mandated by standards. This is a reason why testing is necessary, as it reflects a regulation or a compliance requirement that mandates testing for certain types of software products or systems that have high safety or security risks.

? D) The risks associated with delivering the system are far higher than the cost of testing. This is a reason why testing is necessary, as it reflects a risk-based approach that considers testing as an investment to reduce the probability and impact of potential failures or defects. The following statement does not describe a reason why testing is necessary:

? B) The acquisition of test automation tools was based on the assumption that it will be used in all projects. This is not a reason why testing is necessary, as it reflects a business decision or a resource allocation that does not justify the need or purpose of testing. Test automation tools are not always suitable or beneficial for all projects, and testing can be performed with or without test automation tools. Verified References: [A Study Guide to the ISTQB® Foundation Level 2018 Syllabus - Springer], Chapter 1, page 5-6.

**NEW QUESTION 210**

Which ONE of the following statements about acceptance testing is NOT correct?

- A. Testing of disaster recovery and backup/restore is usually NOT part of acceptance testing.
- B. The customers or system users are often responsible for the acceptance testing.
- C. The main goal of acceptance testing is to build confidence in the system, not find defects.
- D. Acceptance testing is the last level of testing performed prior to system release.

**Answer:** A

**Explanation:**

Acceptance testing is a level of testing performed to verify that a software product meets the agreed acceptance criteria and is acceptable for delivery. Acceptance testing is often performed by the customers or system users, who are the main stakeholders of the software product. The main goal of acceptance testing is to build confidence in the system, not find defects, as defects should have been detected and fixed in earlier levels of testing. Acceptance testing is the last level of testing performed prior to system release, unless there are any changes or fixes that require re-testing. Testing of disaster recovery and backup/restore is usually part of acceptance testing, as these are important aspects of system reliability and security that affect the customer satisfaction and trust. Therefore, statement A is not correct, while statements B, C and D are correct. Verified References: A Study Guide to the ISTQB® Foundation Level 2018 Syllabus - Springer, Chapter 2, page 20-21.

**NEW QUESTION 213**

Which of the following tools is most likely to detect defects in functions or methods in source code?

- A. configuration management tool
- B. unit test framework tool
- C. test design tool
- D. monitoring tool

**Answer:** B

**Explanation:**

A unit test framework tool is a tool that supports the creation, execution, and reporting of unit tests, which are tests that verify the functionality and quality of individual software components (such as functions or methods) in source code. A unit test framework tool can help to detect defects in functions or methods in source code by providing features such as test case generation, test case execution, test result comparison, test coverage measurement, etc. Some examples of unit test framework tools are JUnit, NUnit, TestNG, etc. The other options are not tools that are likely to detect defects in functions or methods in source code. A configuration management tool is a tool that supports the management and control of different versions and variants of software products or components. A test design tool is a tool that supports the design and generation of test cases based on some criteria or rules. A monitoring tool is a tool that monitors the behavior or performance of a system or component under test. Verified References: A Study Guide to the ISTQB® Foundation Level 2018 Syllabus - Springer, page 10.

**NEW QUESTION 215**

Which of the following statements is an example of testing contributing to higher quality?

- A. A test leader writes a test summary report
- B. A project manager asks to a test leader to estimate the test effort
- C. A tester installs a test ten in the lest environment
- D. A tester finds a bug which is resolved prior to release

**Answer:** D

**Explanation:**

? The question is about identifying an example of testing contributing to higher quality. Quality is the degree to which a component, system or process meets specified requirements and/or user/customer needs and expectations<sup>1</sup>. Testing is the process consisting of all lifecycle activities, both static and dynamic, concerned with planning, preparation and evaluation of software products and related work products to determine that they satisfy specified requirements, to demonstrate that they are fit for purpose and to detect defects<sup>2</sup>.

? Therefore, testing contributes to higher quality by verifying and validating that the software products and related work products meet the specified requirements, are fit for purpose and have no defects, or at least have a reduced number of defects. Testing also provides information about the quality of the software products and related work products to the stakeholders, who can make informed decisions based on the test results<sup>3</sup>.

? Out of the four given statements, only option D is an example of testing contributing to higher quality, as it shows that testing has detected a defect (a flaw in a component or system that can cause the component or system to fail to perform its required function<sup>4</sup>) and that the defect has been resolved (fixed and confirmed) prior to release (delivery of the software product to the customer or end user). This means that testing has prevented a potential failure (an event in which a component or system does not perform a required function within specified limits) from occurring in the operational environment, and thus has improved the quality of the software product.

? Option A is not an example of testing contributing to higher quality, as it is a reporting activity that summarizes the test results and evaluates the test objectives, but does not directly affect the quality of the software product or related work products. A test summary report is a document that records and communicates the outcomes of testing activities, including test completion criteria, test results, incident reports, test summary and evaluation, and lessons learned.

? Option B is not an example of testing contributing to higher quality, as it is a planning activity that estimates the resources and time needed for testing activities, but does not directly affect the quality of the software product or related work products. A test effort estimate is an approximation of the amount of work and/or the duration of time required to perform testing activities.

? Option C is not an example of testing contributing to higher quality, as it is a preparation activity that sets up the test environment (an environment containing hardware, instrumentation, simulators, software tools, and other support elements needed to conduct a test), but does not directly affect the quality of the software product or related work products. A test environment installation is a process of installing and configuring the test environment according to the test environment specification. References:

? 1: ISTQB Certified Tester Foundation Level Syllabus 2018, Version 4.0, p. 10

? 2: ISTQB Certified Tester Foundation Level Syllabus 2018, Version 4.0, p. 11

? 3: ISTQB Certified Tester Foundation Level Syllabus 2018, Version 4.0, p. 12

? 4: ISTQB Certified Tester Foundation Level Syllabus 2018, Version 4.0, p. 13

? : ISTQB Certified Tester Foundation Level Syllabus 2018, Version 4.0, p. 13

? : ISTQB Certified Tester Foundation Level Syllabus 2018, Version 4.0, p. 77

? : ISTQB Certified Tester Foundation Level Syllabus 2018, Version 4.0, p. 78

? : ISTQB Certified Tester Foundation Level Syllabus 2018, Version 4.0, p. 79

? : ISTQB Certified Tester Foundation Level Syllabus 2018, Version 4.0, p. 80

? : ISTQB Certified Tester Foundation Level Syllabus 2018, Version 4.0, p. 81

? : ISTQB Certified Tester Foundation Level Syllabus 2018, Version 4.0, p. 82

? : ISTQB Certified Tester Foundation Level Syllabus 2018, Version 4.0, p. 83

? : ISTQB Certified Tester Foundation Level Syllabus 2018, Version 4.0, p. 84

? : ISTQB Certified Tester Foundation Level Syllabus 2018, Version 4.0, p. 85

? : ISTQB Certified Tester Foundation Level Syllabus 2018, Version 4.0, p. 86

? : ISTQB Certified Tester Foundation Level Syllabus 2018, Version 4.0, p. 87

? : ISTQB Certified Tester Foundation Level Syllabus 2018, Version 4.0, p. 88

? : ISTQB Certified Tester Foundation Level Syllabus 2018, Version 4.0, p. 89

? : ISTQB Certified Tester Foundation Level Syllabus 2018, Version 4.0, p. 90

? : ISTQB Certified Tester Foundation Level Syllabus 2018, Version 4.0, p. 91

? : ISTQB Certified Tester Foundation Level Syllabus 2018, Version 4.0, p. 92

? : ISTQB Certified Tester Foundation Level Syllabus 2018, Version 4.0, p. 93

? : ISTQB Certified Tester Foundation Level Syllabus 2018, Version 4.0, p. 94

? : ISTQB Certified Tester Foundation Level Syllabus 2018, Version 4.0, p. 95

? : ISTQB Certified Tester Foundation Level Syllabus 2018, Version 4.0, p. 96

? : ISTQB Certified Tester Foundation Level Syllabus 2018, Version 4.0, p. 97

? : ISTQB Certified Tester Foundation Level Syllabus 2018, Version 4.0, p. 98

? : ISTQB Certified Tester Foundation Level Syllabus 2018, Version 4.0, p. 99

? : ISTQB Certified Tester Foundation Level Syllabus 2018, Version 4.0, p. 100

? : ISTQB Certified Tester Foundation Level Syllabus 2018, Version 4.0, p. 101

? : ISTQB Certified Tester Foundation Level Syllabus 2018, Version 4.0, p. 102

? : ISTQB Certified Tester Foundation Level Syllabus 2018, Version 4.0, p. 103

? : ISTQB Certified Tester Foundation Level Syllabus 2018, Version 4.0, p. 104

? : ISTQB Certified Tester Foundation Level Syllabus 2018, Version 4.0, p. 105

? : ISTQB Certified Tester Foundation Level Syllabus 2018, Version 4.0, p. 106

? : ISTQB Certified Tester Foundation Level Syllabus 2018, Version 4.0, p. 107

**NEW QUESTION 216**

.....

## Thank You for Trying Our Product

\* 100% Pass or Money Back

All our products come with a 90-day Money Back Guarantee.

\* One year free update

You can enjoy free update one year. 24x7 online support.

\* Trusted by Millions

We currently serve more than 30,000,000 customers.

\* Shop Securely

All transactions are protected by VeriSign!

**100% Pass Your ISTQB-CTFL Exam with Our Prep Materials Via below:**

<https://www.certleader.com/ISTQB-CTFL-dumps.html>