DUMPSBOSS.

ISTQB Certified Tester Foundation Level - Automotive Software Tester

iSQI CTFL-AuT

Version Demo

Total Demo Questions: 7

Total Premium Questions: 77

Buy Premium PDF

https://dumpsboss.co

support@dumpsboss.co

support@dumpsboss.co
dumpsboss.co

Topic Break Down

Topic	No. of Questions
Topic 1, Automotive Software Engineering	6
Topic 2, Automotive Testing	10
Topic 3, Automotive Test Techniques	4
Topic 4, Automotive Testing Environments and Tools	24
Topic 5, Standards and Regulations	33
Total	77

QUESTION NO: 1

Which statement regarding requirements-based testing is true?

A. Requirements-based testing can be combined with other test practices or testing techniques.

B. Requirements-based testing can solve issues like complete requirements

C. Only extensive and delated requirements can be covered completely by the tester.

D. Requirements-based testing is a testing technique

ANSWER: A

Explanation:

Requirements-based testing can be combined with other test practices or testing techniques. This is correct because requirements provide an essential basis for deriving test conditions, test cases, and traceability, but they do not limit the overall test strategy to one technique. A test team can use requirements-based testing to demonstrate coverage of specified functional and non-functional needs, while also applying complementary practices that address additional risks and quality characteristics. For example, exploratory testing can help investigate behavior not fully anticipated in documented requirements, and performance, security, or robustness testing can provide evidence about quality attributes that need specialized test design. Combining techniques supports a more complete and risk-focused assessment of the item under test while preserving the traceability needed for requirements coverage. This aligns with the ISTQB approach that test analysis and design select suitable techniques according to the test basis, product risks, and test objectives. The Automotive Software Tester certification also emphasizes applying testing practices appropriately in the automotive context, where requirements traceability and risk-based evidence are especially important. See the ISTQB Automotive Software Tester certification overview and the ISTQB Certified Tester Foundation Level resources.

QUESTION NO: 2

Which of the statements regarding traceability in Automotive SPICES® 3.x is true?

A. The linking of requirements, test cases and test togs provides traceability and enables the desired verification of test cases for test case coverage

B. A1:N relation exists; for example If a test case is tested with the help of several requirements and is therefore linked to several requirements

C. The linking of a requirement, a derived test case and the associated test implementation provides the base for a complete testability.

D. With complete traceability, it is possible to trace test cases to their requirements and test cases which have been executed to their corresponding results.

ANSWER: D

Explanation:

“With complete traceability, it is possible to trace test cases to their requirements and test cases which have been executed to their corresponding results.” is correct because Automotive SPICE expects consistent, bidirectional relationships between work products across the engineering and test lifecycle. In testing, this means that a test case can be related back to the requirement or other basis it verifies, and that execution evidence can be related to the test case that produced it. The resulting trace chain provides objective evidence that requirements have been addressed by testing and that recorded test results belong to identifiable test executions.

This traceability is especially important for automotive development because it supports impact analysis when requirements change, demonstrates coverage of the agreed test basis, and enables efficient investigation of observed results or defects. A complete relationship from requirement to test case and onward to execution result makes the test process assessable and auditable. Automotive SPICE process-assessment guidance treats traceability as an essential characteristic of controlled engineering and verification work products. See the VDA QMC Automotive SPICE overview and the ISTQB Certified Automotive Software Tester certification page for the relevant Automotive SPICE and automotive-testing context.

QUESTION NO: 3

As a unit tester (.module tester'') you are working on a project and are supposed to develop a test strategy.

Which of the following test and analysis procedures is. according to Automotive SPIC® NOT an appropriate clamant of a verification strategy for unit tests (software units), which ware "manually coded"?

Note The components to verify are non-safety relevant and safely relevant (up to ASIL-C) components

A. Tool supported static analysis

B. Hardware-in-the-Loop-Tests

C. Unit tests (Test of Software-Units)

D. Code-Reviews

ANSWER: B

Explanation:

Hardware-in-the-Loop-Tests is the procedure that is not an appropriate element of a verification strategy for manually coded software units. Automotive SPICE software-unit verification focuses on demonstrating that an individual unit meets its detailed design and specified requirements. This is normally achieved through unit-level dynamic testing, reviews, and tool-supported static analysis. These techniques can examine the unit's logic, interfaces, control flow, data flow, coding-rule compliance, and robustness while faults can still be localized directly to the unit under test.

Hardware-in-the-loop testing instead executes software in a representative hardware and plant-simulation environment. It is valuable at later verification levels, where the objective is to assess integration behavior, timing, hardware interaction, and system-level behavior under simulated operating conditions. That scope is beyond verification of an isolated manually coded software unit. The fact that units may support safety-related components up to ASIL C increases the need for a suitably rigorous unit-verification strategy, but it does not turn HIL testing into a unit-test technique. Automotive SPICE process-assessment information is published by VDA QMC Automotive SPICE; the automotive tester certification context is available from ISTQB Automotive Software Tester.

QUESTION NO: 4

Which statement regarding the coding standard MISRA-C:2012 Is true?

A. MISRA-C defines rules and guidelines. Rules are verifiable by static analysis tools and are always required.

B. MISRA-C defines rules and guidelines that are supposed to help avoid anomalies In object-oriented developed software (e.g. in C++)

C. MISRA-C defines rules that include among others that the source code should not include nested comments.

D. A typical rule in MISRA-C is that the developer has documented the implemented behavior wen.

ANSWER: C

Explanation:

MISRA-C:2012 includes a specific requirement concerning comment delimiters: the character sequences `/*` and `//` must not be used within a comment. This prevents nested or misleading comment constructs that can cause a compiler or analysis tool to interpret the remainder of source text differently than intended. For example, placing `/*` inside an existing block comment can make the comment structure difficult to understand and potentially conceal code or introduce portability issues. The statement that the source code should not include nested comments correctly reflects this MISRA-C requirement, although the formal wording is expressed in terms of the prohibited delimiter sequences rather than simply using the term "nested comments." MISRA-C is a C-language coding guideline set intended to support the development of safer, more reliable, and more secure embedded software, particularly in safety-related domains such as automotive systems. Its guidance covers language usage patterns that reduce ambiguity and make code more consistently analysable.

The MISRA organization provides an overview of the MISRA C standard at MISRA C; the standard is also available through the MISRA C:2012 product page.

QUESTION NO: 5

In Automotive SPICIE. which capability level in software Integration testing is characterized by the following statement: ,,Define the objectives for the execution of the process."

A. Capability level 2

B. Capability level 0

C. Capability level 1

D. Capability level 3

ANSWER: A

Explanation:

Capability level 2 is correct because this level assesses whether a process is managed. In Automotive SPICE, process management includes establishing objectives for performing the process, planning how it will be performed, assigning responsibilities, monitoring performance against the plan, and taking action where needed. Therefore, defining objectives for the execution of software integration testing is an indicator of managed process performance rather than merely carrying out the process.

For software integration testing, these objectives provide the basis for controlled execution: for example, defining expected integration-test progress, scope, resources, schedules, responsibilities, and relevant quality or completion criteria. The objectives allow the organization to manage the testing work and assess whether the intended process performance is achieved. This corresponds to the performance-management process attribute associated with capability level 2 in the Automotive SPICE assessment model.

The official Automotive SPICE materials and current model downloads are available from the VDA QMC Automotive SPICE downloads page. Additional information about Automotive SPICE assessments and the model framework is provided by the International Assessor Certification Scheme (intacs).

QUESTION NO: 6

Which statement regarding fault injection in an automotive test environment is true?

A. It introduces controlled abnormal conditions to evaluate the reaction of the test object.

B. It is used only after release to reproduce field failures in customer vehicles.

C. It replaces the need for normal functional testing.

D. It can be applied only in a MIL test environment.

ANSWER: A

Explanation:

Fault injection is the deliberate introduction of faults, failures, invalid inputs, or abnormal operating conditions in order to evaluate the behaviour and robustness of the test object. In a HiL environment, for example, an electrical fault simulation can be used to simulate a short circuit, an open circuit, an implausible sensor signal, or a supply-voltage disturbance under controlled conditions.

The purpose is not to damage the real control unit. The injected condition is defined and controlled so that the tester can assess fault detection, diagnostic reactions, fallback behaviour and recovery according to the relevant requirements.

QUESTION NO: 7

Which statement regarding AUTOSAR Is NOT TRUE?

A. One of the objectives of AUTOSAR is to support the development of reliable systems.

B. One of the objectives of AUTOSAR Is the replicability of software components between different vehicle platforms and between different partners.

C. One of the objectives of AUTOSAR is cooperation regarding the standards and the implementation.

D. One of the objectives of AUTOSAR is the definition of a maintainable as well as adjustable open architecture.

E. One of the objectives of AUTOSAR is to eliminate all OEM- and supplier-specific implementation decisions.

ANSWER: E

Explanation:

“One of the objectives of AUTOSAR is to eliminate all OEM- and supplier-specific implementation decisions.” is the statement that is not true. AUTOSAR is a collaborative automotive software standard that defines common architectures, methodologies, interfaces, and exchange formats to improve interoperability, reuse, portability, quality, and development efficiency. It does not require every vehicle manufacturer and supplier to make identical implementation decisions. Instead, AUTOSAR deliberately supports configurable software architectures and standardized interfaces, allowing an OEM or supplier to select, configure, and integrate functionality appropriate to a particular vehicle platform, electronic control unit, safety goal, and product strategy. This flexibility is essential because automotive systems differ substantially in hardware resources, vehicle functions, market requirements, and supplier ecosystems. Standardization therefore provides a common technical basis for cooperation and component integration while preserving necessary product-specific variation. The AUTOSAR partnership describes its purpose as developing and establishing open automotive software architecture standards, rather than prescribing one universal implementation for all participants. Its platform documentation also emphasizes standardized mechanisms and interfaces that can be configured for individual system needs. See the AUTOSAR overview and the AUTOSAR Classic Platform information.