98-379: Software Testing Fundamentals

Audience Profile

Candidates for this exam are seeking to prove core testing skills. Before taking this exam, candidates should have solid foundational knowledge of the topics outlined in this Exam Details page. It is recommended that candidates be familiar with the concepts of and have some hands-on experience with the technologies or practices described here either by taking relevant training courses or by working with tutorials and samples available on MSDN and in Microsoft Visual Studio.

Objective Domain

1. Describe Testing Fundamentals

1.1. Describe software testing.

This objective may include but is not limited to: measuring software quality and testing benefits

1.2. Describe software and hardware components.

This objective may include but is not limited to: distinguishing operating systems, network, data, and hardware and software and their interaction and dependencies.

1.3. Describe fundamentals of programming.

This objective may include but is not limited to: data types, distinguishing programming languages such as compiled or interpreted, and analyzing simple algorithms

1.4. Describe application lifecycle management.

This objective may include but is not limited to: agile, waterfall, spiral, product and project lifecycles

2. Describe Testing Methodology

2.1. Describe testing techniques.

This objective may include but is not limited to: manual testing, automated testing, distinguish black box and white box testing

2.2. Describe testing levels.

This objective may include but is not limited to: unit, component, and integration testing

2.3. Describe testing types.

This objective may include but is not limited to: functional, performance structural, regression, security, stress, accessibility, usability, and localization testing

3. Create Software Tests

3.1. Describe user-centric testing.

This objective may include but is not limited to: business need and issues, customer requirements, and scenarios

3.2. Describe software testability.

This objective may include but is not limited to: test-driven development and testing hooks

3.3. Create test plan components.

This objective may include but is not limited to: test schedule, scope, methodology, scenarios, and tools

3.4. Describe feature tests.

This objective may include but is not limited to: distinguishing the functionality in the appropriate feature test

3.5. Define appropriately scoped test cases.

This objective may include but is not limited to: boundary conditions, level of detail, and validity

4. Manage Software Testing Projects

4.1. Describe testing milestones.

This objective may include but is not limited to: process fundamentals, exit criteria, and sign off

4.2. Describe the agile process.

This objective may include but is not limited to: scrum, kanban, and sprint management

4.3. Work with distributed teams.

This objective may include but is not limited to: communication, risk management, schedule management, and delivery process

4.4. Define test reports.

This objective may include but is not limited to: define appropriate status and project report components, define reporting cadence to meet project milestones, identify appropriate recipients for various report types

5. Work with Bugs

5.1. Detect software defects.

This objective may include but is not limited to: executing test cases, running automation scripts

5.2. Log bugs.

This objective may include but is not limited to: priority, severity, dependency, and repro steps

5.3. Manage bugs.

This objective may include but is not limited to: triage, resolution, closing, monitoring, and bug summary reports

6. Automate Software Testing

6.1. Describe test automation.

This objective may include but is not limited to: benefits, candidates for automation, and automation process

6.2. Define test automation strategies.

This objective may include but is not limited to: code coverage, logging, and automation priority

6.3. Write automation tests.

This objective may include but is not limited to: logic, error handling, commenting, and virtual machines

6.4. Manage test scripts.

This objective may include but is not limited to: smoke test, build verification test, and lab management