Regents competency test in writing p 12 File PDF

Regents Competency Test in Writing P 12

The Regents Competency Test in Writing P 12 is a critical assessment designed for high school students in New York State to demonstrate their proficiency in writing skills. This exam serves as a key component for graduation requirements, ensuring students possess the essential writing abilities necessary for college, career, and civic responsibilities. Understanding the structure, expectations, and preparation strategies for the P 12 segment of the Regents exam can significantly enhance a student's performance and confidence.

Understanding the Regents Competency Test in Writing P 12

The Regents Competency Test (RCT) in Writing is divided into several parts, with P 12 being a specific section that assesses students? ability to produce clear, coherent, and well-organized writing. This section emphasizes the importance of effective communication, grammar, and composition skills.

What is P 12 in the RCT Writing?

P 12 refers to a particular prompt or set of prompts within the RCT that requires students to write an essay or a constructed response based on a given topic. It often involves:

- Analyzing a prompt or question
- Planning a response
- Writing a structured essay or paragraph
- Demonstrating proper grammar, punctuation, and spelling

This part of the test aims to evaluate students' ability to develop ideas logically, support their opinions with evidence, and communicate their thoughts clearly.

Structure and Format of the P 12 Section

Understanding the format of the P 12 section is essential for effective preparation. Typically, the section includes:

- Prompt or Question: A statement or question that guides the writing task
- Writing Time: Usually around 45-60 minutes
- Expected Response: An essay or extended paragraph of approximately 250-300 words

The prompts may cover various topics, including personal experiences, social issues, or literary analysis. Students are expected to craft a coherent response that addresses all parts of the prompt.

Key Components of a Strong Response

To excel in the P 12 section, students should focus on:

- Thesis Statement: Clearly stating the main idea or argument
- Organization: Using clear paragraphs with topic sentences
- Development: Supporting ideas with examples, details, and evidence
- Conclusion: Summarizing main points and reinforcing the thesis
- Language Use: Employing proper grammar, punctuation, and vocabulary

Preparation Strategies for the Regents P 12 in Writing

Effective preparation involves a combination of skills practice, familiarity with the test format, and developing good writing habits. Here are some strategies to help students succeed:

1. Practice Writing Regularly

- Write essays or responses to various prompts
- Focus on developing a clear thesis and supporting details
- Time yourself to simulate test conditions

2. Review Grammar and Style

- Refresh knowledge of sentence structure, punctuation, and usage
- Practice editing and revising drafts for clarity and correctness

3. Develop Planning Skills

- Use graphic organizers (e.g., mind maps, outlines) to plan responses
- Identify main ideas and supporting details before writing

4. Study Sample Prompts and Responses

- Analyze high-scoring essays to understand what makes them effective
- Practice responding to similar prompts

5. Seek Feedback and Revise

- Share practice essays with teachers or peers for constructive criticism
- Revise responses to improve coherence and accuracy

Scoring Criteria and Expectations

The P 12 section is scored based on specific criteria that evaluate the quality of the writing. These include:

- Content and Development: Addressing all parts of the prompt, providing relevant details
- Organization: Logical structure with paragraphing and transitions
- Language Use: Correct grammar, punctuation, spelling, and vocabulary
- Conventions: Proper sentence structure and adherence to standard writing conventions

High-scoring responses effectively communicate ideas, demonstrate critical thinking, and showcase language proficiency.

Common Challenges and How to Overcome Them

Many students encounter hurdles when preparing for the P 12 section. Some common challenges include:

- Time Management: Struggling to complete responses within the allotted time
- Idea Development: Having difficulty elaborating ideas or providing sufficient support
- Language Errors: Making frequent grammatical or spelling mistakes

To overcome these issues:

- Practice timed writing exercises
- Focus on developing a strong plan before writing

- Review common grammar rules and vocabulary words regularly

Resources and Support for P 12 Preparation

Students can utilize various resources to enhance their skills:

- School Writing Labs: Participate in workshops or tutoring sessions
- Online Practice Tests: Use practice exams available on educational websites
- Guides and Rubrics: Study scoring rubrics to understand expectations
- Sample Essays: Review model responses to identify characteristics of high-quality writing

Conclusion

The Regents Competency Test in Writing P 12 plays a vital role in assessing students? writing capabilities and readiness for post-secondary opportunities. Success requires understanding the test format, practicing consistently, and honing essential writing skills. By focusing on organization, clarity, grammar, and content development, students can improve their performance and confidently approach the exam. Adequate preparation, combined with strategic study habits, will position students to achieve their best possible scores and meet graduation requirements with assurance.

Regents Competency Test in Writing P-12: A Comprehensive Review

The Regents Competency Test (RCT) in Writing P-12 is a vital assessment designed to evaluate the writing skills of students within the P-12 education system. It serves as a key component for high school graduation requirements in New York State, ensuring that students possess the essential writing competencies necessary for academic success and future endeavors. This detailed review explores every critical aspect of the RCT in Writing P-12, from its purpose and structure to preparation strategies, scoring criteria, and tips for success.

Understanding the Purpose and Significance of the RCT in Writing P-12

The Regents Competency Test in Writing P-12 is more than just an exam; it is a measure of a student's ability to communicate effectively through writing. Its primary objectives include:

- Ensuring Academic Readiness: Confirming that students can write clearly, coherently, and correctly across various contexts.
- Graduation Requirement: As part of the Regents diploma requirements, passing this test is essential for students aiming to graduate.

- Skill Development: Encouraging students to develop essential writing skills that will benefit them in college, careers, and daily life.

The test's focus on writing competence aligns with the broader goals of the P-12 education system, which emphasizes critical thinking, clarity, organization, and correctness.

Test Structure and Content Overview

Understanding the structure of the RCT in Writing P-12 is crucial for effective preparation. The test typically consists of two main parts:

Part 1: Multiple-Choice Questions

- Number of Questions: Usually around 40-50.
- Content Focus:
- Grammar and usage
- Sentence structure
- Punctuation
- Word choice
- Coherence and organization
- Mechanics of writing
- Purpose: To assess students' understanding of fundamental language conventions and their ability to recognize correct writing practices.

Part 2: Writing Task

- Type of Prompt: Students are presented with a specific scenario or topic.
- Task Requirements:
- Write an organized essay, letter, or response that addresses the prompt.
- Demonstrate clarity of ideas, coherence, and proper mechanics.
- Typically, students are asked to develop a main idea, support it with details, and conclude effectively.
- Word Limit: Usually around 150?200 words, but students should confirm current requirements as they can vary.

Key Skills Assessed in the RCT in Writing P-12

The test evaluates a broad range of writing skills, including:

- Organization and Structure
- Clear introduction, body, and conclusion
- Logical progression of ideas
- Content Development
- Relevance to the prompt
- Use of supporting details and examples
- Language Use
- Appropriate vocabulary
- Sentence variety
- Mechanics
- Grammar

- Subject-verb agreement
- Correct tense usage
- Proper punctuation
- Capitalization
- Spelling

- Conventions
- Paragraphing
- Transition words and phrases
- Correct formatting

Preparation Strategies for Success

Adequate preparation is essential for passing the RCT in Writing P-12. Here are comprehensive strategies:

1. Familiarize Yourself with the Test Format

- Review sample questions and prompts.
- Practice both multiple-choice and writing tasks.
- Understand time limits; plan your pace accordingly.

2. Strengthen Grammar and Usage Skills

- Use grammar workbooks and online resources.

- Focus on common errors such as sentence fragments, run-ons, and punctuation mistakes.
- Practice identifying and correcting errors in sample sentences.

3. Develop Effective Writing Habits

- Practice brainstorming ideas quickly.
- Create outlines before writing to organize thoughts.
- Work on writing clear and focused paragraphs.
- Review and revise your work for clarity and correctness.

4. Expand Vocabulary

- Study synonyms and context clues.
- Incorporate varied vocabulary into writing practice.

5. Practice Timed Writing

- Simulate test conditions.
- Allocate specific amounts of time for planning, writing, and revising.
- Aim for a balanced approach to complete the task confidently within the time limit.

6. Seek Feedback and Use Resources

- Have teachers or peers review practice essays.
- Utilize online resources, writing centers, and study guides.
- Attend prep workshops if available.

Scoring Criteria and What Examiners Look For

The RCT in Writing P-12 employs a holistic scoring approach, emphasizing the overall effectiveness of the student?s writing. Factors include:

- Content and Development: Does the response address all parts of the prompt? Are ideas well-developed with relevant details?
- Organization and Structure: Is the writing logically organized, with clear transitions?
- Language and Style: Is the vocabulary appropriate? Are sentences varied and clear?

- Mechanics and Conventions: Are grammar, punctuation, spelling, and capitalization correct?
- Adherence to Instructions: Does the writing meet the word count and format requirements?

Scores typically range from 1 to 4, with 3 or higher generally required to pass. The scoring rubric provides detailed descriptors for each score level, guiding students on the expectations.

Common Challenges and How to Overcome Them

Students often face specific hurdles when taking the RCT in Writing P-12. Addressing these proactively can improve outcomes:

- Time Management: Practice writing within set time limits to ensure completion.
- Understanding the Prompt: Carefully read and analyze the prompt to ensure your response is relevant.
- Organizational Skills: Use outlines to structure ideas before writing.
- Mechanical Errors: Regularly review grammar rules and proofread your work.
- Confidence Issues: Practice writing regularly to build confidence and reduce anxiety.

Test Day Tips and Final Recommendations

On the day of the test, keep these tips in mind:

- Arrive Prepared: Bring necessary materials such as pencils, erasers, and any allowed reference materials.
- Read Instructions Carefully: Ensure understanding of each section?s requirements.
- Plan Your Time: Allocate time for planning, writing, and reviewing.
- Stay Calm: Take deep breaths and approach each part methodically.
- Review Your Work: Leave time at the end to proofread for errors and clarity.

Post-Test Considerations and Next Steps

- Results and Retakes: If you do not pass on the first attempt, review your score report to identify areas for improvement. Many students retake the test after targeted preparation.
- Utilize Feedback: Use feedback from practice essays and scoring guides to refine your writing skills.
- Continuous Practice: Regular writing exercises, reading diverse materials, and seeking feedback can help boost performance in future attempts.

Conclusion

The Regents Competency Test in Writing P-12 is a comprehensive assessment that plays a crucial role in confirming students' readiness for graduation and their ability to communicate effectively. Success on this test requires a solid understanding of writing mechanics, organization, content development, and test strategies. With diligent preparation, focused practice, and a clear understanding of the test structure and scoring criteria, students can confidently approach the RCT in Writing P-12 and achieve their academic goals. Remember, consistent effort and constructive feedback are your best tools for mastering the skills needed to excel in this important examination.

QuestionAnswer What is the purpose of the Regents Competency Test in Writing P 12? The Regents Competency Test in Writing P 12 assesses students' ability to write clear, coherent, and well-structured essays as a requirement for graduation and to demonstrate proficiency in writing skills. What are the main components of the Regents Competency Test in Writing P 12? The test typically includes multiple-choice questions on grammar and usage, as well as a writing task where students compose an essay or short response based on a given prompt. How can students prepare effectively for the Regents Competency Test in Writing P 12? Students should review grammar rules, practice writing essays on various topics, familiarize themselves with the test format, and seek feedback on their writing to improve coherence, grammar, and organization. What is the passing score for the Regents Competency Test in Writing P 12? The passing score may vary by year, but generally, students need to achieve a minimum score set by the New York State Education Department, often around 75 or higher, to meet graduation requirements. Are there any accommodations available for students with special needs taking the Regents Competency Test in Writing P 12? Yes, students with documented disabilities can request accommodations such as extended time, separate testing rooms, or use of assistive technology in accordance with their Individualized Education Program (IEP) or 504 plan.

Related keywords: regents competency test, writing, P12, NYS regents, standardized test, essay writing, writing skills, test preparation, exam requirements, academic assessment

Microsoft test manager tutorial

Microsoft Test Manager Tutorial: A Comprehensive Guide to Efficient Test Management

In the world of software development, ensuring the quality and reliability of your applications is paramount. Microsoft Test Manager (MTM) is a powerful tool designed to streamline the testing process, enabling teams to plan, execute, and track testing efforts with ease. Whether you're a beginner or looking to enhance your testing workflows, this **Microsoft Test Manager tutorial** will

provide you with a step-by-step guide to harnessing MTM's full potential for your projects.

Understanding Microsoft Test Manager

Before diving into the tutorial, it's essential to grasp what Microsoft Test Manager is and how it integrates into the broader Visual Studio ecosystem.

What is Microsoft Test Manager?

Microsoft Test Manager is a test management tool integrated with Visual Studio Team Services (VSTS) and Azure DevOps. It provides an intuitive interface for managing test plans, creating and executing test cases, and tracking testing progress. MTM supports both manual and exploratory testing, making it versatile for various testing scenarios.

Key Features of Microsoft Test Manager

- Test Planning: Create and organize test plans, test suites, and test cases.
- Test Execution: Execute manual tests and record results efficiently.
- Test Tracking: Monitor testing progress and generate detailed reports.
- Exploratory Testing: Conduct ad-hoc testing sessions with session-based testing capabilities.
- Integration: Seamlessly connects with Azure DevOps for continuous testing workflows.

Setting Up Microsoft Test Manager

Getting started with MTM involves installing and configuring the tool for your environment.

Prerequisites

- Visual Studio Enterprise or Professional edition (with test management features).
- Azure DevOps account or TFS (Team Foundation Server) setup.
- Appropriate permissions to access test management features.

Installing Microsoft Test Manager

- Download the Visual Studio Installer from the official Microsoft website.
- Choose the edition that includes testing tools (e.g., Visual Studio Enterprise).
- During installation, select the "Testing Tools" workload to install MTM.
- Complete the installation and launch Visual Studio.

- Connect to your Azure DevOps project or TFS server within Visual Studio.

Creating and Managing Test Plans

A well-structured test plan is the foundation of successful testing. MTM simplifies organizing your testing efforts through test plans and suites.

Creating a Test Plan

- Open Microsoft Test Manager from Visual Studio or the Azure DevOps portal.
- Navigate to the "Test Plans" section.
- Click "New Test Plan" and provide a descriptive name, description, and start/end dates.
- Associate the test plan with a specific build or release if needed.
- Save the test plan to begin adding test cases.

Organizing Test Suites

Test suites help categorize test cases based on features, modules, or testing strategies.

- **Static Test Suites:** Manually organize test cases into logical groups.
- **Requirement-Based Test Suites:** Link test cases to specific requirements or user stories.
- **Query-Based Test Suites:** Dynamically generate test cases based on queries.

Adding Test Cases

- Within your test suite, click "New Test Case" or import existing cases.
- Define the test case steps, expected results, and associated requirements.
- Assign priority, owner, and other metadata to facilitate management.
- Save the test case to include it in your suite.

Executing Tests with Microsoft Test Manager

Once your test cases are organized, the next step is executing tests and recording results.

Manual Test Execution

- Open the test plan and select the test suite to execute.

- Click "Run" on the test case you wish to execute.
- Follow the test steps as documented, marking each step as "Passed" or "Failed."
- If a test fails, record detailed bug information directly from the test runner.
- Complete the test run, and the results will be saved automatically.

Using Action Recording and Data Collection

MTM allows capturing detailed logs and screenshots during test execution, aiding in defect analysis and reproducibility.

Exploratory Testing

For ad-hoc testing, MTM offers session-based testing:

- Schedule a testing session with session details.
- Conduct testing while documenting observations and findings.
- Record bugs and issues discovered during the session.

Tracking and Reporting Testing Progress

Effective test management requires insight into testing status and quality metrics.

Monitoring Test Results

- Navigate to the "Test Results" section in MTM or Azure DevOps.
- Review individual test case outcomes and overall progress.
- Filter results by tester, status, or test plan for detailed analysis.

Generating Reports

MTM provides various built-in reports to visualize testing status:

- Test Results Summary
- Test Case Progress and Trends
- Bugs and Defects Reports

Integrating with Bug Tracking

During testing, bugs can be logged directly from MTM, linked to specific failed test cases, ensuring traceability and quick resolution.

Advanced Tips and Best Practices

Maximize your testing efficiency with these expert tips:

Automate Repetitive Tests

Integrate MTM with automated testing frameworks like MSTest, NUnit, or Selenium to run automated tests alongside manual efforts.

Maintain Test Cases Regularly

Keep your test cases updated with application changes to ensure relevance and accuracy.

Use Tags and Filters

Leverage tags and filters to quickly locate and execute specific test cases based on criteria like priority, feature, or owner.

Collaborate Effectively

Encourage team collaboration by sharing test plans and results, and conducting regular review meetings to discuss testing progress.

Conclusion

Microsoft Test Manager is an essential tool for teams aiming to implement structured and efficient testing workflows. From creating comprehensive test plans and organizing test suites to executing tests and tracking outcomes, MTM provides a centralized platform that enhances test visibility and quality assurance. By following this **Microsoft Test Manager tutorial**, you can streamline your testing process, improve defect detection, and deliver higher-quality software products. Remember to stay updated with the latest features and best practices to maximize your testing success with MTM.

Microsoft Test Manager tutorial: A Comprehensive Guide to Streamlining Your Testing Process

Microsoft Test Manager (MTM) is a powerful tool integrated within the Azure DevOps suite, designed to facilitate efficient test planning, management, and execution. Whether you are a QA professional, a developer, or a project manager, mastering MTM can significantly enhance your

testing workflows, improve defect detection rates, and ensure higher quality software releases. This tutorial aims to provide an in-depth overview of Microsoft Test Manager, guiding you through its features, setup, best practices, and real-world applications.

Introduction to Microsoft Test Manager

Microsoft Test Manager is a dedicated testing management environment that helps teams plan, execute, and track testing activities seamlessly. Originally part of Visual Studio Test Professional, MTM is now integrated within Azure DevOps, offering a unified platform for Agile testing, exploratory testing, and manual test case management.

Key Features of Microsoft Test Manager

- Test Planning: Create and organize test plans, test suites, and test cases.
- Test Execution: Run manual tests, record results, and capture screenshots.
- Test Management: Track defects, manage test configurations, and generate reports.
- Integration: Seamlessly connect with Azure DevOps for continuous integration and automated testing workflows.
- Exploratory Testing: Record exploratory sessions with detailed notes and recordings.

Setting Up Microsoft Test Manager

Before diving into testing, proper setup of MTM is essential.

Prerequisites

- A valid Azure DevOps account with appropriate permissions.
- Installed Visual Studio Test Professional or access via Azure DevOps.
- Access to a test environment or lab machines.

Installation and Configuration

- Download and Install: Download Microsoft Test Manager from the Visual Studio downloads page or via Azure DevOps.
- Connect to Azure DevOps: Launch MTM and connect it to your Azure DevOps project by signing in with your credentials.
- Configure Test Environments: Set up test machines, configurations, and network environments to align with your testing needs.

Tips for Effective Setup

- Organize your test plans hierarchically for easier management.
- Maintain a clear mapping between test environments and real-world configurations.
- Regularly update test artifacts to ensure they reflect current application versions.

Creating and Managing Test Plans

Test planning is the foundation of effective testing.

Creating a Test Plan

- In MTM, navigate to the "Test Plans" section.
- Click "New Test Plan" and provide a descriptive name, start/end dates, and any relevant details.
- Assign ownership and stakeholders.

Organizing Test Suites

- Static Test Suites: Manually added test cases grouped logically.
- Requirement-based Suites: Linked to user stories or requirements for traceability.
- Query-based Suites: Dynamic grouping based on queries.

Best Practices

- Break down large test plans into smaller, manageable suites.
- Link test cases to user stories or bugs for better traceability.
- Regularly review and update test plans to reflect current project scope.

Designing and Managing Test Cases

Creating effective test cases is critical to uncovering defects early.

Writing Test Cases

- Clear, concise steps with expected results.
- Use descriptive titles and tags for easy filtering.

- Include preconditions and postconditions.

Reusing Test Cases

- Modularize test steps for reuse across different test cases.
- Attach relevant documentation, screenshots, or scripts.

Organizing Test Cases

- Group related test cases into shared test suites.
- Use labels and tags for categorization.

Tips for Effective Test Cases

- Prioritize test cases based on risk and impact.
- Keep test cases up-to-date with application changes.
- Incorporate exploratory testing notes within test cases or as separate sessions.

Test Execution and Result Recording

Executing tests efficiently and accurately recording results is vital.

Running Manual Tests

- Select a test case from your test suite.
- Follow each step, observing the actual behavior.
- Record outcomes: Passed, Failed, Blocked, or Not Applicable.
- Capture screenshots or record videos if needed.

Recording Defects

- Link defects directly to failed test cases.
- Provide detailed descriptions, steps to reproduce, and severity.
- Attach logs or screenshots to aid debugging.

Managing Test Runs

- Use test configurations to run multiple test cases under different environments.
- Schedule and assign test runs to team members.
- Monitor real-time progress and results.

Pros and Cons of Test Execution Features

Pros:

- Streamlined process for manual testing.
- Rich media capture for detailed documentation.
- Easy defect linkage and traceability.

Cons:

- Manual process can be time-consuming for large test suites.
- Limited automation capabilities within MTM itself.

Integrating Automated Testing with Microsoft Test Manager

While MTM excels at manual testing, integration with automated testing frameworks enhances overall efficiency.

Integration with Visual Studio Test Automation

- Use Visual Studio Test tools to develop automated test scripts.
- Link automated tests to test cases in MTM.
- Run automated tests as part of continuous integration pipelines.

Benefits of Integration

- Reduces manual effort.
- Ensures consistency between manual and automated testing.
- Facilitates comprehensive test coverage.

Limitations

- Setup can be complex for beginners.
- Requires scripting knowledge for automation.

Reporting and Tracking

Effective reporting provides insights into testing progress and quality metrics.

Built-in Reports

- Test Run Summary
- Test Results Trend
- Defect Status and Age
- Requirements Traceability Matrix

Custom Reports

- Use Azure DevOps Analytics or Power BI for tailored dashboards.
- Export data for external analysis.

Best Practices

- Regularly review reports to identify bottlenecks.
- Use metrics to prioritize testing efforts.
- Maintain up-to-date dashboards for stakeholder visibility.

Best Practices for Using Microsoft Test Manager

- Plan Early: Incorporate testing early in the development lifecycle.
- Maintain Test Artifacts: Keep test cases and plans updated with application changes.
- Leverage Linkages: Connect requirements, test cases, defects, and build versions.
- Automate Where Possible: Use automation to handle repetitive and regression testing.
- Collaborate: Engage team members through shared test plans and results.
- Review Regularly: Conduct test reviews and retrospectives for continuous improvement.

Common Challenges and How to Overcome Them

Challenge: Managing Large Test Suites

Solution: Break down into smaller, manageable suites; use query-based suites for dynamic grouping.

Challenge: Keeping Test Cases Up-to-date

Solution: Establish a review cycle aligned with development sprints; assign ownership.

Challenge: Integration Complexity

Solution: Invest in automation and continuous integration pipelines; use documentation and training.

Challenge: Limited Automation within MTM

Solution: Use external automation tools integrated with Azure DevOps, such as Selenium or Coded UI.

Conclusion

Microsoft Test Manager is a robust tool that, when used effectively, can significantly boost your testing efficiency, coverage, and traceability. Its comprehensive features for test planning, execution, defect management, and reporting make it indispensable for teams adopting Agile and DevOps practices. While there are challenges, especially around automation and managing large test repositories, with proper setup, training, and best practices, MTM can become a cornerstone of your software quality assurance process.

By following this tutorial, you should now have a clear understanding of how to leverage Microsoft Test Manager to streamline your testing operations, improve collaboration, and ultimately deliver higher quality software to your users. Remember, continuous learning and adaptation are key to maximizing the benefits of MTM in your projects.

QuestionAnswer What are the main features of Microsoft Test Manager (MTM)? Microsoft Test Manager (MTM) provides features such as test planning, manual test execution, test case management, bug tracking integration, and collaboration tools to streamline the testing process within Visual Studio and Azure DevOps. How do I create and organize test plans in Microsoft Test Manager? In MTM, you can create test plans by navigating to the 'Test Plans' hub, clicking 'New Test Plan,' and defining its name and details. You can then add test suites and test cases to organize tests based on requirements, features, or test types for better management. Can I automate tests using Microsoft Test Manager, and how is it integrated with other tools? While MTM primarily supports manual testing, it integrates seamlessly with Visual Studio and Azure DevOps, enabling automation through test scripts and frameworks like MSTest, Selenium, or CodedUI. Automated tests can be linked to test cases for comprehensive test management. What are the best

practices for using Microsoft Test Manager effectively? Best practices include maintaining detailed and reusable test cases, organizing tests into logical suites, utilizing shared steps for common actions, integrating with bug tracking, and regularly updating test plans based on project changes to ensure thorough testing coverage. Is Microsoft Test Manager suitable for Agile development environments? Yes, MTM is well-suited for Agile teams as it supports iterative testing, easy integration with Azure DevOps Agile tools, and flexible test planning, allowing teams to adapt testing activities to sprint cycles and continuous delivery workflows.

Related keywords: Microsoft Test Manager, TFS testing, test planning, test case management, automated testing, test execution, bug tracking, test lab management, test reporting, test automation tools

Read the full article online: MICROSOFT TEST MANAGER TUTORIAL