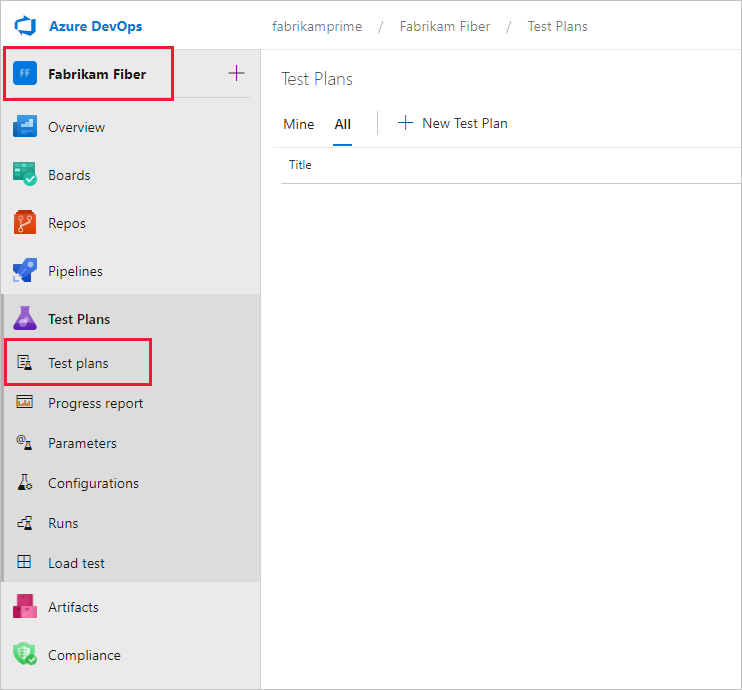
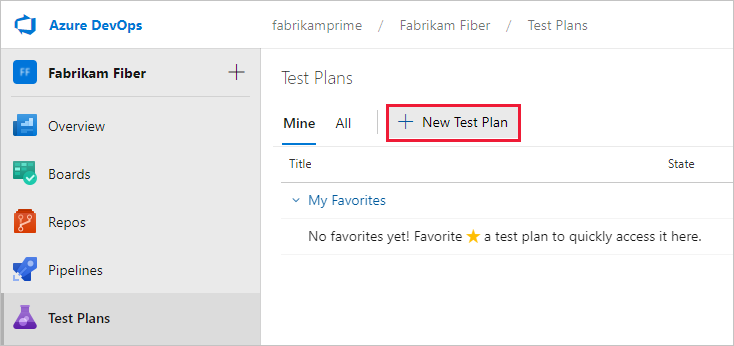
**Create a test plan**

In general, you create test plans to test requirements. Before creating a test plan, you may want to [define your backlog of requirements](https://learn.microsoft.com/en-us/azure/devops/boards/backlogs/create-your-backlog?view=azure-devops).

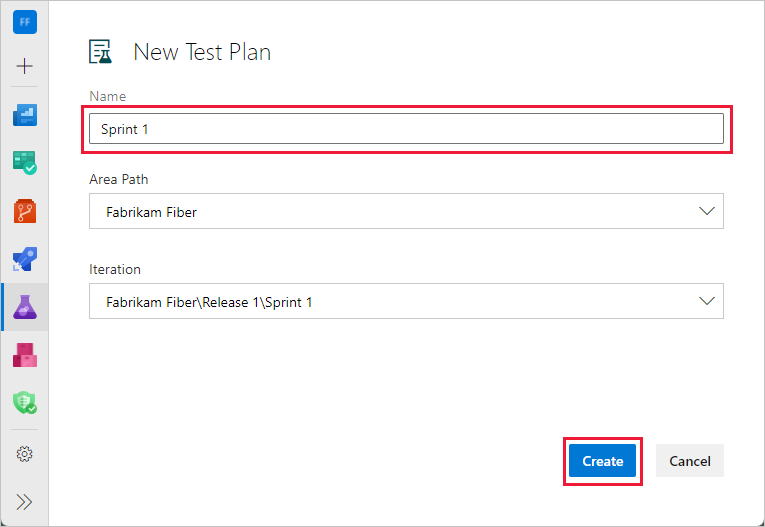
1. From the web portal, open your project and go to **Test Plans**. If you already have a test plan, select **Test Plans** to go to the page that lists all test plans.



1. In the **Test Plans** page, select **New Test Plan** to create a test plan for your current sprint.



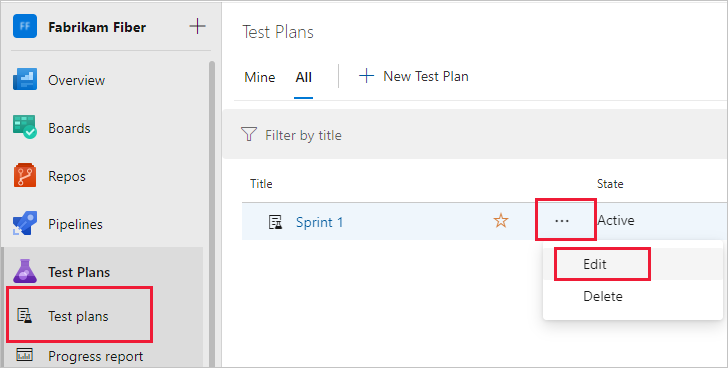
1. In **New Test Plan**, enter a name for the test plan. Verify that the area path and iteration are set correctly, then select **Create**.



**Rename a test plan**

To rename a test plan, use this procedure:

1. Select **Test Plans** to view test plans.
2. Next to the test plan name, select **More Actions** > **Edit**.



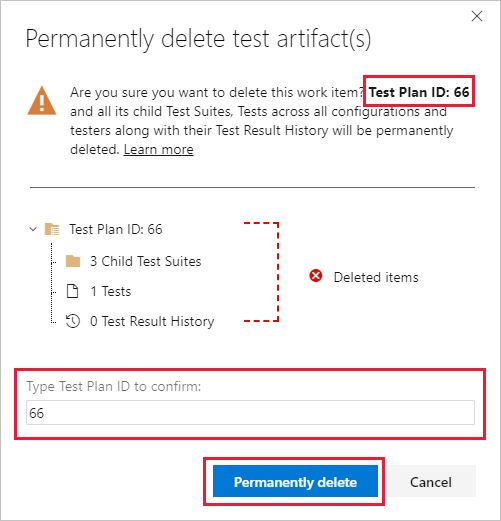
1. Change the name and then select **Save & Close**.

You can make other changes to the test plan here.

**Delete a test plan**

To delete a test plan, use this procedure:

1. Select **Test Plans** to view test plans.
2. Next to the test plan name, select **More Actions** > **Delete**.
3. The **Permanently delete test artifacts** dialog box explains exactly what will be deleted. Type the test plan ID to confirm that you want to delete, and then select **Permanently delete**.



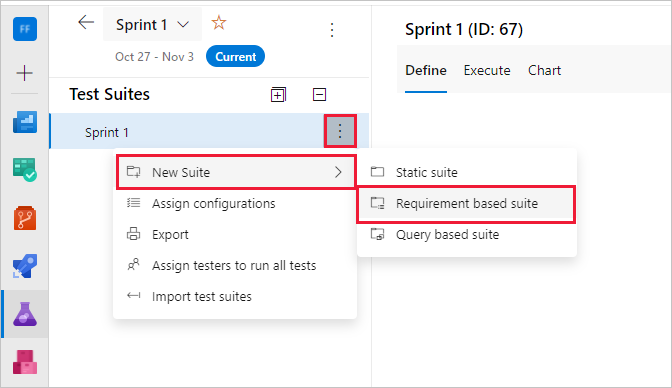
**Add a requirement-based test suite**

Now add test suites for the backlog items that need manual tests. These tests could be user stories, requirements, or other work items based your project.

**Note**

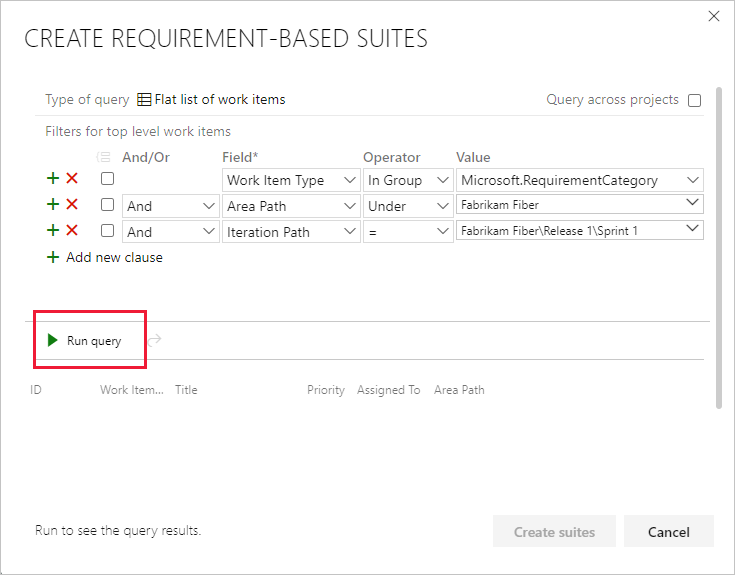
Requirement tracking is supported only for test cases linked through a **Requirement-based test suite**. Work items include a User Story ([**Agile**](https://learn.microsoft.com/en-us/azure/devops/boards/work-items/guidance/agile-process?view=azure-devops)), Product Backlog Item ([**Scrum**](https://learn.microsoft.com/en-us/azure/devops/boards/work-items/guidance/scrum-process?view=azure-devops)), Requirement ([**CMMI**](https://learn.microsoft.com/en-us/azure/devops/boards/work-items/guidance/cmmi-process?view=azure-devops)), and Issue ([**Basic**](https://learn.microsoft.com/en-us/azure/devops/boards/get-started/plan-track-work?view=azure-devops)). The association between a requirement work item and manual test execution is only formed when the test case is linked by using a **Requirement-based test suite**.

1. To add a suite to a test plan, select **More options** for the test suite, and then select **New Suite** > **Requirement based suite**.

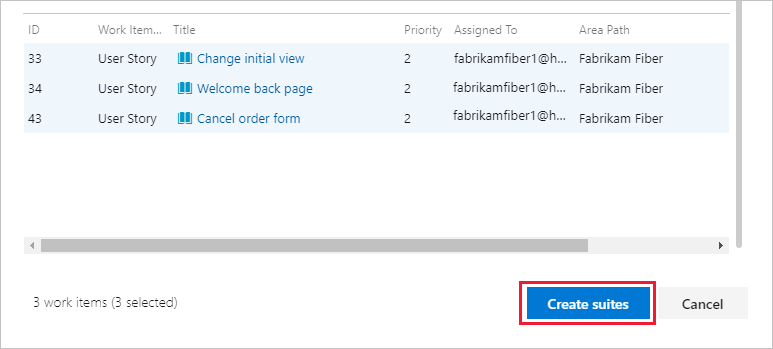


You use requirement-based suites to group your test cases together. That way, you can track the testing status of a backlog item. Each test case that you add to a requirement-based test suite is automatically linked to the backlog item.

1. In **Create requirement-based suites**, add one or more clauses to filter your work items by the iteration path for the sprint. Run the query to view the matching backlog items.

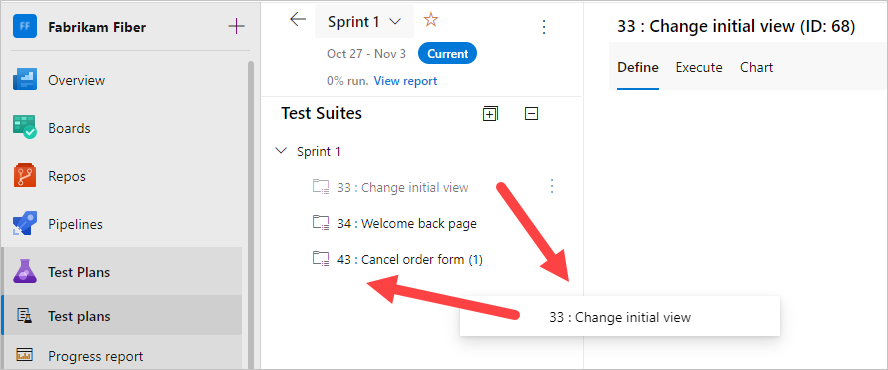


1. In the list of work items returned by the query, select the backlog items you want to test in this sprint. Select **Create suites** to create a requirement-based suite for each one.



**Work with test suites**

You can create a static test suite that can contain any type of test suites. Use these test suites like folders. Drag test suites to group them in a static test plan. Drag and drop tests to reorder them.

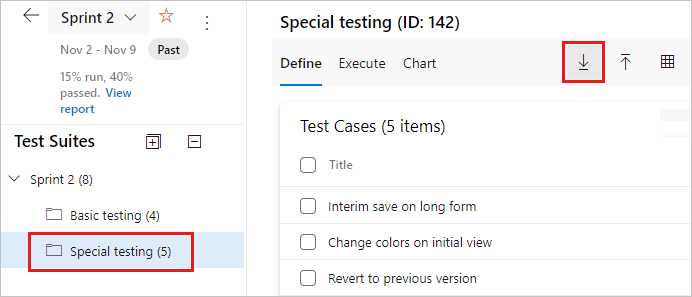


You can track changes to test plans and test suites. Open the work item for the test plan or test suite, then view the work item history.

For test suites, other actions are tracked in the **Test Suite Audit** field. For example, adding and removing test cases from a test suite are tracked in this field.

You can export test plans, test suites, and test cases.

Select **Export test cases to CSV**.



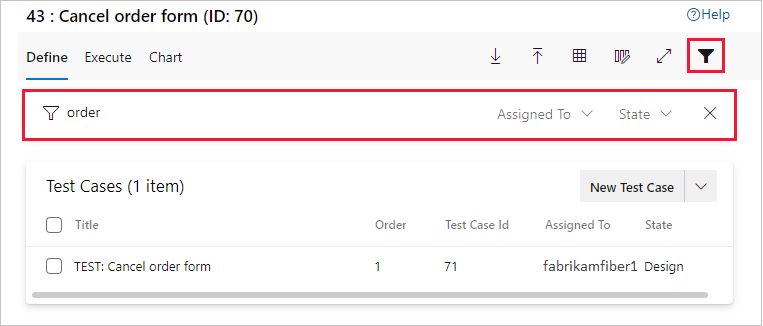
Change the test case fields in the report by adding or removing columns from the list view of the test suite.

**Important**

You cannot export more than 75 Test Suites in a single operation. The email supports up to 1MB of data.

**Find a test case in a test plan**

In **Test Plans** for your test plan, use the  filter icon to show the search and filter list. It can help find the tests you want.



**Next steps**

[**Create manual test cases**](https://learn.microsoft.com/en-us/azure/devops/test/create-test-cases?view=azure-devops#test-cases)

# Create manual test cases

* Article
* 02/25/2023
* 8 contributors

Feedback

**Azure DevOps Services | Azure DevOps Server 2022 - Azure DevOps Server 2019 | TFS 2018**

Create manual test cases to check that each deliverable meets your users' needs. Manual test cases define individual steps testers perform, including steps that are shared across test cases. To test different data, you specify parameters for the test steps. To learn more, see [Share steps between test cases](https://learn.microsoft.com/en-us/azure/devops/test/share-steps-between-test-cases?view=azure-devops) and [Repeat a test with different data](https://learn.microsoft.com/en-us/azure/devops/test/repeat-test-with-different-data?view=azure-devops). For an overview of test artifacts, see [Test objects and terms](https://learn.microsoft.com/en-us/azure/devops/test/test-objects-overview?view=azure-devops).

Organize your test cases by adding them to test plans and test suites. Then assign testers to run the tests.

For an overview of test artifacts, see [Test objects and terms](https://learn.microsoft.com/en-us/azure/devops/test/test-objects-overview?view=azure-devops).

**Note**

Test iterations are designed to support data-driven scenarios, not workflow-driven scenarios. From a best practice perspective, if you have two test scenarios where the workflows are different, consider creating separate test cases. Also see [**FAQs for manual testing**](https://learn.microsoft.com/en-us/azure/devops/test/reference-qa?view=azure-devops#testcases).

## Prerequisites

* You must connect to a project. If you don't have a project yet, [create one](https://learn.microsoft.com/en-us/azure/devops/user-guide/sign-up-invite-teammates?view=azure-devops).
* You must be added to a project. To get added, [Add users to a project or team](https://learn.microsoft.com/en-us/azure/devops/organizations/security/add-users-team-project?view=azure-devops).
* To view test-related work items, you must have **Basic** access or higher and permissions to view work items under the corresponding Area Path.
* To [add test plans and test suites](https://learn.microsoft.com/en-us/azure/devops/test/create-a-test-plan?view=azure-devops), [manual test cases](https://learn.microsoft.com/en-us/azure/devops/test/create-test-cases?view=azure-devops), [delete test artifacts](https://learn.microsoft.com/en-us/azure/devops/boards/backlogs/delete-test-artifacts?view=azure-devops), and [define test configurations](https://learn.microsoft.com/en-us/azure/devops/test/test-different-configurations?view=azure-devops) you must have [Basic + Test Plans](https://marketplace.visualstudio.com/items?itemName=ms.vss-testmanager-web) access level or have one of the following **Visual Studio subscriptions**:
  + [Enterprise](https://visualstudio.microsoft.com/vs/enterprise/)
  + [Test Professional](https://visualstudio.microsoft.com/vs/test-professional/)
  + [MSDN Platforms](https://visualstudio.microsoft.com/msdn-platforms/)
* To add or edit test-related artifacts, you must have the following permissions:
  + To add or modify test plans, test suites, test cases, or other test-based work item types, you must have **Edit work items in this node** permission set to **Allow** under the corresponding **Area Path**.
  + To modify test plan properties such as build and test settings, you must have **Manage test plans** permission set to **Allow** under the corresponding **Area Path**.
  + To create and delete test suites, add, and remove test cases from test suites, change test configurations associated with test suites, and modify a test suite hierarchy (move a test suite), you must have **Manage test suites** permission set to **Allow** under the corresponding **Area Path**.

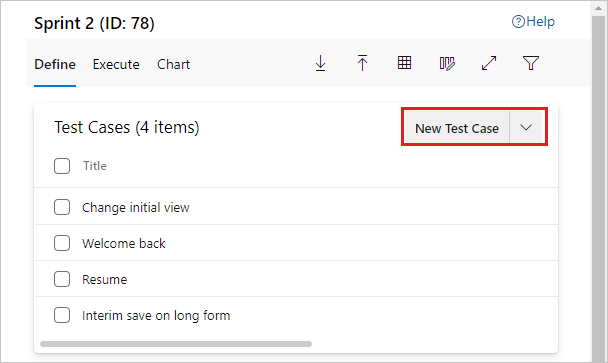
To learn more, see [Manual test access and permissions](https://learn.microsoft.com/en-us/azure/devops/test/manual-test-permissions?view=azure-devops).

**Note**

If you're new to using Azure Test Plans, review [**Navigate Test Plans**](https://learn.microsoft.com/en-us/azure/devops/test/navigate-test-plans?view=azure-devops) to understand how to use the user interface to access select functions.

## Create test cases

1. If you haven't already, [create a test plan and requirement-based test suites](https://learn.microsoft.com/en-us/azure/devops/test/create-a-test-plan?view=azure-devops).
2. Select a requirement-based test suite. To the right, select the **New Test Case** button.

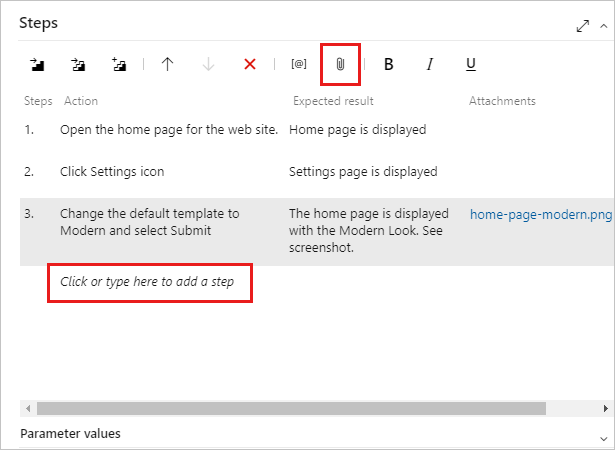


If you have no existing test cases, the button appears in the lower right.

**Note**

The [**test suite**](https://learn.microsoft.com/en-us/azure/devops/test/create-a-test-plan?view=azure-devops) shown here was created from a User Story work item in the team's backlog board. When you add a test case to this kind of suite, the test case is automatically linked to the backlog item. To create test cases this way, open the context menu for the work item and choose **Add test**.

1. In the new work item, enter a title. Select the **Click or type here to add a step** option.



Add test steps with a description of the action required to carry out the test and the expected results so that any team member can run the test. You can add attachments to a step if you want. Repeat until you have added all the steps for the test.

**Note**

You can share steps between test cases. For more information, see [**Share steps**](https://learn.microsoft.com/en-us/azure/devops/test/share-steps-between-test-cases?view=azure-devops).

Now you've created a test case that you can run.

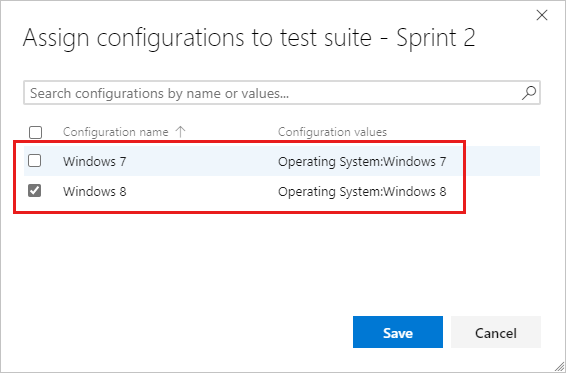
You can also copy test cases. For more information, see [Copy or clone stories, issues and other work items](https://learn.microsoft.com/en-us/azure/devops/boards/backlogs/copy-clone-work-items?view=azure-devops).

## Assign configurations to test cases

You can specify configurations, such as different operating systems, web browsers, and other variations for your tests.

To assign test configurations to all test cases in a test suite, select the test suite, select **More options** or right-click to open the context menu, and then select **Assign configurations**.

In the **Assign configurations to test suite** dialog box, select the configurations you want.



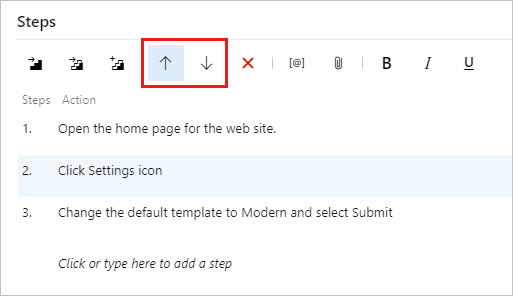
You can also assign configurations to individual test cases. Select one or more test cases, select **More options** or right-click to open the context menu, and then select **Assign configuration**.

The **Assign configurations to selected test cases** dialog box opens. Make and save your changes.

For more information about configurations, see [Test different configurations](https://learn.microsoft.com/en-us/azure/devops/test/test-different-configurations?view=azure-devops).

## Reorder test cases

You can reorder manual test cases in static suites, requirement-based suites, and query-based suites. Open a test case, then use the up and down arrows to change the order.

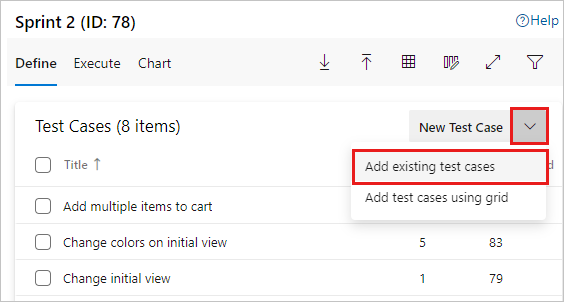


There are also options to change order in the step context menu.

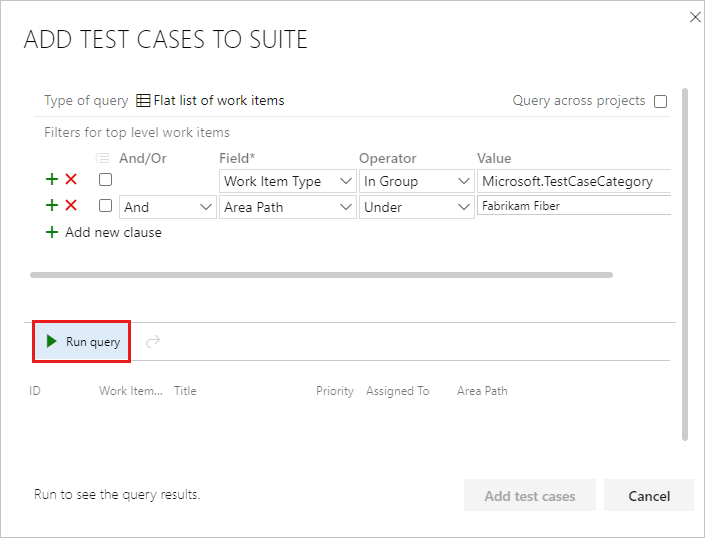
## Add existing test cases to a test suite

You can add existing test cases to a test suite.

1. Select a test suite. From the **New Test Case** menu, select **Add existing test case**.



1. In the **Add test cases to suite** dialog box, add search clauses, as needed, and then select **Run query**.



1. After you find the test cases you want, select **Add test cases**.

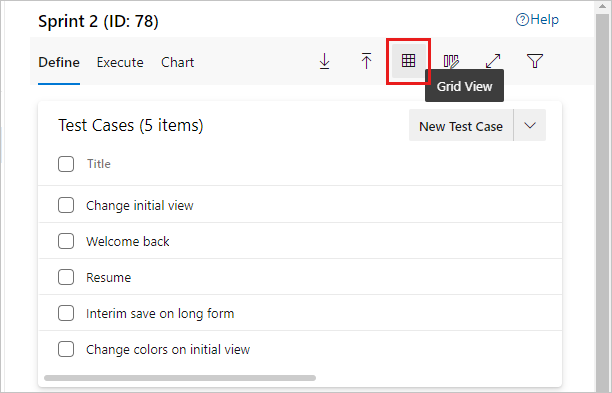
**Tip**

You can create a test case that automatically links to a requirement—User Story ([**Agile**](https://learn.microsoft.com/en-us/azure/devops/boards/work-items/guidance/agile-process?view=azure-devops)), Product Backlog Item ([**Scrum**](https://learn.microsoft.com/en-us/azure/devops/boards/work-items/guidance/scrum-process?view=azure-devops)), Requirement ([**CMMI**](https://learn.microsoft.com/en-us/azure/devops/boards/work-items/guidance/cmmi-process?view=azure-devops)), or Issue ([**Basic**](https://learn.microsoft.com/en-us/azure/devops/boards/get-started/plan-track-work?view=azure-devops))—when you create a test from the Kanban board. To learn more, see [**Add, run, and update inline tests**](https://learn.microsoft.com/en-us/azure/devops/boards/boards/add-run-update-tests?view=azure-devops).

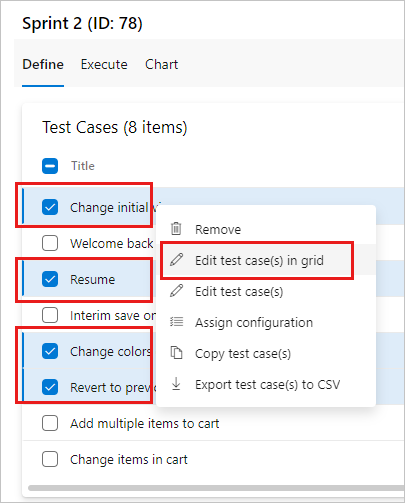
## Use the Grid view to edit test cases

You can copy and paste test cases into the **Grid** view.

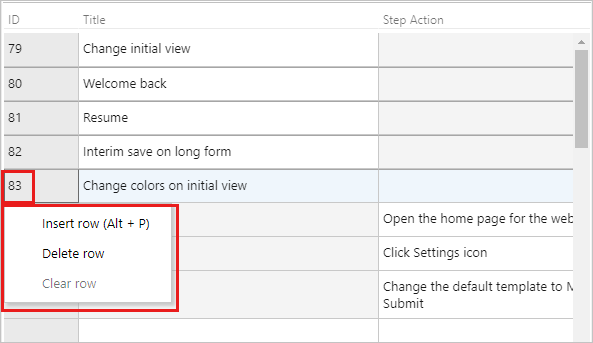
To open the **Grid** view, select the **Grid View** icon at the right.



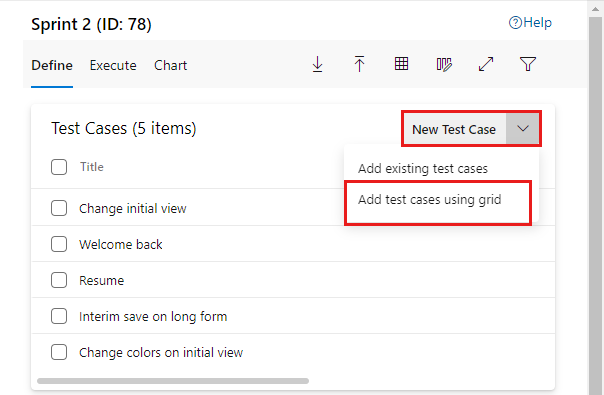
Instead, you can select several test cases and open them for editing in the **Grid** view. In the context menu, select **Edit test case(s) in grid**.



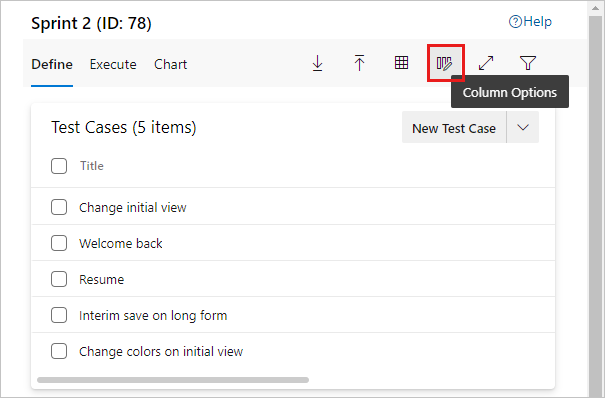
On the **Grid** context menu, you can add, delete, or clear rows.



You can use the **Grid** view when you add multiple test cases to the test suite:



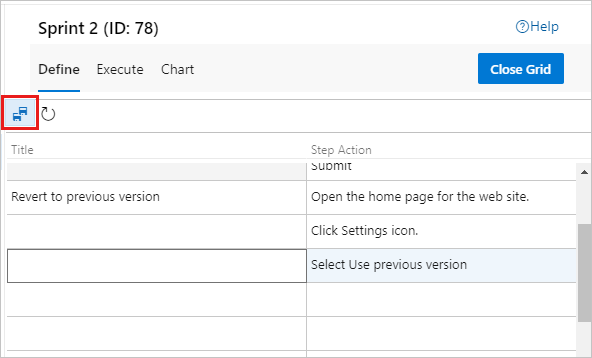
You can edit other fields in the **Grid** view. In the **List** view, use the column options to select the fields in the test case work item.



You can then view and edit these fields when you switch to the **Grid** view.

### Use Excel to copy test case information into a Grid view

You can copy test cases and test steps from an existing Excel worksheet. Copy the columns from Excel that you want to use for the title, action, and expected results fields. Copying does not copy column formatting, other than multiline, from the worksheet. Paste these columns into the **Grid** view, edit if necessary, and save them.



You can copy the data from the **Grid** view and paste it into your Excel worksheet. Copying does not copy test step formatting, other than multiline, into the worksheet.

**Note**

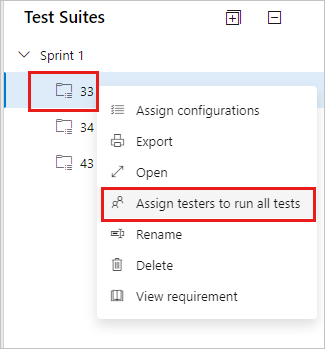
Do not use the Teams plugin for Excel to add or update test case work items. Excel cannot parse the format used to store test steps, and in some cases this may affect the formatting of the test case work items.

## Assign testers

You can reassign test cases so that a different tester can run them. You can assign all the test cases in a test suite to multiple testers, which is useful for acceptance testing.

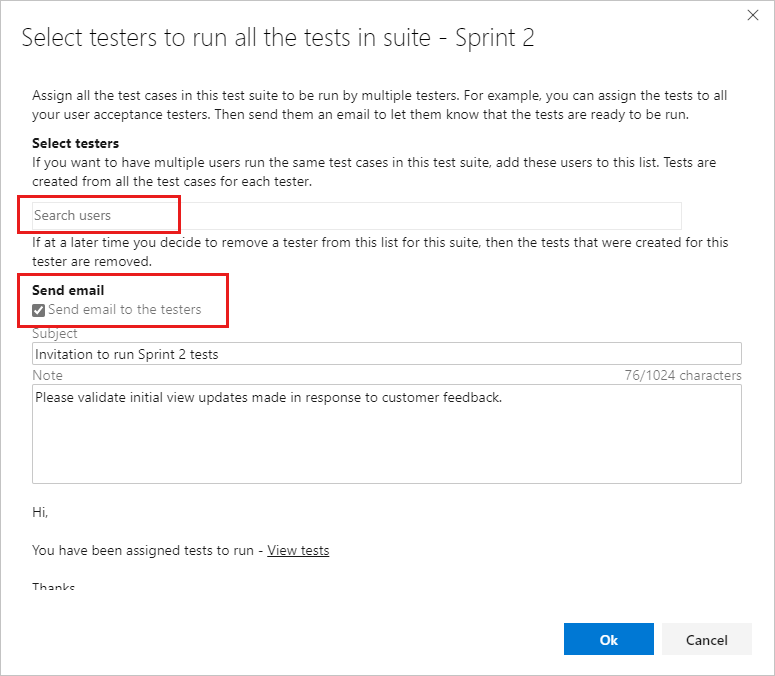
Testers need [Basic access](https://learn.microsoft.com/en-us/azure/devops/organizations/security/access-levels?view=azure-devops) to run tests from Azure Test Plans.

1. In the context menu for a test suite, select **Assign testers to run all tests**.



The **Select testers to run all the tests in suite** dialog box opens.

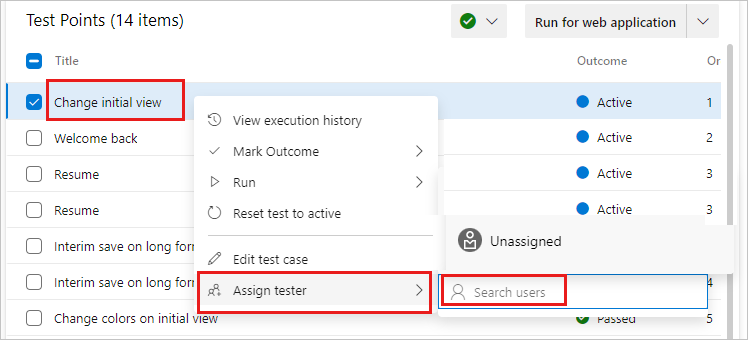
1. Add or remove testers from the list. After you select the testers, select **Send email** and edit the message as required so they know that tests are ready for them to run.



The email contains a link that testers can open to see the list of assigned tests.

You can assign an individual test case to a tester.

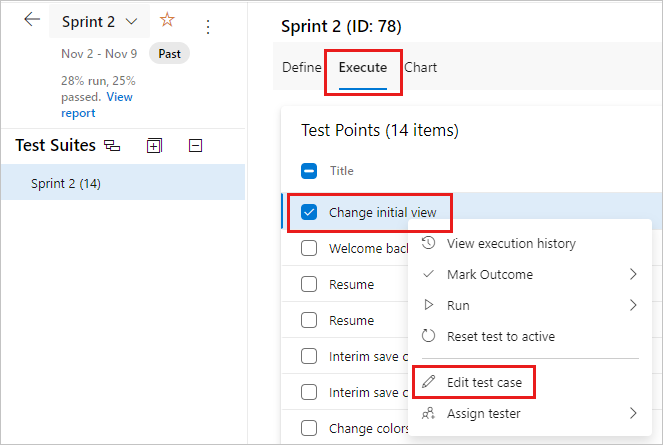
1. In the **Execute** tab for a test suite, select a test, and then open the context menu.



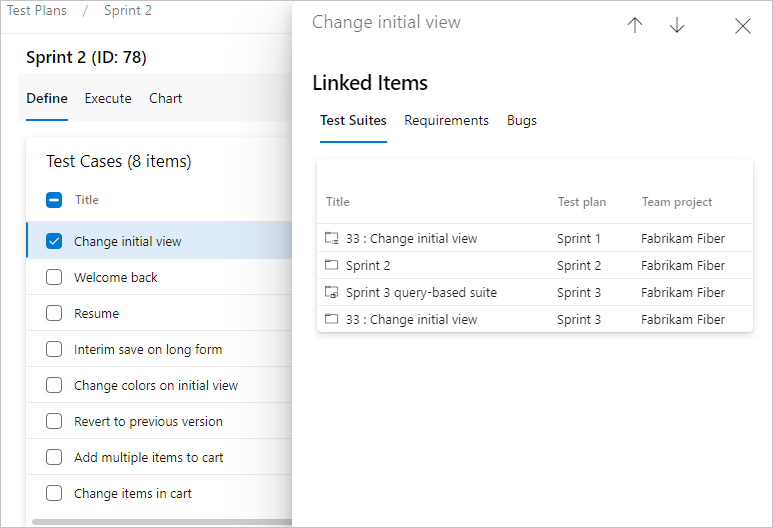
1. Select **Assign tester**. Search for and select a tester.

## Manage test cases

You can open a test case to view it or edit it. To open a test case in a test suite, in the **Define** tab, double-select the name of the test case to open. In the **Execute** tab, select a test case, open its context menu, and select **Edit test case**.



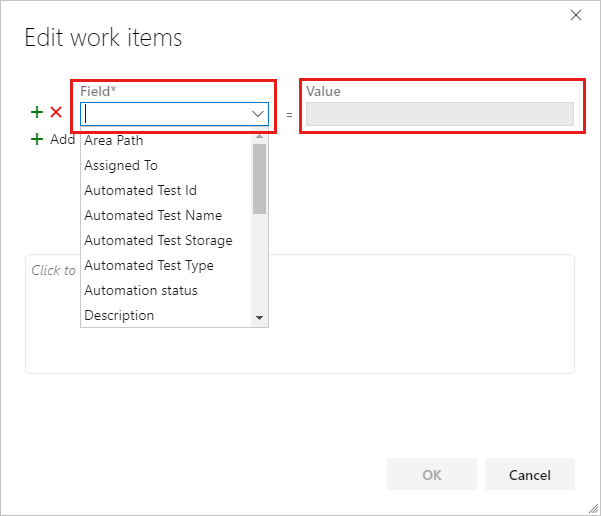
A test case can be linked to test suites, requirements, and bugs. To see linked items, in the **Define** tab, open the context menu for a test case, and select **View Linked Items**.



In the **Linked Items** dialog box, you can view **Test Suites**, **Requirements**, and **Bugs**.

## Bulk edit test cases

You can edit more than one test case at a time. Select several test cases in a test suite and select **Edit test case(s)**.



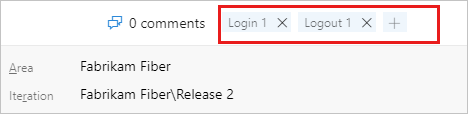
Select a **Field** and enter a **Value**. Select **Add new field** to add another field-value pair.

### Use tags for test cases

You can tag test cases and view only the ones with specific tags. For example, tag all the tests related to signing in so that you can rerun these tests if a bug is fixed for that page. You can filter on that tag from the **Test Plans** web portal.

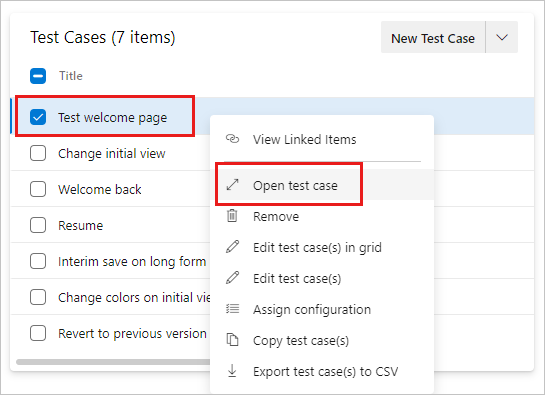
To add new tags to add to work items, you must have **Basic** access or higher and have the project-level **Create new tag definition permissions** set to **Allow**. for more information, see [Add work item tags](https://learn.microsoft.com/en-us/azure/devops/boards/queries/add-tags-to-work-items?view=azure-devops).

You can add and edit tags when you edit a test case, or bulk edit tags in the **Grid** view. You can also create suites based on queries when you use tags.

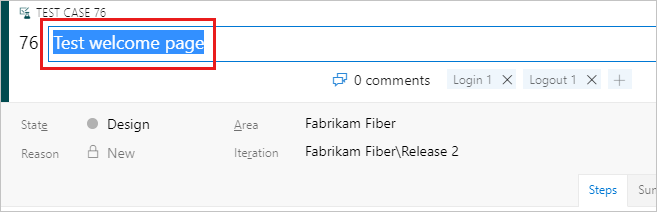


## Rename or remove test cases

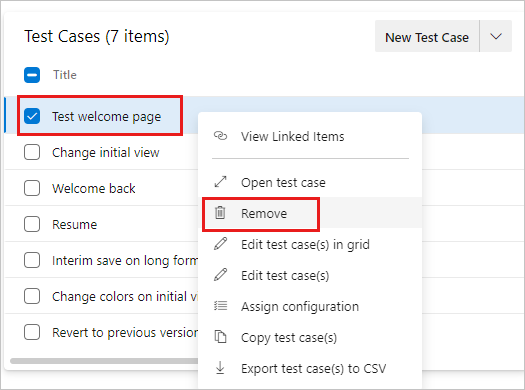
You can rename or remove test cases. Open the test case from its context menu.



Here you can rename the test case.



Or you can remove the test case from the test suite. From the context menu for the test case, select **Remove**.



To permanently delete test plans and test suites, you must be a member of the Project Administrators group or have the Area Path node-level [**Manage test plans** or **Manage test suites**](https://learn.microsoft.com/en-us/azure/devops/organizations/security/set-permissions-access-work-tracking?view=azure-devops#manage-test-artifacts) permission set to **Allow**. To manage or delete test artifacts, you must also have your [access level](https://learn.microsoft.com/en-us/azure/devops/organizations/security/access-levels?view=azure-devops) set to **Basic + Test Plans** or **Visual Studio Enterprise**. For more information, see [Delete test artifacts in Azure Boards](https://learn.microsoft.com/en-us/azure/devops/boards/backlogs/delete-test-artifacts?view=azure-devops).

## Next step

# Run manual tests

* Article
* 02/25/2023
* 11 contributors

Feedback

**Azure DevOps Services | Azure DevOps Server 2022 - Azure DevOps Server 2019 | TFS 2018**

Run your manual tests and record the test results for each test step using Microsoft Test Runner. You can run tests for both web applications and desktop apps.

Test Runner lets you run all active tests as a batch or specific test cases. Modify tests while running them. You can run tests for a specific build.

While testing, gather information like a screenshot or your testing actions. If you find an issue when testing, use Test Runner to create a bug. Include test steps, screenshots, and comments in the bug.

**Note**

You execute test points and not test cases. When you add a test case to a test suite, test point(s) are generated. A test point represents a unique combination of test case, test suite, configuration, and tester.

For example, a test case named Test login functionality, which has two configurations for the Edge and Chrome browsers, generates two test points. You can execute or run each of these test points. On execution, test results are generated. Through the test results view, or execution history, you can see all executions of a test point. The latest execution for the test point is what you see in the **Execute** tab.

## Supported clients and run options

In addition to running tests for a specific build, run options support various test scenarios. To see the options, in the **Execute** tab, select a test, and then select **Run with options**.

**Select test type and runner** offers these options:

* **Manual tests using Web Browser based runner** You can select a specific build to test, as described in [Run tests for a build](https://learn.microsoft.com/en-us/azure/devops/test/run-manual-tests?view=azure-devops#run-tests-for-a-build). Select **Find builds** to open the **Find builds** dialog box and search for a build to test against.
* **Manual tests using Test Runner client** You can run manual tests for desktop applications by using the Test Runner client. If necessary, install the [Test Runner desktop client](https://aka.ms/ATPTestRunnerDownload).
* **Automated tests using release stage** To run tests using a release stage, select a build. Select a release pipeline and stage to consume the build's artifacts. For more information, see [Run automated tests from test plans](https://learn.microsoft.com/en-us/azure/devops/test/run-automated-tests-from-test-hub?view=azure-devops).
* **Manual tests using Microsoft Test Manager 2017 client** Collects data from a local machine. Select a build. Select one or more data collectors: **Action log**, **Event log**, **Screen and voice recorder**, and **System information**.
* **Manual tests using Microsoft Test Manager 2015 or earlier client**

The Test Center in the Microsoft Test Manager client is a desktop-based manual testing solution. Testers can use it for manual testing needs. For more information, see [Guidance on Microsoft Test Manager usage](https://learn.microsoft.com/en-us/previous-versions/azure/devops/test/mtm/guidance-mtm-usage). To get Microsoft Test Manager, install [Visual Studio Enterprise](https://visualstudio.microsoft.com/downloads/) or [Visual Studio Test Professional](https://visualstudio.microsoft.com/vs/test-professional/).

## Prerequisites

* You must connect to a project. If you don't have a project yet, [create one](https://learn.microsoft.com/en-us/azure/devops/user-guide/sign-up-invite-teammates?view=azure-devops).
* You must be added to a project. To get added, [Add users to a project or team](https://learn.microsoft.com/en-us/azure/devops/organizations/security/add-users-team-project?view=azure-devops).
* To view or run manual or automated tests, you must have **Basic** access or higher.

To learn more, see [Manual test access and permissions](https://learn.microsoft.com/en-us/azure/devops/test/manual-test-permissions?view=azure-devops).

## Work with the TCM command-line tool

To exercise the TCM command-line tool, you must have the following in place:

* Installed Visual Studio 2017 Professional or earlier version. You access TCM from the command prompt and the following directories: %programfiles(x86)%\Microsoft Visual Studio\2017\Professional\Common7\IDE  
  %programfiles(x86)%\Microsoft Visual Studio\2017\Enterprise\Common7\IDE %programfiles(x86)%\Microsoft Visual Studio 14.0\Common7\IDE
* To run a TCM command, specify the /collection and /teamproject parameters, and /login as needed.

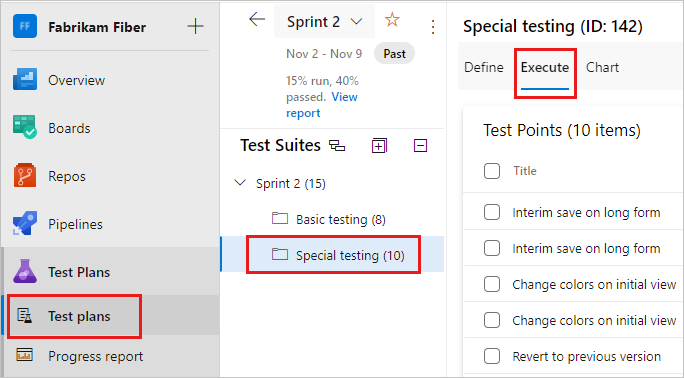
| **Parameter** | **Description** |
| --- | --- |
| **/collection**:CollectionURL | Required. Specifies the URI of the team project collection.The format for the URI is as follows: - For Azure DevOps Services: http://dev.azure.com/OrganizationName - For Azure DevOps Server: http://ServerName:Port/VirtualDirectoryName/CollectionName. If no virtual directory is used, then the format for the URI is as follows:http://ServerName:Port/CollectionName |
| **/teamproject**:project | Required. The name of the project that contains the test objects you want to clone or import automated tests into. |
| **/login**:username,[password] | Optional. Specifies the name and password of a valid Azure DevOps user and who has permissions to run the command. Use this option if your Windows credentials don't have the appropriate permissions, or you're using basic authentication, or you're not connected to a domain. |

* You must be a valid member of the project(s) you want to access and have the required permissions based on the commands you run. To learn more, see [Manual test access and permissions](https://learn.microsoft.com/en-us/azure/devops/test/manual-test-permissions?view=azure-devops).
* To clone or import test objects, you must have been granted the same access levels required to add test plans and test suites as described in [Prerequisites](https://learn.microsoft.com/en-us/azure/devops/test/run-manual-tests?view=azure-devops#prerequisites).

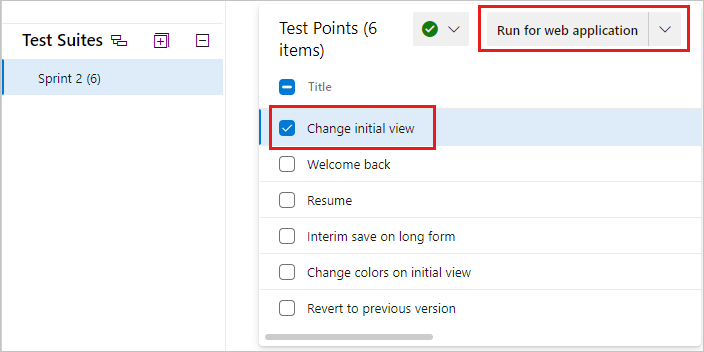
## Run tests for web apps

Follow these steps to run tests for web applications.

1. From the web portal, open your project and select **Test Plans** > **Test plans**.
2. If you haven't already, [create your manual tests](https://learn.microsoft.com/en-us/azure/devops/test/create-test-cases?view=azure-devops#test-cases).
3. Select **Mine** or **All**, or use **Filter by title** to find your test plan and select it. Select the **Execute** tab.

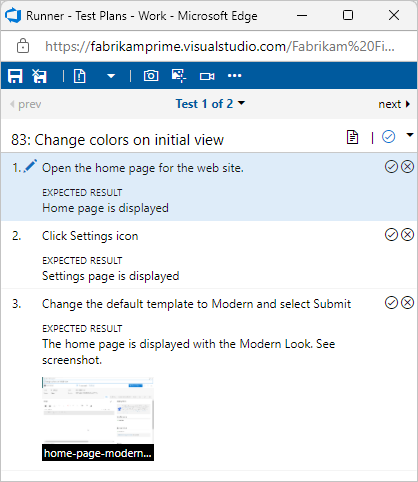


1. Select one or more tests, or all the tests from a test suite. Then select **Run for web application**.



Microsoft Test Runner opens and runs in a new browser.

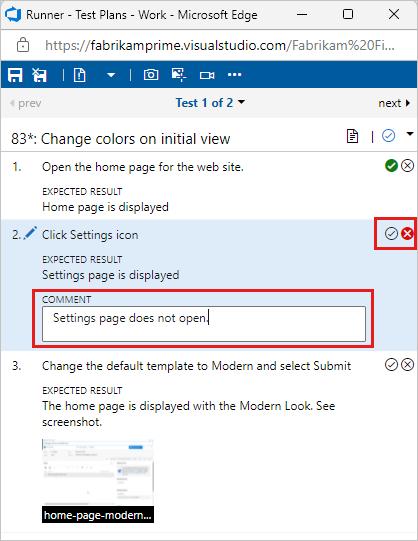
1. Start the app that you want to test.



Your app doesn't have to run on the same computer as Test Runner. You just use Test Runner to record which test steps pass or fail while you manually run a test.

For example, you might run Test Runner on a desktop computer and run your store app for Windows 8 that you test on a Windows 8 tablet.

1. Mark each test step as either passed or failed based on the expected results.



If a test step fails, you can enter a comment on why it failed or [collect diagnostic data for the test](https://learn.microsoft.com/en-us/azure/devops/test/collect-diagnostic-data?view=azure-devops). You can also [Create or add to a bug](https://learn.microsoft.com/en-us/azure/devops/test/run-manual-tests?view=azure-devops#create-or-add-to-a-bug).

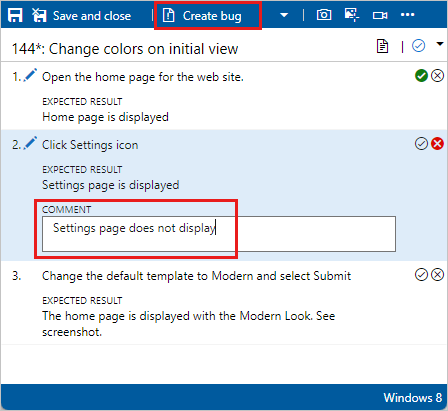
**Important**

Any test step that has expected result is called a validation test step. Testers must mark a test step with a status if it is a validation test step. The overall result for a test case reflects the status of all the test steps that the tester marked. Therefore, the test case will have a status of failed if the tester marked any test step as failed or not marked.

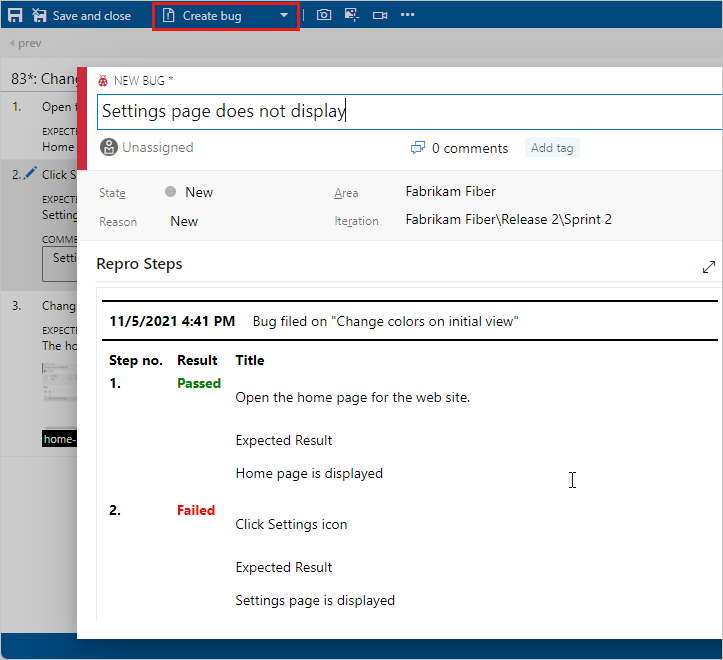
## Create or add to a bug

If a test step fails, enter a comment about the behavior and collect diagnostic data. You can create a bug to capture and track the issue. You can also update an existing bug with information about the failure.

1. When a step fails, enter a comment and select **Create bug**.



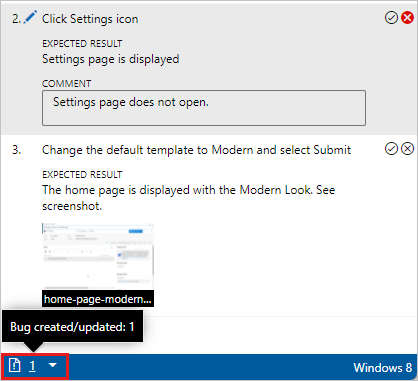
1. In the **New bug** dialog box, enter a name for the bug.



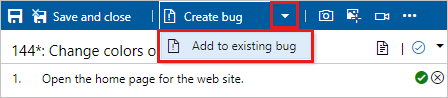
The steps and your comments are automatically added to the bug. If Test Runner is running in a web browser window, you can copy a screenshot from the clipboard directly into the bug.

1. You can assign the bug, enter comments, or link to other issues. Select **Save & Close** when done. The test case is linked to the bug you created.

You can see any bugs reported during your test session.



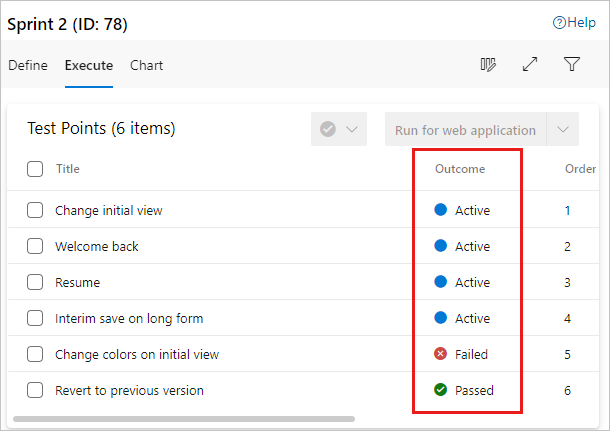
Instead of creating a bug, you can update an existing bug with information about this step. Select **Add to existing bug** from the **Create bug** drop-down menu.



## Save results, close the session, and review results

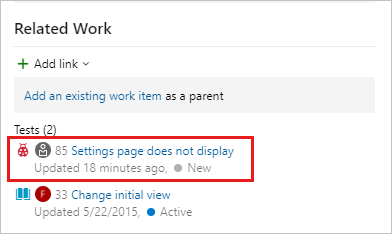
After you complete testing, save your results, close the session, and review test results.

1. When you've run all your tests, select **Save and close**. All the test results are stored in Azure Test Plans.
2. View the testing status for your test suite. You see the most recent results for each test.



If you haven't run a test yet, its state is active. Reset the state of a test to active if you want to rerun it.

1. Open a test suite and choose the test case in the **Related Work** section. Then use the child links in the **Related Work** section of that work item to view the bugs filed by the tester.

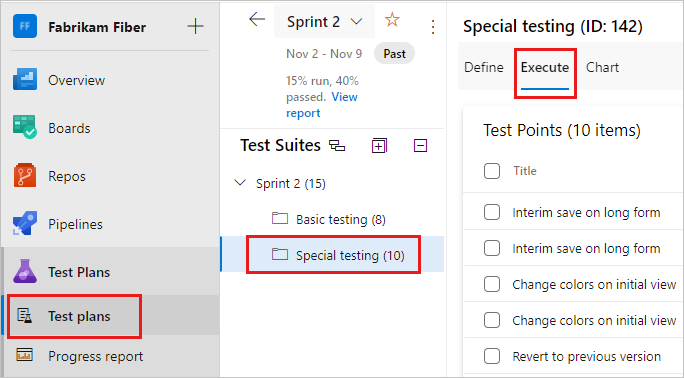


You can run tests offline and then import the results. For more information, see the [Offline Test Execution extension](https://marketplace.visualstudio.com/items?itemName=ms-devlabs.OfflineTestExecution).

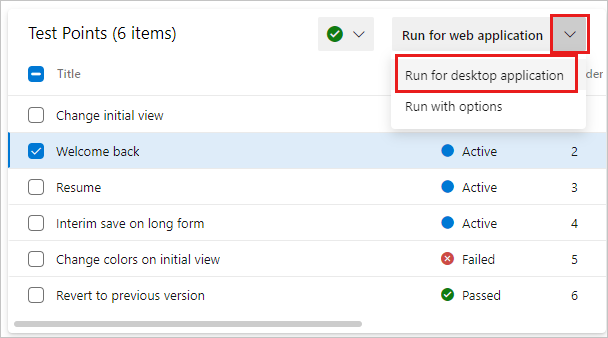
## Run tests for desktop apps

If you want to collect more diagnostic data for your desktop application, run your tests using Test Runner client.

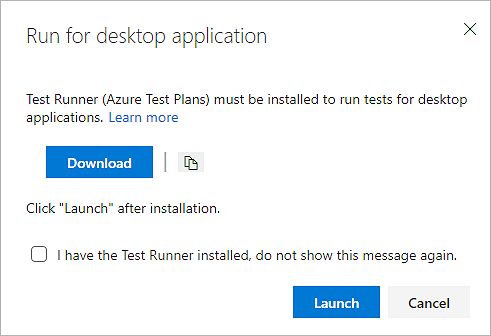
1. From the web portal, open your project and select **Test Plans** > **Test plans**.
2. Select **Mine** or **All**, or use **Filter by title** to find your test plan and select it. Select the **Execute** tab.



1. Launch Test Runner from Azure Test Plans by selecting **Run for desktop application** from the dropdown menu.



1. If necessary, download and install the [Test Runner desktop client](https://aka.ms/ATPTestRunnerDownload).



**Note**

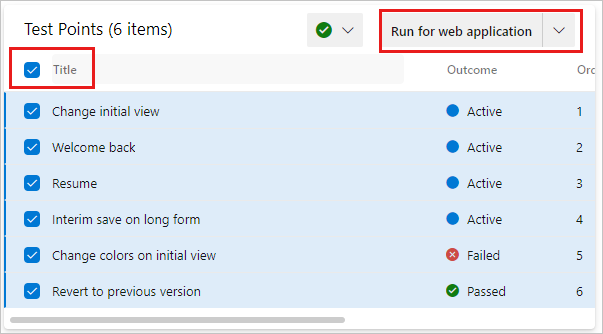
Check that the Test Runner client is available for your platform. Currently, the Test Runner client is available only for x64.

1. Select **Launch** and start testing as described in the previous section. For more information about data collection, see [Collect diagnostic data while testing](https://learn.microsoft.com/en-us/azure/devops/test/collect-diagnostic-data?view=azure-devops).

## Run all tests

You can run all the tests in a test suite at once.

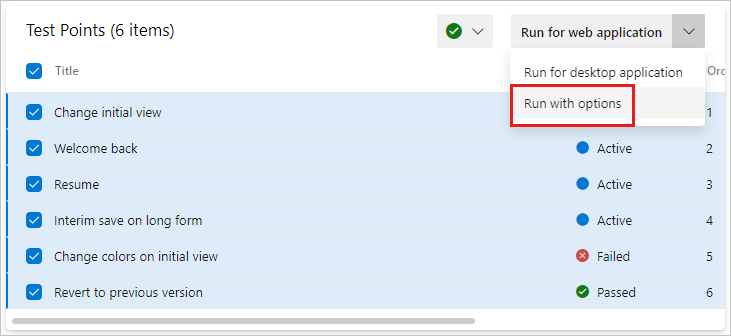
Select a test suite and select **Run for web application** or **Run for desktop application** to run all the active tests.



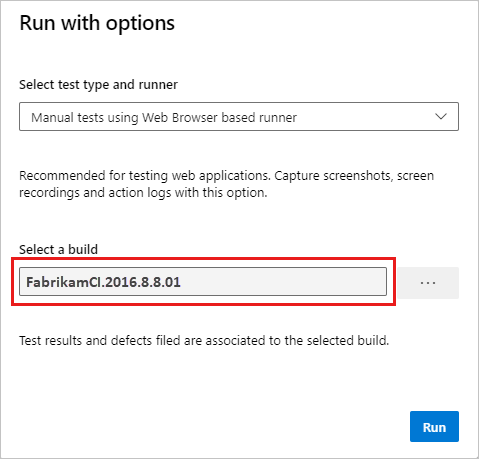
## Run tests for a build

Choose a build to run tests against.

1. From the dropdown, select **Run with options**.



1. In the **Run with options** dialog box, select the build you want.



**Note**

The selected build must be from the project in which the tests are defined.

You can select a build for the following options:

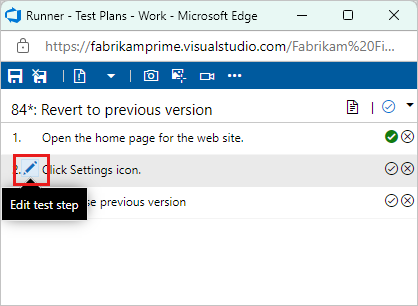
* Manual tests using Web Browser based runner
* Automated tests using release stage
* Manual tests using Microsoft Test Manager 2017 client

The fields that the dialog box offers differ depending on which option you select. For more information, see [Supported clients and run options](https://learn.microsoft.com/en-us/azure/devops/test/run-manual-tests?view=azure-devops#supported-clients-and-run-options).

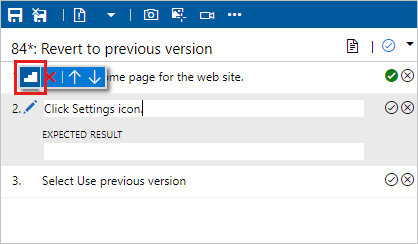
Any bug filed during the run is associated with the selected build. The test outcome will be published against that build.

## Modify a test step during a test run

Fix problems with your test steps while the test is still running. Select the **Edit test step** icon.



You can insert, reorder, or delete steps. You can also edit the text itself.



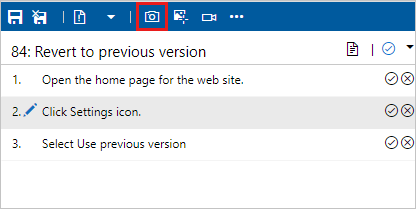
## Capture rich diagnostic data

While running your tests, you can add screenshots, capture actions as a log, and record video or voice.

### Add a screenshot

Add a screenshot to the test results while running a test.

If you use Google Chrome or Firefox, use the web runner to take screenshots of the web app while testing. For Microsoft Internet Explorer or Microsoft Edge browsers, or for desktop app testing, use the [Test Runner desktop client](https://aka.ms/ATPTestRunnerDownload).

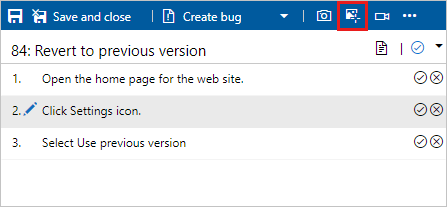


For more information, see [Collect diagnostic data](https://learn.microsoft.com/en-us/azure/devops/test/collect-diagnostic-data?view=azure-devops#web-screenshot).

### Capture actions from a test

Capture your actions on the application as a log.

If you use Google Chrome or Firefox, use the web runner capture your actions on the web app as image logs while testing. For Microsoft Internet Explorer or Microsoft Edge browsers, or for desktop app testing, use the [Test Runner desktop client](https://aka.ms/ATPTestRunnerDownload).

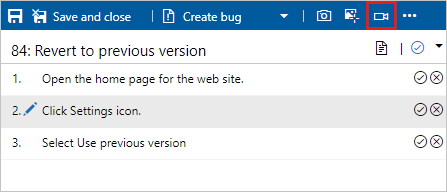


For more information, see [Collect diagnostic data](https://learn.microsoft.com/en-us/azure/devops/test/collect-diagnostic-data?view=azure-devops#web-log).

### Capture screen recordings of your app being tested

Capture screen recordings of my app during testing.

If you use Google Chrome or Firefox, use the web runner to capture screen recordings of your web and desktop apps while testing. For Microsoft Internet Explorer or Microsoft Edge browsers, or for desktop app testing, use the [Test Runner desktop client](https://aka.ms/ATPTestRunnerDownload).



For more information, see [Collect diagnostic data](https://learn.microsoft.com/en-us/azure/devops/test/collect-diagnostic-data?view=azure-devops#web-recording).

## Run tests with TCM

You can run tests that are part of a test plan using the TCM command-line tool. This tool lets you create and start a test run, and then manage all your existing test runs. Use the tcm commands documented here to accomplish these tasks.

[List test runs](https://learn.microsoft.com/en-us/azure/devops/test/run-manual-tests?view=azure-devops#list-test-runs) | [Create test runs](https://learn.microsoft.com/en-us/azure/devops/test/run-manual-tests?view=azure-devops#create-test-runs) | [Execute test runs](https://learn.microsoft.com/en-us/azure/devops/test/run-manual-tests?view=azure-devops#execute-test-runs) | [Abort test runs](https://learn.microsoft.com/en-us/azure/devops/test/run-manual-tests?view=azure-devops#abort-test-runs) | [Delete test runs](https://learn.microsoft.com/en-us/azure/devops/test/run-manual-tests?view=azure-devops#delete-test-runs) | [Export test runs](https://learn.microsoft.com/en-us/azure/devops/test/run-manual-tests?view=azure-devops#export-test-runs) | [Publish test runs](https://learn.microsoft.com/en-us/azure/devops/test/run-manual-tests?view=azure-devops#publish-test-runs)

### List test runs

Use tcm run /list to list the runs available in a test plan and to show their **ID**. The **ID** corresponds to the work item ID defined when the run was created.

tcmCopy

tcm run /list /collection:teamprojectcollectionurl /teamproject:project

[/planid:id | /querytext:query] [/login:username,[password]]

| **Parameter** | **Description** |
| --- | --- |
| **/planid**:id | Optional. Indicates that only those test runs associated with the specified test plan are returned in the list. |
| **/querytext**:query | Optional. Specifies the query to use to list a subset of test runs. |

For descriptions of /collection, /teamproject and /login parameters, see [Work with the TCM command-line tool](https://learn.microsoft.com/en-us/azure/devops/test/run-manual-tests?view=azure-devops#work-tcm-cli).

**Example**

The following command lists the test runs defined for the Fabrikam Fiber project hosted in the fabrikamprime organization. The **ID** and **Title** correspond to the work item ID and title defined for the test run. For example, test run 1000052 is titled Test Plan for Cycle 1 (Manual).

tcmCopy

tcm run /list /collection:https://fabrikamprime.visualstudio.com /teamproject:"Fabrikam Fiber"

Id Title Owner Date Completed

--------- ---------------------------------- ------------------- -----------

1000006 Sprint 2 (Manual) Thomas Margand 11/5/2021

1000032 33 : Change initial view (Manual) Danielle Brasseur 11/11/2021

1000040 Sprint 2 (Manual) Thomas Margand 11/16/2021

1000042 Sprint 3 (Manual) Thomas Margand 11/16/2021

1000046 Special testing (Manual) Nicoletta Guibord 11/18/2021

1000052 Test Plan for Cycle 1 (Manual) Bukhosi Bhengu 12/1/2021

1000060 Game Shopping (Manual) Bukhosi Bhengu 12/6/2021

### Create test runs

Use tcm run /create to create a test run associated with the specified test plan. In addition to the test plan, you also define the test suite and configuration you want to use by their corresponding **ID**. You can gather these **IDs** using the tcm plans /list, tcm suites /list, and tcm configs /list commands.

tcmCopy

tcm run /create /title:title /planid:id /collection:CollectionURL /teamproject:project

(suiteid:id /configid:configid | /querytext:query)

[/settingsname:name] [/owner:owner] [/builddir:directory]

[/testenvironment:name] [/login:username,[password]] [/include]

| **Parameter** | **Description** |
| --- | --- |
| **/title**:title | Specifies the title that you want to use for the test run that you create. |
| **/planid**:id | Specifies the test plan that where you want to create the test run. |
| **/suiteid**:id | Specifies the test suites that you want to use for your test run. |
| **/configid**:id | Specifies the test configuration you want to run for your test suites. |
| **/querytext**:query | Optional if you specify suiteid and configid. Specifies the query to use to select the tests that you want to run.  **Tip:** You can use the /querytest parameter to run more than one test suite. For example: querytext:“SELECT \* FROM TestPoint WHERE (ConfigurationId=20 OR ConfigurationId=21) AND (Suiteid=1185 OR Suiteid=1186)”. |
| **/settingsname**:name | Optional. Specifies the test settings that you want to use for this test run. If you don't select test settings, the default test settings in the test plan are used. |
| **/owner**:owner | Optional. Specifies the owner of the test run. |
| **/builddir**:directory | Optional. Specifies the build directory to use to locate the test assemblies for the test. If this isn't specified, the build location is used based on the build that is currently assigned to the test plan. |
| **/testenvironment**:name | Optional. Specifies the test environment that you want to use for this test run. If you don't select a test environment, the default test environment in the test plan is used. |
| **/include** | Optional. Specifies that all tests that are selected for the test run are included, even if the tests are not currently set to the Active state. |

For descriptions of /collection, /teamproject and /login parameters, see [Work with the TCM command-line tool](https://learn.microsoft.com/en-us/azure/devops/test/run-manual-tests?view=azure-devops#work-tcm-cli).

**Example**

The following command creates a test run called **MyTestRun** in the test plan with **ID** 77. The run uses the test suite with **ID** 161 and the test configuration with **ID** 9. The run is defined for the Fabrikam Fiber project hosted in the fabrikamprime organization.

In this example, a test run is created with an **ID** of 1000082.

tcmCopy

tcm run /create /title:MyTestRun /planid:77 /collection:https://fabrikamprime.visualstudio.com /teamproject:"Fabrikam Fiber" /suiteid:161 /configid:9

Run created with ID: 1000082.

### Execute test runs

Use tcm run /execute to kick off one of the runs in your test plan. The **ID** you specify corresponds to the work item ID defined when the run was created. You can see a list of all test run IDs with the [tcm run /list](https://learn.microsoft.com/en-us/azure/devops/test/run-manual-tests?view=azure-devops#list-test-runs) command.

tcmCopy

tcm run /execute /id:id /collection:teamprojectcollectionurl /teamproject:project [/login:username,[password]]

| **Parameter** | **Description** |
| --- | --- |
| **/id**:id | Specifies the **ID** for the test run that you want to run. |

For descriptions of /collection, /teamproject and /login parameters, see [Work with the TCM command-line tool](https://learn.microsoft.com/en-us/azure/devops/test/run-manual-tests?view=azure-devops#work-tcm-cli).

**Example**

The following command starts a test run for the **ID** 1000082 for the Fabrikam Fiber project hosted in the fabrikamprime organization. The results are returned in your CLI window.

tcmCopy

tcm run /execute /id:1000082 /collection:https://fabrikamprime.visualstudio.com /teamproject:"Fabrikam Fiber"

Executing run: MyTestRun

Results

------------------------

Total: 2

Passed: 1

Failed: 1

Inconclusive: 0

### Abort test runs

Use tcm run /abort to cancel a test run that is in progress. The **ID** you specify corresponds to the work item ID defined when the run was created.

tcmCopy

tcm run /abort /id:id /collection:teamprojectcollectionurl /teamproject:project [/login:username,[password]]

| **Parameter** | **Description** |
| --- | --- |
| **/id**:id | Specifies the **ID** for the test run that you want to cancel. |

For descriptions of /collection, /teamproject and /login parameters, see [Work with the TCM command-line tool](https://learn.microsoft.com/en-us/azure/devops/test/run-manual-tests?view=azure-devops#work-tcm-cli).

**Example**

The following command stops the test run with the **ID** 1000082 for the Fabrikam Fiber project hosted in the fabrikamprime organization. The results confirm the **ID** and **Title** of the cancelled run.

tcmCopy

tcm run /abort /id:1000082 /collection:https://fabrikamprime.visualstudio.com /teamproject:"Fabrikam Fiber"

Run with ID [1000082] and title [MyTestRun] has been aborted.

### Delete test runs

Use tcm run /delete to delete a test run from your test plan. The **ID** you specify corresponds to the work item ID defined when the test run was created.

tcmCopy

tcm run /delete /id:id [/noprompt] /collection:teamprojectcollectionurl /teamproject:project [/login:username,[password]]

| **Parameter** | **Description** |
| --- | --- |
| **/id**:id | Specifies the **ID** for the test run that you want to delete. |
| **/noprompt** | Optional. Specifies that the user isn't prompted to confirm deletion of a test run. |

For descriptions of /collection, /teamproject and /login parameters, see [Work with the TCM command-line tool](https://learn.microsoft.com/en-us/azure/devops/test/run-manual-tests?view=azure-devops#work-tcm-cli).

**Example**

The following command deletes the test run with the **ID** 1000082 for the Fabrikam Fiber project hosted in the fabrikamprime organization. The user is prompted to confirm that they want to delete the specified test run and the result is provided.

tcmCopy

tcm run /delete /id:1000082 /collection:https://fabrikamprime.visualstudio.com /teamproject:"Fabrikam Fiber"

Are you sure you want to delete run [MyTestRun]? (Yes/No) y

Run [MyTestRun] has been deleted.

### Export test runs

Use tcm run /export to export a test run to a specified location. The **ID** you specify corresponds to the work item ID defined when the run was created.

tcmCopy

tcm run /export /id:id /resultsfile:path /collection:teamprojectcollectionurl /teamproject:project [/login:username,[password]]

| **Parameter** | **Description** |
| --- | --- |
| **/id**:id | Specifies the test run **ID** that you want to export. |
| **/resultsfile**:path | Specifies a location and filename for the test run you want to export. |

For descriptions of /collection, /teamproject and /login parameters, see [Work with the TCM command-line tool](https://learn.microsoft.com/en-us/azure/devops/test/run-manual-tests?view=azure-devops#work-tcm-cli).

**Example**

The following command specifies that the test run with the **ID** 1000082 for the Fabrikam Fiber project hosted in the fabrikamprime organization is exported to c:\temp\ResultsForDeveloper.trx.

tcmCopy

tcm run /export /id:1000082 /resultsfile:"c:\temp\ResultsForDeveloper.trx" /collection:https://fabrikamprime.visualstudio.com /teamproject:"Fabrikam Fiber"

### Publish test runs

Use tcm run /publish to publish the results from a Visual Studio test run results file for a specified test plan.

tcmCopy

tcm run /publish /suiteid:id /configid:id /resultowner:owner /resultsfile:path

/collection:teamprojectcollectionurl /teamproject:project [/title:runtitle]

[/runowner:owner] [/build:buildnumber /builddefinition:builddefinition]

[/flavor:flavor] [/platform:platform] [/assignfailurestouser:user]

[/login:username,[password]] [/buildverification]

| **Parameter** | **Description** |
| --- | --- |
| **/suiteid**:id | Specifies the test suite to use when you publish a test run. |
| **/configid**:id | Specifies which test configuration you want to use when you publish a test run. |
| **/resultowner**:owner | Specifies the owner for the test results. |
| **/resultsfile**:path | Specifies the location of the test run you want to publish. For example, "c:\temp\ResultsForDeveloper.trx". |
| **/title**:runtitle | Optional. Specifies a title that you want to use for the test run that you publish. |
| **/runowner**:owner | Optional. Specifies the owner of the test run. |
| **/build**:buildnumber | Optional. Specifies the build number to use to publish a test run. This parameter must be used with /builddefinition. |
| **/builddefinition**:builddefinition | Optional. Specifies the build definition to use to publish a test run. This parameter must be used with /build. |
| **/flavor**:flavor | Optional. Specifies the build flavor, such as **Release**. This parameter can only be used if the /build parameter is used. |
| **/platform**:platform | Optional. Specifies the build platform, such as **x86**. This parameter can only be used if the /build parameter is used. |
| **/assignfailurestouser**:user | Optional. Specifies the user to whom any failed tests in the test run are assigned. |
| **/buildverification** | Optional. Specifies that this test run contains build verification tests that check the basic functionality of your build. |

For descriptions of /collection, /teamproject and /login parameters, see [Work with the TCM command-line tool](https://learn.microsoft.com/en-us/azure/devops/test/run-manual-tests?view=azure-devops#work-tcm-cli).

**Example**

The following command publishes a test run for the test suite with **ID** 161 and test configuration with **ID** 9 and reassigns the owner. This updates the existing test points for the test cases in the test suite that is paired with this configuration and publishes the results in the specified .trx file. And any failed tests in the test run are assigned to the specified user.

tcmCopy

tcm run /publish /suiteid:167 /configid:9 /resultowner:"Thomas Margand" /resul

[Add users or groups to a team or project - Azure DevOps | Microsoft Learn](https://learn.microsoft.com/en-us/azure/devops/organizations/security/add-users-team-project?view=azure-devops&tabs=preview-page) (for learning and configure azure link)