

Tosca Installation : TechSupport@h2kinfosys.com

Introduction To Tricentis Tosca

What is automation testing? Why ?

Regression Testing?

Data Driven Testing?

Tricentis Tosca

Advantages

1. Model-Based Test Automation : (MBT), which separates test logic from test data.
2. Scriptless Automation
3. End-to-End Testing : Req - Scan - Test Plan, test case, execution, Defects, Report
4. Supports Web, Desktop, SAP, Mobile, Salesforce, API, PDF, etc..
5. Test Data Management
6. Continuous Integration & DevOps Support
7. Reduce redundancy and improve efficiency
8. Comprehensive Reporting & Analytics
9. Enterprise Scalability, Tosca is suitable for large teams and enterprise environments due to:

Centralized test case and data management, Integrate with QTest etc.

Drawbacks

1. High Licensing Costs

- **Enterprise Pricing:** Tosca is often considered expensive, especially for small to mid-sized companies. The cost can be prohibitive for teams that need a scalable solution across multiple users or departments.
- **Additional Modules:** Features like Test Data Management or Distributed Execution can add to the licensing costs.

2. Steep Learning Curve

- **Model-Based Testing Approach:** While powerful, Tosca's model-based approach may be challenging for testers unfamiliar with this paradigm.
- **Advanced Features Complexity:** Features like API testing, service virtualization, or CI/CD integration require deeper technical knowledge and additional training.

3. Resource Intensive

- **System Requirements:** Tosca can be heavy on system resources, especially when handling large test suites or executing multiple tests in parallel.
- **Setup and Maintenance:** Initial setup and configuration, particularly in complex environments, can be time-consuming.

4. Limited Open Source Support and Community

- **Proprietary Nature:** Unlike Selenium or Appium, Tosca is not open-source, limiting customizability and third-party plugin support.
- **Smaller Community:** Compared to open-source tools, there is a smaller community and fewer public resources or forums for troubleshooting.

5. Version Compatibility Issues

- **Frequent Updates:** Regular updates can sometimes lead to compatibility issues with existing test cases or integrations, requiring maintenance efforts.

6. Limited Support for Niche Technologies

- **Application Coverage:** While Tosca supports many technologies, support for less common or highly customized technologies may be lacking or require additional configuration.

Download and install the Tricentis Tosca

- 1) Register and login to Tricentis Website. Follow the instructions received in your email and activate your account.
- 2) <https://www.tricentis.com/software-testing-tool-trial-demo/trial-tosca>
- 3) Download the Latest version of Tosca
- 4) Follow the download and installation guidelines to install Tosca on your machine.

Overview of Tosca Commander

- 1) **How to create a new workspace** add Standard.tsu while creating a new workspace.
- 2) Overview of Tosca Commander
Overview of tabs / modules in Tosca Commander
 - 1) Test Planning
 - 2) TestCase Design
 - 3) Requirements
 - 4) Modules
 - 5) TestCases
 - 6) Execution , Execution list
 - 7) Issues
 - 8) Reporting

Tosca Project Info, Options, Settings, About Tosca, Accounts, License, Close

←

Info

Options

Settings

About Tosca

Accounts

License

Close

Tosca Commander: May

Login

User: Admin

Workspace

Type: SQLite

Path: C:\Tosca_Projects\Tosca_Workspaces\May-June-2025-Batch



Revert all

Revert all local changes and release checked out objects



Revert local

Revert all local changes but keep objects checked out



Backup

Backup the currently used Common Repository



Restore

Restore a backup into the Common Repository



Compact Workspace

Reduces disk usage and optimizes the performance of you



Empty local File Service Cache

Removes locally stored files from the File Service. The file redownloaded when you reconnect to the server.

Documentation :

https://documentation.tricentis.com/tosca/2420/en/content/resources/webhelp/cover_web.htm

Overview of sample application (Application Under Test)
Scan Controls (Web Elements) & Create test case & Run in Scratch Book

Overview of vehicle insurance application <https://sampleapp.tricentis.com/101/index.php>

Test Case:

Step by Step instructions to test a specific scenario / functionality.

The test case contains one or more steps.

=====

QTest test case

Test Case: Step by Step Instructions to test a specific requirement / functionality.

Test case contains steps

Test Case : Validate Vehicle Data Info

Step 1; Launch URL /home page

Step 2: Click on Automobile

Step 3: Enter Vehicle Data

Enter ...

Step 4: Close the browser window.

=====

Test case:

Launch Home page : <https://demowebshop.tricentis.com/>

Click on register

Enter valid data, Click on Register button

Close the browser

Create new work space with Standard.tsu

Create a New Module.

Install Chrome/ Edge extension . XSCAN will display a message to install the extension.

Scan application with basic scan on XSCAN , Add Controls.

Review the Properties of controls on Modules & test cases.

Create a new folder on Test case, create new test case, add module to the test case.

Fill the data in each control, run the test case using Run in Scratch Book.

Best Practice to create Test Cases.

https://documentation.tricentis.com/tricentis_cloud/en/content/create_tests/create_testcase.htm

Details of Controls :

https://documentation.tricentis.com/tosca/2420/en/content/best_practices/scan_identifiers.htm?Highlight=web%20controls

https://documentation.tricentis.com/tosca/2420/en/content/tosca_commander/scan_modules_overview.htm

https://documentation.tricentis.com/sap/tta_cloud_solex/en/content/references/supported_controls_web.htm?Highlight=delete%20control

Unable to click on button / Hyperlink on Tosca version 2024.1

Click is not working on 2024.1 version

https://support-hub.tricentis.com/open?sys_kb_id=48536ec783d6d210108092a6feaad32a&id=kb_article_view&sysparm_rank=11&sysparm_tsqueryId=b1141f052ba9665050fcfb2ece91bf74

Try first two options

Option 1: {Invoke[Click]}

Option 2: X

Below options, change the numbers and try

Option 3: {Click[50%][50%]}

Option 4: {Click[80px][20px]}

Fuzzy Search

You can launch Fuzzy Search from the following objects:

- TestCases
- Folders within TestCases
- TestSteps
- TestStepValues
- Reusable TestStepBlocks
- Recovery Scenarios

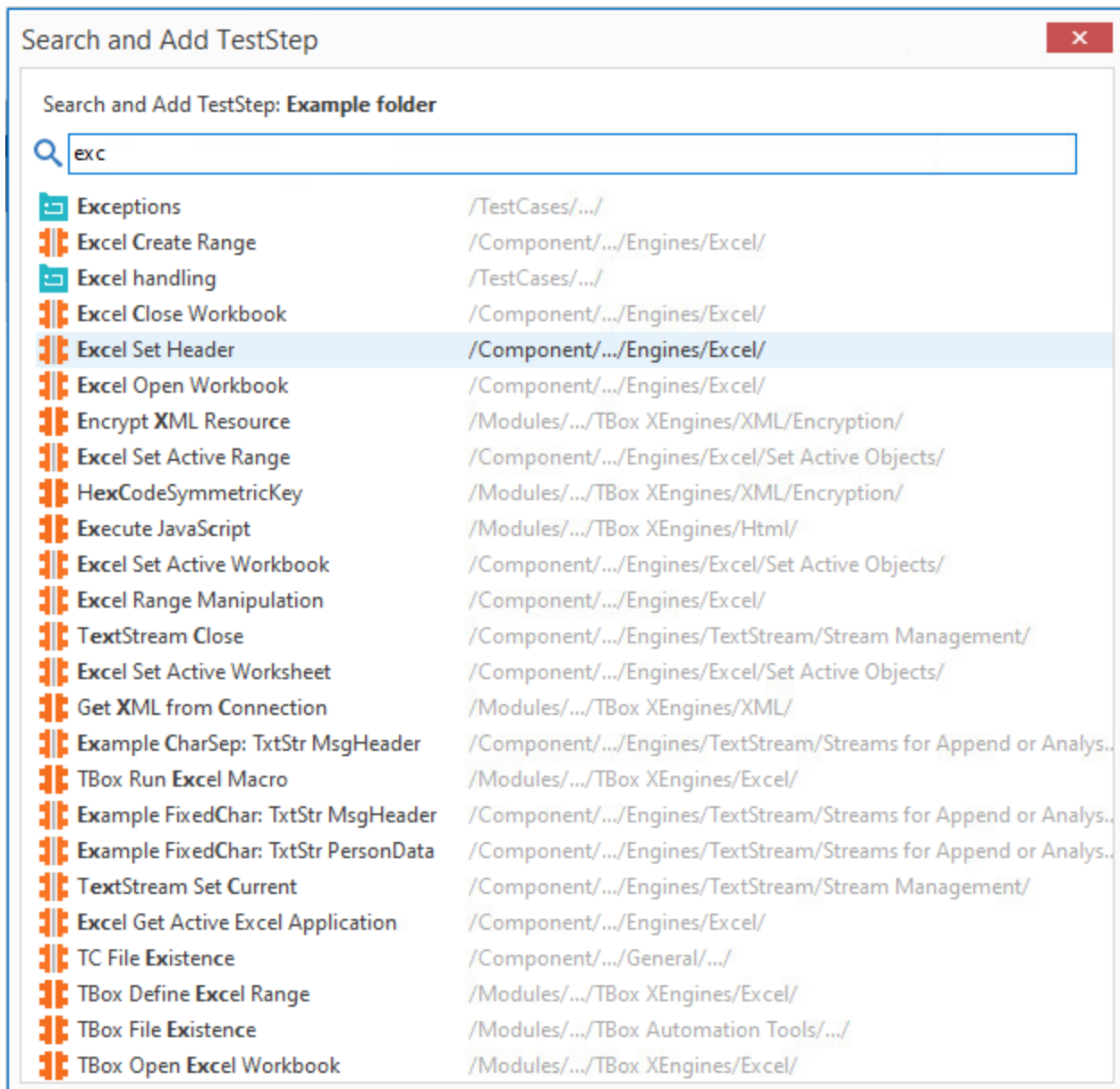
To launch the Fuzzy Search, right-click an object and select **Search and Add TestStep** from the context menu, or select the object and press **Ctrl + T**.

Search for objects

To search with Fuzzy Search, type a search term into the entry field.

The Fuzzy Search searches all Module, XModule, and Reusable TestStepBlock names for the search term and for parts of it. Characters that match the search term are highlighted. The order of the search results depends on how many search term characters match the entry.

The result is up to 25 objects along with their icons, names, and paths within the project.



Search results for **exc**

Create a TestStep

To create a TestStep from an entry in the search results, double-click it, or select the entry and press **ENTER**.

Test Configuration Parameter (TCP) | OpenURL, Close Browser, TBOX Window Operation

Test case:

Browser : Chrome

Launch Home page / URL <https://sampleapp.tricentis.com/101/index.php>

Click on Automobile

Enter Vehicle Information

Close the browser.

1. Test Configuration Parameter (TCP) : Add TCP on Test case folder / test case. In Parameter, select browser, Under Value, select the Browser name ex. Chrome.

Observation : Add TCP Browser on Folder level as Chrome and add another TCP on test case, browser as Edge and see the difference , which takes priority ?

2. OpenURL

To launch the URL

https://documentation.tricentis.com/tosca/2420/en/content/engines_3.0/xbrowser/open_url.htm

1. CloseBrowser

https://documentation.tricentis.com/tosca/2420/en/content/engines_3.0/xbrowser/close_browser.htm?Highlight=closeBrowser

Browser close / individual tabs or all tabs , if title is empty ?

If we want to close all browsers using CloseBrowser , use wildcard character * in title

2. TBOX WIndow Operation from standard module : To close any browser based or desktop or any window / like calculator application etc.

Add custom configuration parameters and use the parameters as needed.

Select Test case and add Configuration parameter and type the name of the parameter like num1, num2 etc..

Name	Value	DataType
num1	10	String
num2		String

Use these config parameters by calling the function CP {CP[num1]}

Name	Value	ActionMode	DataType	WorkState
n1	CP num1	Insert	String	
n2	CP num2	Insert	String	

Name	Value	ActionMode	DataType	WorkState
Add Response				
StatusCode	200 OK	Verify	String	
AddResult	MATH CP num1 + CP num2	Verify	String	

XSCAN - Advanced options

Add New controls , Advanced options, filter, search, click check box to add new control,
 Add extra properties.
 Drag Select to view details information of each control location
 Highlight on screen

How to Identify Unique Properties When a Page Contains Duplicate Controls

In some web applications, the same UI controls (like buttons or links) may appear multiple times on a single page. This is the case with the **Demo Webshop** site: <https://demowebshop.tricentis.com>, where categories such as **Books**, **Computers**, etc., are displayed twice.

Problem in Tosca Scanner

When you scan both sets of these duplicate controls in a single Tosca module, **Tosca Scanner** highlights them in **orange**, indicating they are **not uniquely identifiable** due to duplicated properties.

Solution: Selecting Unique Properties

To resolve this issue and make each control uniquely identifiable:

1. **Open Tosca Scanner** and inspect the duplicate controls.
2. Identify the properties Tosca captured by default (e.g., ID, Name, InnerText, Index).
3. Notice which properties are common across the duplicates (these are not helpful).
4. **Manually select additional or alternate properties** that are unique for each instance. This may include:
 - XPath
 - Position Index (only if stable)
 - CSS Class (if one of them has unique styling)
5. Once unique properties are selected, Tosca will stop marking them in orange, indicating the ambiguity has been resolved.

Best Practices

- Avoid relying solely on **Index** unless it's guaranteed not to change.
- Prefer **semantic properties** like labels, titles, or custom attributes.
- If possible, structure your modules so that each instance of the duplicated control is scanned in a separate module or business component for clarity.

Rescan / Add new controls / Delete existing controls (web elements / objects)

1. How to remove controls (Module Attributes) from Module and remove the control from the test case if a test case is created with that Module Attributes?
2. How to add new controls to existing modules using ReScan ? Auto update by adding them to the test case.
3. Disable control in Test Case (right click on specific control and Disable / Enable). Make sure to select the control under test case / step (Left side panel)

Self Healing feature in Tricentis Tosca

Demo Application:

<https://demowebshop.tricentis.com/>

Steps to Apply Self-Healing in Tosca for Updated Web Element Labels

1. Initial Scan and Test Case Execution

- Scan the web application to capture the objects (controls).
- Execute the relevant test case (e.g., login).
- The test should pass with the initial configuration.

2. Update the Control / Web Element Label

- Modify the label of the target web element directly in the browser using the Inspect tool (Developer Tools).

3. Re-run the Test Case

- Execute the same test case again.
- The test is expected to fail because Tosca cannot recognize the updated label of the control.

4. Rescan the Control with Self-Healing

- Reopen the module and rescan the control that was modified.
- In the Rescan window, select the control and click on the **Self-Healing** icon.
- Tosca will automatically select additional identifying properties.
- You can manually add more properties if necessary.
- Save and close the Rescan window.

5. Configure Self-Healing in the Test Case

- Open the test case in Tosca Commander.
- Add a configuration parameter named **SelfHealing**.
- Set the value to one of the following:
 - **Weighted** – Uses a weighted algorithm to identify the updated control.
 - **Combination** – Uses a combination of control properties for matching.
 - **False** – Disables self-healing if needed.

Add TCP (Test configuration parameter) **SelfHealingWeightThreshold** to set the value and specify the threshold for the weight. The default value is **0,75 (0.75)**, which means that all combinations that have a total weight that's smaller than 0,75 (0.75) will be disregarded during self-healing. Note that depending on your regional settings, you need to specify the value with either a period or a comma as decimal separator.

6. Run the Updated Test Case

- Ensure the web application reflects the updated label on the target element.
- Execute the test case again.

- Tosca should now recognize and interact with the updated control using the enhanced identification logic.

<https://documentation.tricentis.com/tosca/2420/en/content/tbox/selfhealing.htm?Highlight=self%20healing>

Best Practices:

Buffer Operations in Tricentis Tosca
Buffer : {B[parametername]}
XB {XB[ParameterName]} - To extract dynamic text

https://documentation.tricentis.com/tosca/2410/en/content/standard_subset/automation_tools/buffer_operations.htm

TBox Set Buffer : Create new buffer names and assign a value

↻ TBox Set Buffer				
↻ UserID	training@h2kinfosys.com	Input	String	
↻ Pwd	*****	Input	Password	
↻ <Buffername>			String	

TBox Name To Buffer : buffer the name of the TestCase that uses the Module.



In this example, you write the name of the TestCase **TestCase 12345** to a Buffer named **TC_Name**.

Name	Value	ActionMode	DataType	WorkState
↻ TestCase 12345				PLANNED
↻ TBox Name To Buffer				
■ Buffer	TC_Name	Input	String	

TBox Partial Buffer : Buffer parts of a value. Verify parts of a buffered value.

This example shows how to create and verify a partial Buffer.

- The first TestStep creates the Buffer **ExampleBuffer** with the value **Value 12345**.
- The second TestStep writes the last 5 characters of **ExampleBuffer** to the partial Buffer named **PartialBuffer**. Consequently, the value of **PartialBuffer** is **12345**.
- The third TestStep checks whether the value of **PartialBuffer** is **12345**.

	Name	Value	ActionMode	DataType	WorkState
▶	Partial Buffer example				PLANNED
▶	TBox Set Buffer				
	ExampleBuffer	Value 12345	Input	String	
▶	TBox Partial Buffer				
	Buffer	PartialBuffer	Input	String	
	Value	{B[ExampleBuffer]}	Input	String	
	Last	5	Input	Numeric	
▶	TBox Partial Buffer_1				
	Buffer	PartialBuffer	Input	String	
	Value	12345	Verify	String	

TBox Delete Buffer

remove individual or all existing Buffers



The TestStep in this example deletes the Buffers **Buffer1** and **Buffer2**.

	Name	Value	ActionMode	DataType	WorkState
▶	TBox Delete Buffer				
	Buffer	Buffer1	Select	String	
	Buffer	Buffer2	Select	String	

Delete two Buffers

TBox Iterate Array

Name	Value	ActionMode
Query customers		
Buffer all employee IDs		
Data Table		Select
Employee ID	employees	Buffer
While		
Condition		
Iterate employee IDs		
Array To Iterate	B employees	Input
Target Buffer	id	Input
Loop		
Call payroll endpoint		
employeeId	B id	Input

iterate a buffered array

How to extract dynamic text from the screen

XB {XB[ParameterName]}

<https://obstaclecourse.tricentis.com/Obstacles/81012>

NOTE: Remove the / uncheck the property Inner Text after scanning the text for Purchase Completed
Because the innertext has dynamic value and Tosca can not identify the attribute when we launch the
url again , ...

How to extract the dynamic text from the screen

Use the InterText property of the text

Purchase completed! We charged the total amount of \$1284.1 to your credit card!

Success Message	.InnerText==*We charged the total amount of {XB[Amount]} to your credit card!
Enter total amount	

Success Message	* We charged the total amount of XB Amount to your credit card!	Verify
Enter total amount	B Amount	Input

LOGS

Log Viewer

https://documentation.tricentis.com/tosca/2420/en/content/tbox/execution_logging.htm

If Condition, Do Loop , While loop in Tosca

If Condition:

<https://demowebshop.tricentis.com/>

Scan login, logout

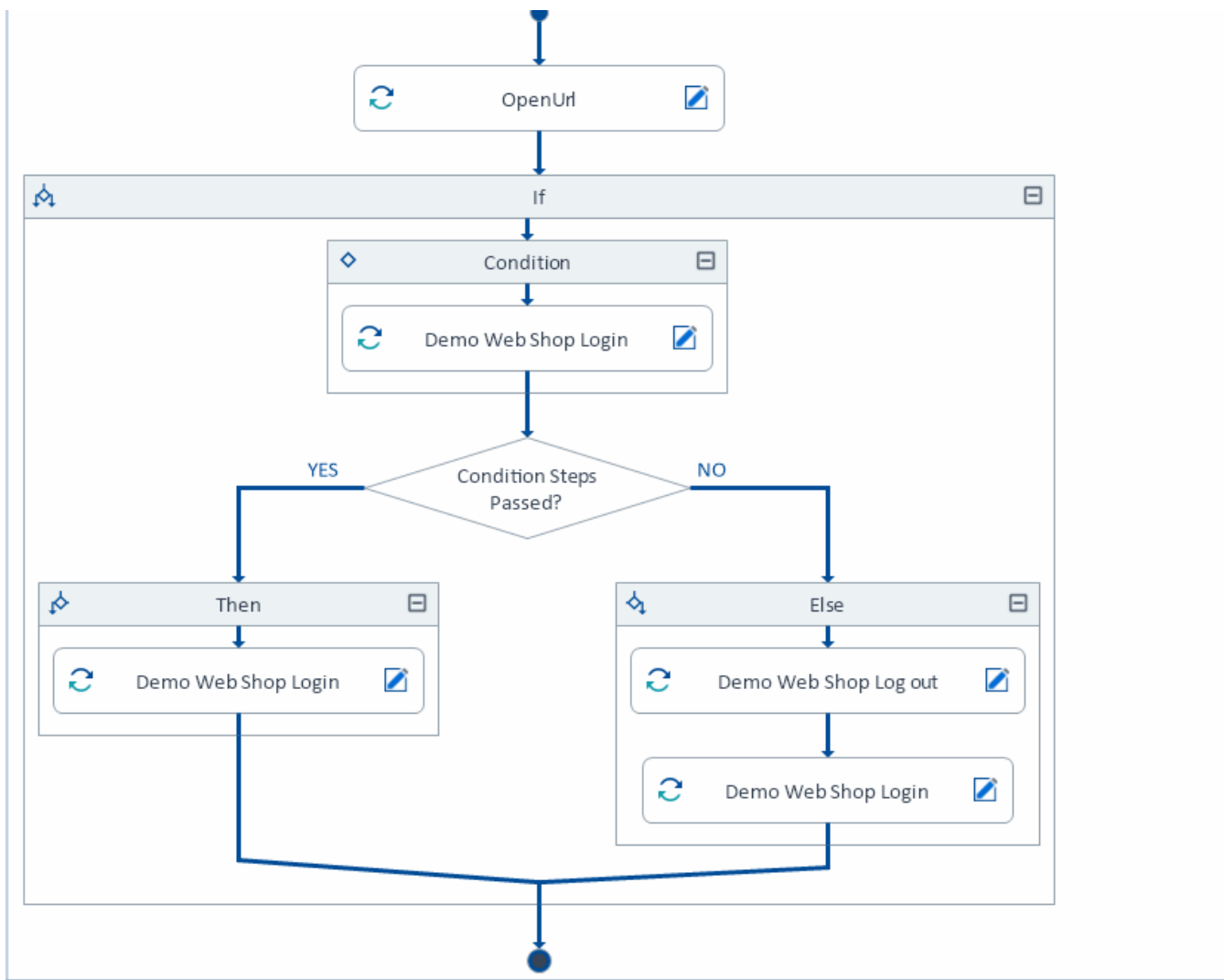
Test case:

Launch home page

Login

Purchase a product

Logout



- 1) Create a test case , add If condition
- 2) Add If condition then and else , under condition add Login module . add Login visible == true , then click on login, Under else condition , add click on logout button.

▲ If				
▲ Condition				
▲ Demo Web Shop Login				
Log in	Visible == True	Verify	String	
▲ Then				
▲ Demo Web Shop Login				
Log in	Click	Input	String	
▲ Else				
▲ Demo Web Shop Log out				
Log out	Click	Input	String	
▲ Demo Web Shop Login				
Log in	Click	Input	String	

Adding Multiple Conditions under If or else And / OR etc..

AND , && OR || all these different operators can be used

{B[X]}<={B[Y]} AND {B[X]}<5

{B[X]}<={B[Y]} OR {B[X]}<5

{B[X]}<={B[Y]} || {B[X]}<5

{B[X]}<={B[Y]} && {B[X]}<25

While Loop: Print 1 to 5 numbers

LOOP

Steps:

Create a test case and Add **TBOX Set Buffer** and create a buffer Count , set the value to 0

Select test case and add While Loop

Select **While** and see the property : **MaximumRepetitions** set to 30 by default to avoid infinite loop. We can update this number as needed.

Below the Condition:

TBox Evaluation Tool , for adding math logical operators like $x < 5$ etc.

Expression : $\{B[Count]\} \leq 5$

Below the loop

Add TBOX Set Buffer , and increase the value of the buffer Count by 1 each time it enter into the loop.

Count : $\{MATH[\{B[Count]\} + 1]\}$

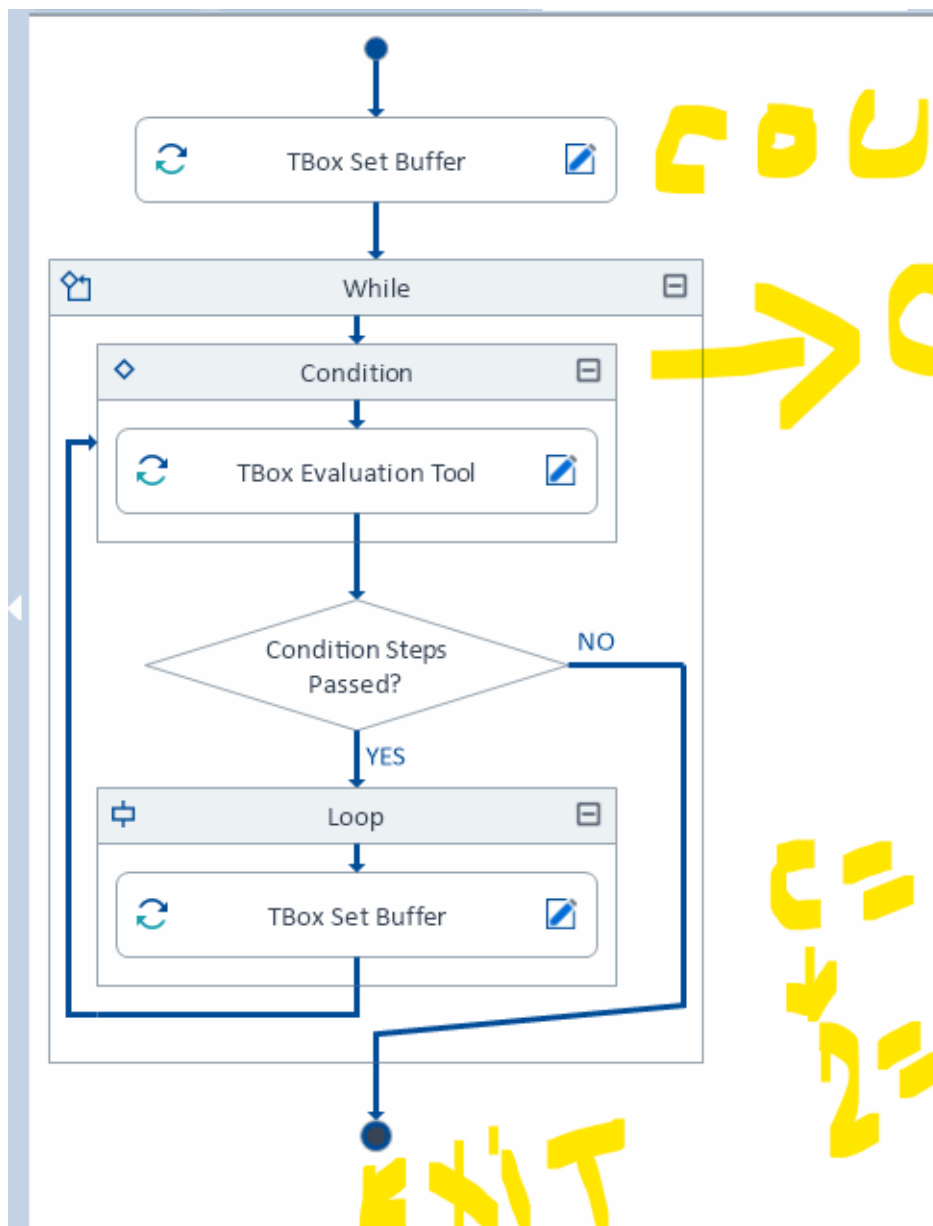
Run the Test case

While

Condition $< \text{statement} > \text{True}$

Loop

Execute the statement



Details

Test configuration

Name	Value	ActionMode	DataType	WorkState
While Loop				PLANNED
Print 1 to 5 numbers				
TBox Set Buffer				
Count	0	Input	Numeric	
<Buffername>			String	
While				
Condition				
TBox Evaluation Tool				
Expression	B Count < 5	Verify	String	
Expression			String	
Loop				
TBox Set Buffer				
Count	MATH B Count +1	Input	String	
<Buffername>			String	

Properties

Name	Value
NodePath	/TestCases/While Loop/Print 1 to 5 numbers/While
CreatedBy	Admin
CreatedAt	6/2/2025 5:08:19 PM
ModifiedBy	Admin
ModifiedAt	6/2/2025 5:21:38 PM
HasMissingReferences	False
Uniqueld	3a1a44a6-2322-5e54-44b4-bddc1d11615a
DisabledDescription	
IsPausable	False
MaximumRepetitions	30

ScratchBook

TBox Set Buffer		
Count		Buffer with name: "Count" has been set to value: "0"
While		
Repetition: 1		
Condition		
TBox Eva...		
Expre...		'0 < 5' evaluated to 'True'
Loop		
TBox Set ...		
Count		Buffer with name: "Count" has been set to value: "1"
Repetition: 2		
Repetition: 3		
Repetition: 4		
Repetition: 5		
Repetition: 6		
Condition		
TBox Eva...		
Expre...		'5 < 5' evaluated to 'False'

Do Loop:

Print numbers from 1 to 10

Steps:

Create a test case and Add **TBOX Set Buffer** and create a buffer Count , set the value to 0

Select test case and add While Loop

Select **While** and see the property : **MaximumRepetitions** set to 30 by default to avoid infinite loop. We can update this number as needed.

Below the Condition:

TBox Evaluation Tool , for adding math logical operators like $x < 5$ etc.

Expression : $\{B[Count]\} \leq 10$

Below the loop

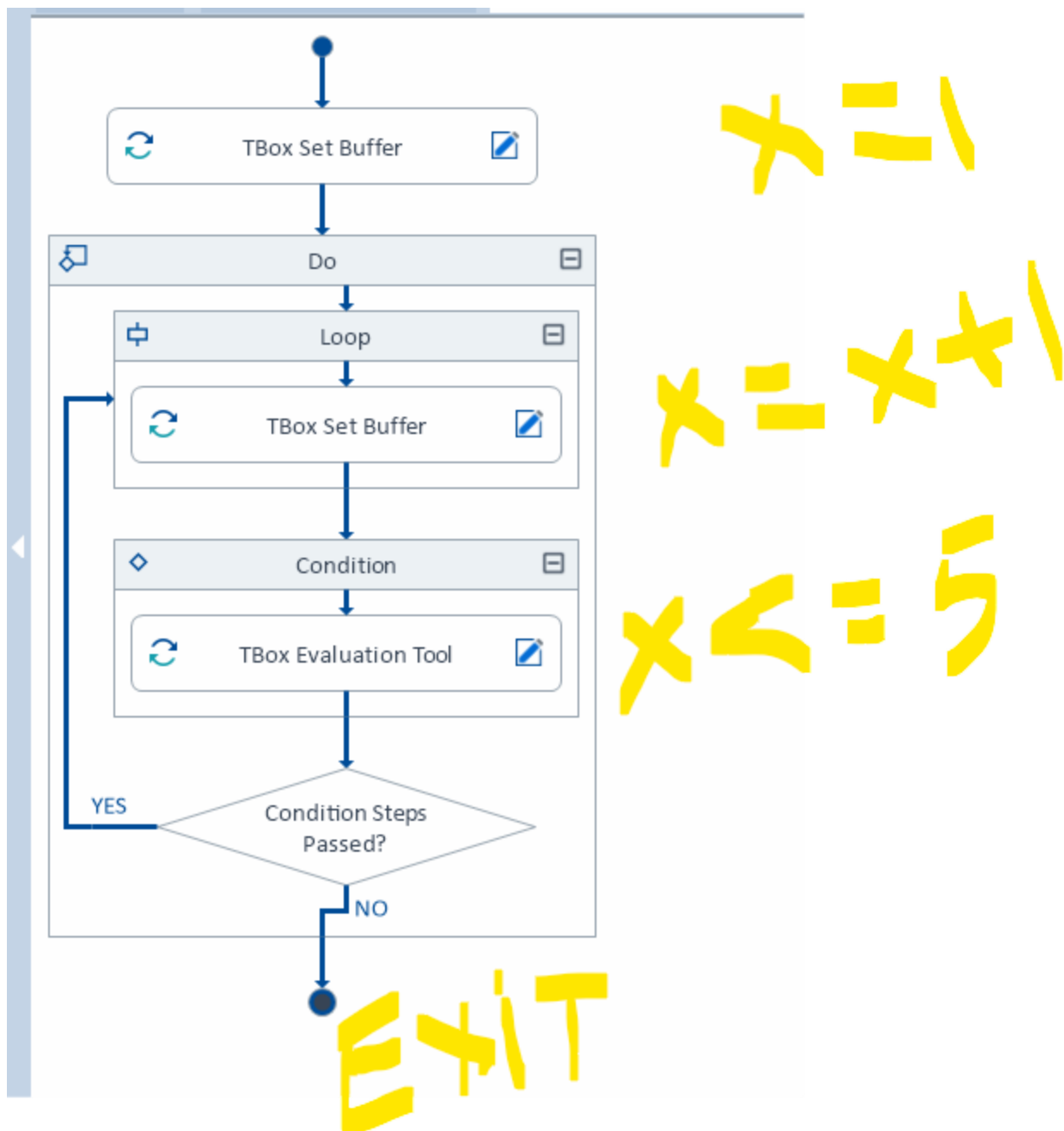
Add TBOX Set Buffer , and increase the value of the buffer Count by 1 each time it enters into the loop.

Count : $\{MATH[\{B[Count]\} + 1]\}$ (don't add any spaces in the math function)

The screenshot displays two panels from a software interface. The left panel, titled 'Test configuration', shows a hierarchical tree of test steps. The right panel, titled 'Properties', shows the configuration for the selected 'Do' step.

Name	Value	ActionMode	Data Type	WorkState
Do Loop to Print 1 to 20 numbers				
Print 1 to 10 numbers				PLANNED
TBox Set Buffer				
X	1	Input	Numeric	
Y	20	Input	String	
<Buffername>			String	
Do				
Loop				
TBox Set Buffer				
X	MATH B X +1	Input	Numeric	
<Buffername>			String	
Condition				
TBox Evaluation Tool				
Expression	B X <= B Y	Verify	String	
Expression			String	

Name	Value
Do	
NodePath	/TestCases/Do Loop to Print 1 to 20 numbers/Print 1 to 10 numbers/Do
CreatedBy	Admin
CreatedAt	6/2/2025 10:09:53 PM
ModifiedBy	Admin
ModifiedAt	6/2/2025 11:14:53 PM
HasMissingReferences	False
Uniqueld	3a1a45ba-3c84-1e96-0d23-a4000e0c2065
DisabledDescription	
IsPausable	False
MaximumRepetitions	{B[Y]}



Difference between Do Loop and While Loop?

Do Loop execute the statement under the loop and check for the condition.

While loop verifies the condition before executing the statement.

Tosca Commander: May-June-2025-Batch (C:\Tosca_Projects\Tosca_Workspaces\May-June-2025-Batch\May-June-2025-Batch.tws)

PROJECT HOME VIEW TOOLS TESTCASES API TESTING

Clipboard: Paste, Cut, Copy, Duplicate, Delete, Modify, Attach File, Edit. Create: Create Folder, Create Object.

TestCases: Run in ScratchBook, Jump to Module, Search and Add TestStep, Translate Value, Instantiate, Convert to Template, Link Data Source, Show TestStep Details, Show Business Param Values, Create Baseline.

Libraries: Libraries, Baseline.

Requirements May-June-2025-Batch TestCaseDesign Execution Issues Configurations TestCases x TestPlanning Modules

TestCases

- Standard module examples
- Virtual folders
- Obstacles Examples
- Self Healing
- Add Remove Controls
- Test 2
- <New Folder>
- Self Healing Enabled
- Demo WebLogin - Self Healing
- Data Driven Testing for Automobile Insurance
- Data Driven Testing with Excel File
- Excel Operations
- Enter Insurant data
- Enter Data for Select Price Option
- Send Quote
- Click OK on sending email success
- SQL DB Connection
- Database Connection
- Example for If Then Else Condition
- Buffer - Set, Name To, Partial, Delete, Iterate
- <New TestCase>
- While Loop
- Do Loop

Details Test configuration

Name	Value	ActionMode	DataType	WorkState
Do Loop				PLANNED
Print Random Numbers between 1 to 10				
TBox Set Buffer				
Count	1	Input	String	
<Buffername>			String	
Do				
Loop				
TBox Set Buffer				
Count	{MATH[{B[Count]}+1]}	Input	String	
<Buffername>			String	
Condition				
TBox Evaluation Tool				
Expression	B Count <10	Verify	String	
Expression			String	

Properties

Name	Value
Count	/TestCases/Do Loop/Print Random Numbers between 1 to 10/Do/Loop/TBox Set Buffer/Count
NodePath	
CreatedBy	Admin
CreatedAt	6/2/2025 5:55:11 PM
ModifiedBy	Admin
ModifiedAt	6/2/2025 5:55:42 PM
HasMissingReferences	False
Uniqueld	3a1a44d1-0c0d-3ac7-3b3c-7e8dbb81f62c
DataType	String
ValueToUse	{MATH[{B[Count]}+1]}
DisabledDescription	
Condition	
Path	

Ready 84°F Partly sunny Search 5:57 PM 6/2/2025 User: Admin

ScratchBook			
TBox Set Buffer			
Count			Buffer with name: "Count" has been set to value: "1"
Do			
Repetition: 1			
Loop			
TBox Set ...			
Count			Buffer with name: "Count" has been set to value: "2"
Condition			
TBox Eva...			
Expre...			'2<10' evaluated to 'True'
Repetition: 2			
Repetition: 3			
Repetition: 4			
Repetition: 5			
Repetition: 6			
Repetition: 7			
Repetition: 8			
Loop			

Name	Value	ActionMode	Loginfo
Condition			
TBox Eva...			
Expre...			'2<10' evaluated to 'True'
Repetition: 2			
Repetition: 3			
Repetition: 4			
Repetition: 5			
Repetition: 6			
Repetition: 7			
Repetition: 8			
Loop			
Condition			
TBox Eva...			
Expre...			'9<10' evaluated to 'True'
Repetition: 9			
Loop			
Condition			
TBox Eva...			
Expre...			'10<10' evaluated to 'False'

WAIT TIMES

WaitOn , Wait Mechanism, TBOX wait
synchronizationTimeout and synchronizationTimeoutDuringWaitOn

Dynamic vs Static wait

TBOX Wait : Static Wait , Wait until the specified time.

WaitBefore, WaitAfter : Static wait , Steering Parameters on Module Attribute level / module level
WaitOn : Dynamic Wait, select this option from ActionMode dropdown box , wait for control to acquire a particular state / value

synchronizationTimeout and synchronizationTimeoutDuringWaitOn

SynchronizationTimeout	Specify how long (in milliseconds) Tricentis Tosca should wait for the test object to reach its expected state. For detailed information, see chapter "Settings - Synchronization" .	<time in ms> 1000
SynchronizationTimeoutDuringWaitOn	Specify how long (in milliseconds) Tricentis Tosca should wait for the test object to reach its expected state. This setting is specifically for TestSteps with ActionMode WaitOn . For detailed information, see chapter "Settings - Synchronization" .	<time in ms> 1000

WaitOn - Dynamic Time out , Default settings on TBOx - Synchronization Wait on 20000 milliseconds. By default, Tosca waits 20 seconds for all **WaitOn** TestSteps. If you need more or less time, change the setting [Synchronization Timeout during WaitOn](#).

To add a dynamic wait time, use [ActionMode WaitOn](#) on the step that Tosca should wait on.

Details			
Control Flow Diagram			
	Name	Value	ActionMode
4 ↻	Click Next		
OK	Back		Input
OK	Send quote		Input
?	Progress bar for send quote	InnerText == 100%	WaitOn
OK	Next	Click	Input

Example:

Home page for obstacle:

<https://obstaclecourse.tricentis.com/Obstacles/List>

Example for WaitOn

<https://obstaclecourse.tricentis.com/Obstacles/33678>

Click on Calculate

Wait until the Send button is enabled

Click on Send button

Details	Test configuration				
	Name	Value	ActionMode	DataType	WorkState
Obstacle					
33678 Obstacle: Wait On					COMPLETED
Click on Calculate					
Calculate		X	Input	String	
Send		Enabled == True	WaitOn	String	
Click Send button					
Calculate				String	
Send		X	Input	String	

WaitBefore : Steering Parameter Module attribute level

WaitAfter : Steering Parameter

https://documentation.tricentis.com/tosca/2320/en/content/best_practices/testcases_wait_times.htm

WaitAfter	This parameter is created on the ModuleAttribute level. It specifies the time that is waited after the control was steered. The time is specified in milliseconds.	Default value = 0
WaitBefore	This parameter is created on the ModuleAttribute level and it specifies the time that is waited before a control is steered. The time is specified in milliseconds.	Default value = 0
WaitIntermediate (JavaFX Engine only)	This parameter can be defined for the JavaFX Engine 3.0. The parameter is created on the ModuleAttribute level and it defines wait times for scrolling in tables. The wait time is specified in milliseconds.	Default value = 130

3. synchronization error, increase sync time in settings.

Adding wait time to each control

https://documentation.tricentis.com/tosca/2320/en/content/best_practices/testcases_wait_times.htm

Obstacles / Select Random Text and select item from Combobox

Dynamic Text selection from Combo box / dropdown box.

<https://obstaclecourse.tricentis.com/Obstacles/73588#>

Test case steps

1. Click on Generate Random Text
2. Copy the / Get the text from the input box
3. Select the randomly generated copied text from the combo box
4. Click on Submit button

Obstacle73588 Dynamic Text Combobox				PLANNED
Generate Random Text and select the item ...				
GENERATE RANDOM TEXT	X	Input	String	
randomtext	RndTxt	Buffer	String	
selectlink	B RndTxt	Input	String	
SUBMIT	X	Input	String	

obstaclecourse.tricentis.com/Obstacles/73588#

s Automation Extension for Chrome" started debugging this browser

Cancel

HOME OBSTACL

TEST AUTOMATION OB

Can you do this one?

Easy Generate random string and select it in the

GENERATE RANDOM TEXT

H5H5iF6B2H

H5H5iF6B2H

SUBMIT

Good job!

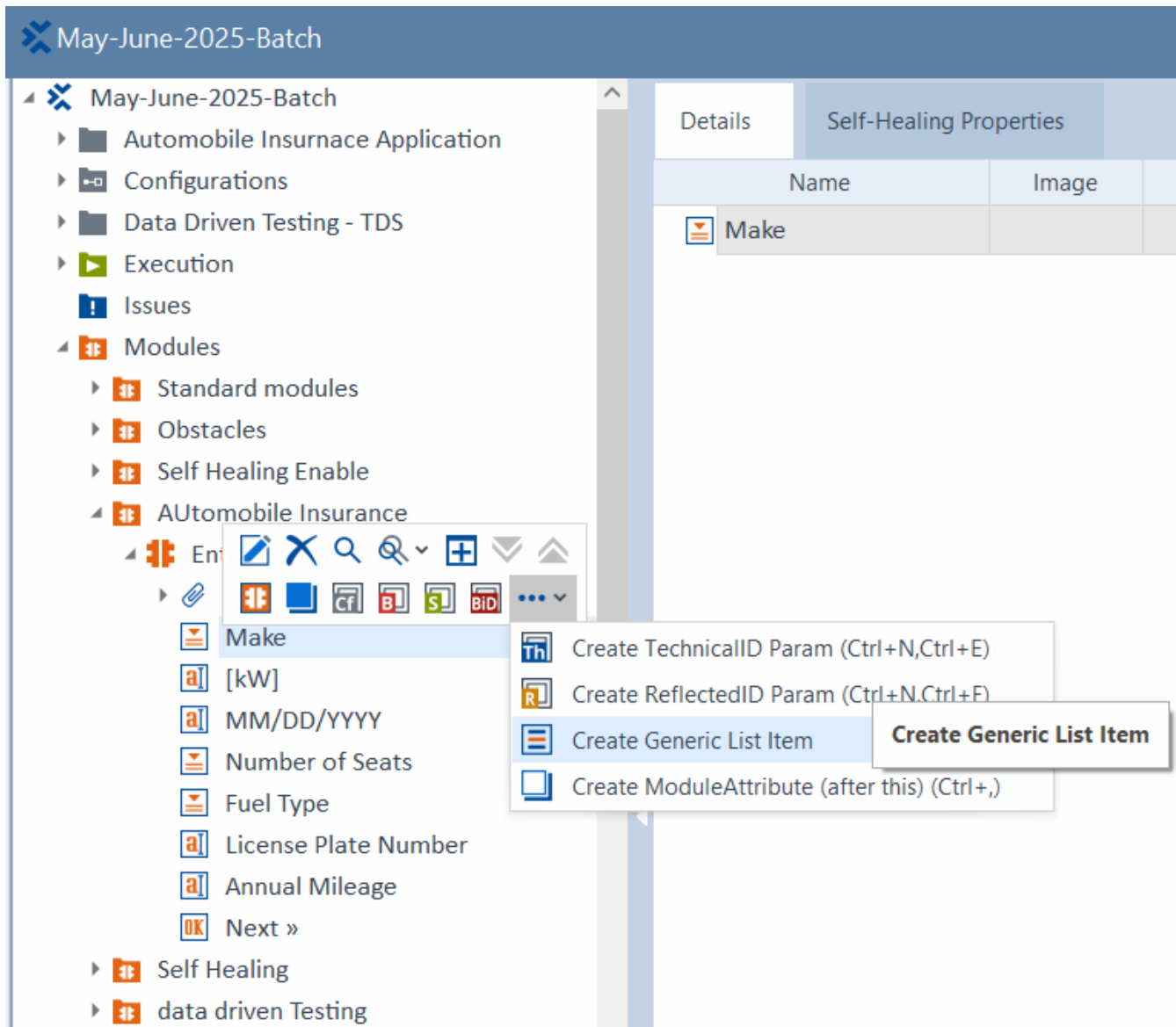
You solved this automation problem.

Let me try again

Hit me with the next riddle!

How to select Random item from Combo box / Dropdown box
How to use generic list item from combobox

Add Generic List Item to the Combo box from the module
How to use Control properties like ResultCount



To find the total number of items from the combo box, use the property called ResultCount
In Item , add the ResultCount and write into a buffer, to store the total number of items in combobox.
To select the random item, use RND function
#{RND[2][{B[TotalCount]}]}

Don't have to enter anything under the value column next to the RND function.

ComboBox-SelectRandomItem			
Select Random Item from Combobox			
Enter Vehicle Data			
Make		Select	String
<Item>	ResultCount → TotalCount	Buffer	String
#{RND[2][[B[TotalCount]]]}		Input	String
<Item>			String
[kW]			String
MM/DD/YYYY			String

Reference Properties for Row, column etc in table

https://documentation.tricentis.com/tosca/2420/en/content/tbox/type_table.htm?Highlight=ResultCount%20property

Property	Description	Available for
ColumnCount	Number of columns in the table.	table
RowCount	Number of rows in the table.	table
ColumnNumber	Index of the selected column. The value for a header column is 0 and the index for the remaining columns is relative to this column.	column
RowNumber	Index of the selected row. The value for a header row is 0 and the index for the remaining rows is relative to this row.	row
RawColumnNumber	Index of the selected column. It always starts at 1.	column
RawRowNumber	Index of the selected row. It always starts at 1.	row
ResultCount	Number of cells in a table which contains the content (constraint) you specify. Note the following: - This property ignores cells that you have configured as headers in your table. If you want to include matching headers as part of the result count, reconfigure the headers of your table accordingly. - The table control must use ActionMode Select. - The cell which contains the content you want to look for in the table must use ActionMode Constraint. - To limit the result count to the cells of a specific row or column, use ActionMode Verify in the row or column that contains the cell with the content you search for. Note that if you don't specify a value for <Col> or <Row>, Tricentis Tosca scans the entire table for the Value you specified in	row, column

Library , Reusable Test Step Blocks, Business Parameters

If a particular test step is required in multiple test cases, it does not need to be recreated each time. Instead, the test step can be defined once within a reusable test step block inside a library. This reusable block can then be referenced across any test cases where the step is needed, promoting consistency and reducing redundancy.

Identify repetitive test steps and create reusable teststep blocks to improve efficiency and reduce maintenance

https://documentation.tricentis.com/tosca/2420/en/content/tosca_commander/reusable_teststepblocks.htm

Example Login related steps required in many test cases.

1. Create a folder
2. Create a library
3. Create Reusable TestStep Library
4. Drag and drop / add the reusable steps from the TBOX or Module.
5. Add business parameters, drag and drop the business parameter name to the related field , it will create something like {PL[parameter name]}
6. Drag and drop this teststep block to the required test case and apply the test data as required and run the test case.
7. Run the test case
8. To disconnect the test case from the library, right click on the test case and Resolve Reference.

Recovery Scenarios / Error handling

https://documentation.tricentis.com/tosca/2420/en/content/tbox/recovery_create_scenario.htm?Highlight=recovery%20scenarios

A Recovery Scenario is a collection of TestSteps that Tricentis Tosca executes if particular tests fail.

1. Enable Recovery
 - a. Settings Apply to all objects in the workspace.
 - b. Configuration parameter - individual test case level
2. Recovery Scenario
 - a. Testcase folders
 - b. Test cases
3. Specify when the recovery scenario should be applied in the property **Retry Level**
 - a. TestStepValue : if test step value fails
 - b. TestStep : If test step fails
 - c. TestCase: If a test case fails
4. NOTE: **Run the test cases from Execution Tab only** . Drag and drop the test case / test case folder to the Execution List Under Execution Tab, and run from the Execution List. Then only e Recovery Scenarios will be executed. Run in Scratch Book will not execute the recovery scenarios.
5. **In Tosca, when a step fails within a recovery scenario, the default behavior is to retry the failed step after the recovery actions are completed.**

1. Settings change to Recover : apply to to the entire workspace

Tosca Settings [Role ADMIN] - C:\Program Files (x86)\TRICENTIS\Tosca Testsuite\Settings\XML\MetaSettings.xml

Settings View ?

Settings

- Commander
- Diagnostics
- License
- TBox
 - DokuSnapper
 - Dynamic temporal expressions
- Engines
 - Execution
 - Execution recorder
 - Extension loading
- Logging
 - Number formats
- OCR
 - Recorder
 - Recovery
- Simulation
 - Special Dates
- String Operations
- Synchronization
 - XScan
- TestCase from XL
 - Tosca Data Integrity
 - Tricentis Services

Name	Value	Comment
On dialog failure	Execute next TestCase	Choose what Tricentis Tosca should do in case of a dialog failure.
On exception failure	Execute next TestCase	Choose what Tricentis Tosca should do in case of an exception.
On verification failure	Continue	Choose what Tricentis Tosca should do in case of a verification failure.
TestCase retries	1	Maximum number of recovery attempts for a TestCase.
TestStep retries	2	Maximum number of recovery attempts for a TestStep.
TestStepValue retries	3	Maximum number of recovery attempts for a TestStepValue.

2nd option to apply Recovery scenarios for a specific folder or test case by adding configuration parameters.

Test configuration		
Name	Value	DataType
Recovery		
☐ OnDialogFailure	Recover	String
☐ OnExceptionFailure	Recover	String
☐ OnVerificationFailure	Recover	String
☐ TestCaseRetries	1	String
☐ TestStepRetries	2	String
☐ TestStepValueRetries	2	String

Test configuration			Properties	
Name	Value	ActionMode	Name	Value
RecoveryScenario			Recovery Scenario	
*** Recovery Scenarios ***			☐ NodePath /TestCases/R... ☐ CreatedBy Admin ☐ CreatedAt 6/9/2025 3:2... ☐ ModifiedBy Unknown ☐ ModifiedAt Unknown ☐ HasMissingReferences False ☐ Uniquelid 3a1a6855-95... ☐ DisabledDescription ☒ RetryLevel TestCase	
Recovery Scenario				

TestCases

Details

Test configuration

Name	
Login On Main menu	
Click on Login from Main Menu	
*** Recovery Scenarios ***	
Recovery Click on Login	
Logout_Reference	
Demo Web Shop Log out	
Log out	
Click on Login_Reference	
Login	
Log in	

Properties

Name	Value
Recovery Click on Login	/TestCases/Login On Main menu/Click on Login from Main Menu/*** Recovery Scenarios ***/Recovery Click on Login
NodePath	
CreatedBy	Admin
CreatedAt	6/9/2025 8:33:58 PM
ModifiedBy	Admin
ModifiedAt	6/9/2025 8:36:21 PM
HasMissingReferences	False
Uniqueld	3a1a696e-e5f5-f984-4329-a54a557f254f
DisabledDescription	
RetryLevel	TestCase

Execution

Execution

ExecutionLists

Exploratory Testing

Interactive Testing

Configurations

TestEvents

Details

Test configuration

Name	Number of A...	Description
Execution		
ExecutionLists		
Standard module examples		
Virtual folders		
<New ExecutionList>		
Login-CheckOut-Logout		
Click on Login from Main Menu		
Login		
Log in		
Recovery Click on Login		
Demo Web Shop Log out		
Login		
Exploratory Testing		
Interactive Testing		
Configurations		
TestEvents		

Best Practices :

https://documentation.tricentis.com/tosca/2420/en/content/best_practices/testcases_error_handling.htm?Highlight=recovery%20scenarios

Cleanup Scenarios

Check out process for <https://www.saucedemo.com/>

1. Open the url
2. Login
3. **Click on add to cart**
4. Checkout
5. Continue with the rest of the steps and logout , close the browser.

Success Scenario:

Add To cart is enabled and available to click

PreConditions: Login to <https://www.saucedemo.com/>

Execute below step

Success_Recovery_CleanUp_Scenarios				
Main Test Case Success_Scenario				PLANNED
Swag Labs Add To Cart				
Add to cart	x	Input	String	

Recovery Scenario :

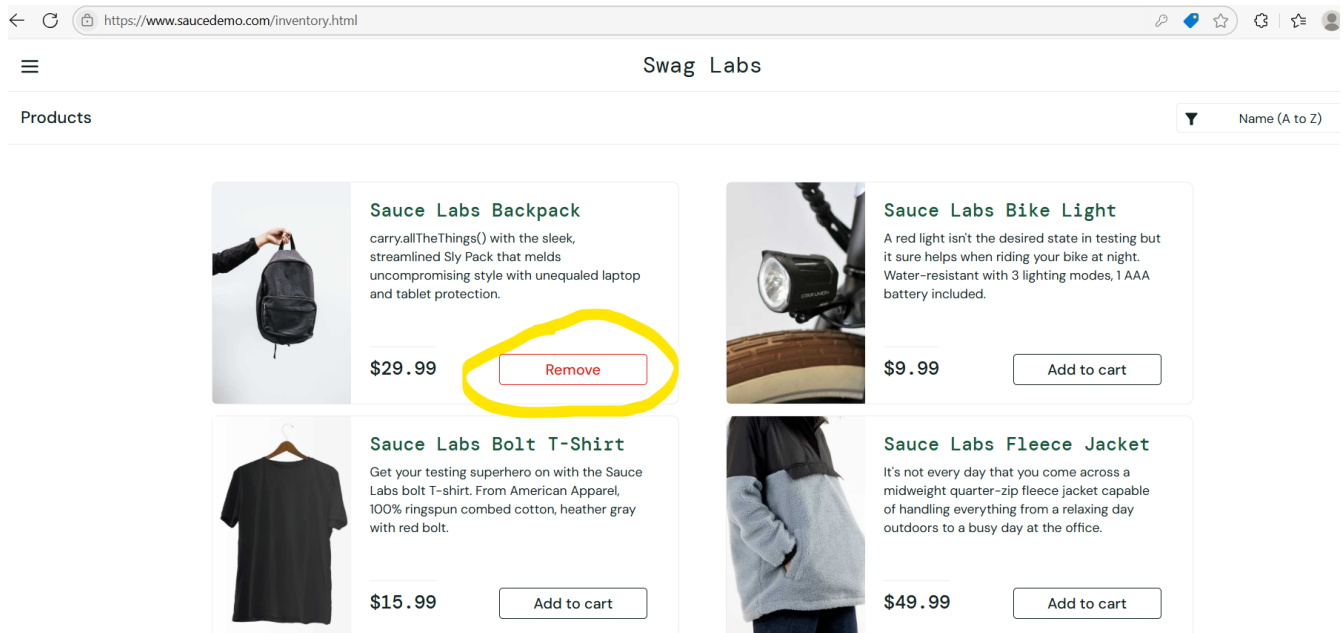
Add to cart screen is available before run recovery scenario :

Since Add to cart is disabled / Remove is enabled , the Remove button should be clicked.

Recovery Scenario				COMP... ▾
*** Recovery Scenarios ***				
Recovery Scenario				
Swag Labs Remove Button				
Remove	Enabled == True	Verify	String	
Swag Labs Remove Button				
Remove	x	Input	String	
Swag Labs Add To Cart				
Add to cart	x	Input	String	

CleanUp Scenario

Pre Condition to test Cleanup scenario : Remove button on inventory screen



Make the recovery scenario fail purposefully by verifying Enabled == False , so that cleanup scenario will execute.

CleanUp Scenario				PLANNED
*** Recovery Scenarios ***				
Recovery Scenario				
Swag Labs Remove Button				
Remove	Enabled == False	Verify	String	
CleanUp Scenario				
Swag Labs Menu				
Open Menu	x	Input	String	
All Items			String	
About			String	
Logout			String	
Reset App State	x	Input	String	
Close Menu			String	
Swag Labs Menu_1				
Open Menu			String	
All Items			String	
About			String	
Logout	x	Input	String	
Reset App State			String	
Close Menu			String	
CloseBrowser				
Title	Swag*	Input	String	
Swag Labs Add To Cart				
Add to cart	x	Input	String	

NOTE: Execute this test case from Execution List on Execution Tab

Test Results:

CleanUp Scenario			
Swag Labs Add To Cart			
Add to cart			
Recovery Scenario			
Swag Labs Remove Button			
Remove			
CleanUp Scenario			
Swag Labs Menu			
Swag Labs Menu_1			
CloseBrowser			

Purpose of Cleanup Scenarios

Environment Reset After Recovery Failures

When a Recovery Scenario fails, the test environment may be left in an unstable or inconsistent state. In such cases, a Cleanup Scenario is used to reset the environment, ensuring that the next test can execute under reliable conditions.

Illustrative Example

For instance, if a Recovery Scenario attempts to close and reopen a browser tab but leaves the session partially active, a Cleanup Scenario can:

- Log the user out,
- Close the browser,
- Reopen the browser,
- Log the user back in.

This comprehensive reset ensures a clean starting point for subsequent test executions.

How to Create a Cleanup Scenario in Tosca

1. Access the Recovery Scenarios Folder
Right-click the relevant *Recovery Scenarios* folder in Tosca.
2. Initiate Cleanup Scenario Creation
Select "Create Cleanup Scenario" from the mini toolbar.
3. Define Cleanup Steps
Add the required TestSteps to restore the test environment to a stable state. These steps should reverse any partial operations or environmental changes caused during the failed Recovery Scenario.

4. Configure Execution Conditions
Specify when Tosca should execute the Cleanup Scenario.
Typically, Tosca will trigger the Cleanup Scenario only if the Recovery Scenario fails to resolve the issue.
5. In Tosca, when a step fails within a recovery scenario, the default behavior is to retry the failed step after the recovery actions are completed.

Successful Scenario

Open URL, login, click on add to cart, logout, close the browser.

For make the test case simple : manually login to the <https://www.saucedemo.com/>

Success Scenario to click on Add To Cart

Success_Recovery_CleanUp_Scenarios				
Main Test Case Success_Scenario				PLANNED
Swag Labs Add To Cart				
Add to cart	x	Input	String	

Recovery Scenario : Add to cart is not available, verify Remove is enabled then click on Remove button.

Recovery Scenario				PLANNED
*** Recovery Scenarios ***				
Recovery Scenario				
Swag Labs Remove Button				
Remove	Enabled == True	Verify	String	
Swag Labs Add To Cart				
Add to cart	x	Input	String	

Cleanup Scenario : Make the recovery scenario fail purposefully by verifying Enabled == False , so that cleanup scenario will execute.

CleanUp Scenario					PLANNED
*** Recovery Scenarios ***					
Recovery Scenario					
Swag Labs Remove Button					
Remove	Enabled == False	Verify	String		
CleanUp Scenario					
Swag Labs Menu					
Open Menu			String		
All Items			String		
About			String		
Logout			String		
Reset App State	x	Input	String		
Close Menu			String		
Swag Labs Menu					
Open Menu			String		
All Items			String		
About			String		
Logout	x	Input	String		
Reset App State			String		
Close Menu			String		
CloseBrowser					
Title	Swag*	Input	String		
Swag Labs Add To Cart					
Add to cart	x	Input	String		

CleanUp Scenarios

https://documentation.tricentis.com/tosca/2420/en/content/tbox/recovery_cleanup.htm?Highlight=what%20is%20TestStepValues

Data-driven testing, How to Integrate Excel to the Test case?

Dynamic TestCase generation

Data Driven Testing: Testing the same functionality again and again with different sets of data.

1. Prepare test data in excel . Use the existing excel template. Located in C:\Tosca_Projects\ToscaCommander\Templates\BO
2. Example Excel file can be accessed from below URL

3. https://docs.google.com/spreadsheets/d/1q0-_F51rndqTHRBaWg1fHMBqyB0iL112/edit?usp=s_haring&oid=106800767575990603978&rtpof=true&sd=true
4. Create a test case in Tosca commander with the required module .
5. Convert test case to template
6. In Test case properties, Click on Schema path, add the Excel file and select the sheet name.
7. Map the fields from excel file to the required objects /control.
8. Example : {XL[Make]} {XL[Gendor.Female]}
9. Select the test case, right click, click on three dots, click on Create Template Instance.
10. Follow the steps and click on Ok to start the instantiation.
11. Template instances will be created.
12. Right click on the template and run in the scratch book to execute the test case.
13. If we add more records / test cases to the excel file, select the Template instance and right click and click on Reinstanaciate the Instance. Data will be refreshed in all the fields, or new test cases will be created if we add new records to excel file.

Test data can be fetched from excel file and generate dynamic test cases and apply the test data to the required fields.

1. Prepare test data from existing excel template located
C:\Tosca_Projects\ToscaCommander\Templates\BO
Microsoft® Excel templates can be found in the
%TRICENTIS_PROJECTS%\ToscaCommander\Templates\BO directory.

We can find the Excel settings from the Settings -> TestCase From XL -> TestSet

Tosca Settings [Role ADMIN] - C:\Program Files (x86)\TRICENTIS\Tosca Testsuite\Settings\XML\MetaSettings.xml

Settings View ?

Settings View ?

Name	Value	Comment
Domain column	5	Column containing attribute's domain name
First row containing data	2	First row containing data is typically first row below header
First TestCase column	6	First column containing TestCase data
Last structuring column	4	Last column forming Business Object and structure of its attributes
Row of TestCase title	1	Specifies the row in which the title of the TestCase can be found.
Treat empty cells as Null	Yes	Skip data transfer for empty cells. Empty cells get the same status as cells with gray background color.
Verification Backcolor	lightgreen	Specify background color to be used for verification of cells in Excel (see also documentation)

	A	B	C	D	E	F	G	H
1	Object/Attribute				Domain	TC_001	TestData_002	
2	Make					BMW	Ford	
3	Eng Perf					200	200	
4	Date Of Manf					02/22/2024	02/22/2024	
5	No Of Seats					6	6	
6	Fuel Type					Petrol	Petrol	
7	List Price					85000	85000	
8	License Plate No					H2k Info	H2k Info	
9	Annual Mil					12000	12000	
10								
11								
12								
13								

	A	B	C	D	E	F	G	H	I
1	Object/Attribute				Domain	TC_Object_001	TC_Object_002	TC_Object_003	TC_Object_004
2	Object					Object_001	Object_002	Object_002	Object_001
3		height				11	14	14	11
4		width				12	15	15	12
5		depth				13	16	16	13
6	Material					MadeOf_001	MadeOf_002	MadeOf_001	MadeOf_002
7		color				green1	red1	red1	green1
8		material				wood1	iron1	wood1	iron1

Object/Attribute										
	A	B	C	D	E	F	G	H	I	J
1	Object/Attribute				Domain	Valid Data 1	Valid Data 2			
2	Make				Vehicle Data	BMW	Ford			
3	Engine Per					333	555			
4	Date Of Manf					02/05/2025	02/05/2024			
5	No Of seats					6	6			
6	Fuel Type					Gas	Petrol			
7	List Price					55444	55444			
8	Licen Plate No					H2k Info	H2k Info			
9	Annula Mileage					12000	12000			
10										
11	First name				Insurant Data					
12	Last name									
13	DOB									
14	Gendor									
15		Male								
16		Female								
17	Hobbies									
18		Bugee Jump								
19		Sky Dive								
20		Other								
21										
22										
23										
24										
25										
26										
27										
28										
29										
30										
31										
32										
33										
34										
35										
36										

Object/Attribute										
	A	B	C	D	E	F	G	H	I	J
1	Object/Attribute				Domain	Valid Data 1	Valid Data 2			
2	Make				Vehicle Data	BMW	Ford			
3	Engine Per					333	555			
4	Date Of Manf					02/05/2025	02/05/2024			
5	No Of seats					6	6			
6	Fuel Type					Gas	Petrol			
7	List Price					55444	55444			
8	Licen Plate No					H2k Info	H2k Info			
9	Annula Mileage					12000	12000			
10										
11	First name				Insurant Data					
12	Last name									
13	DOB									
14	Gendor									
15		Male								
16		Female								
17	Hobbies									
18		Bugee Jump								
19		Sky Dive								
20		Other								
21										
22										
23										
24										
25										
26										
27										
28										
29										
30										
31										
32										
33										
34										
35										
36										

How it's used in Tosca:

Object / Attribute:

As per the settings in TestCase From XL -> TestSet

Object / Attribute :

In the example Excel file used within Tricentis Tosca (specifically, the BO templates), the "Domain" field typically refers to the logical grouping or context within which a specific value or object belongs. It's a way to categorize or organize data for better management and testing within Tosca.

The "Domain" field is used to relate test objects to the appropriate context. For instance, a test case for logging into the website would be linked to the "Users" domain, while a test case for placing an order would be linked to the "Orders" domain. This allows Tosca to manage the data and execute tests correctly based on the context of the specific test case or test object.

https://documentation.tricentis.com/tosca/2420/en/content/tosca_commander/test_data_from_excel.htm#:~:text=Test%20case%20data%20can%20be,are%20filled%20with%20data%20sets.

Mapping related document :

https://documentation.tricentis.com/tosca/2420/en/content/tosca_commander/combining_templates.htm?Highlight=%7BXL%7D

=====

Test Data Management TDM
Test Data Service TDS

tosca server

Download and Install Tricentis Tosca Server

Note: You must be logged in to the Tricentis Support Hub before accessing the download link. Register and login

1. Login to the Tricentis Support Hub

Use the following link to log in:

[Login - Tricentis Support](#)

2. Access the Download Page

After logging in, navigate to the download page directly via the link [Downloads - Tricentis Support](#)

3. For 2024.02 :

<https://support-hub.tricentis.com/open?id=downloads&filter=nameLIKETricentis%20Tosca%20Server%202024.2>

4.

5. Search for your desired Tosca Server version.

- For example, here is the link for Tricentis Tosca Server 2024.2

- [Downloads - Tricentis Support](#)

For example, here is the link for Tricentis Tosca Server 2025.

- <https://support-hub.tricentis.com/open?id=downloads&filter=nameLIKEserver>

<https://support-hub.tricentis.com/open?id=downloads&version=4bd1091293af4a544f1832847aba1063&sap=false&sel=5516ad36dbd7941035d3e2a94b96194a>

Tricentis Downloads

All your Tricentis downloads in one place

Search

My Recent Downloads

☐ SAP SolEx Customer Certified

Product Line

All Product Lines

Tricentis Device Cloud

Tricentis LiveCompare

Tricentis Mobile Agent

Tricentis NeoLoad

Tricentis qTest

Tricentis Sealights

Tricentis Testim

All Downloads

Tricentis Tosca Server 20

Product Line	Version	Patches	Product	Title	Build Date	Size in MBs
Tricentis Tosca 2024.2 LTS	2024.2	Tosca Server - 2024.2 - Tosca Server 202...	Tosca Server	Tricentis Tosca Server 2024.2.2 LTS.zip	2025-02-25	1337
Tricentis Tosca 2024.2 LTS	2024.2		Tosca Server	Tricentis Tosca Server 2024.2	2024-11-27	1338

Downloads 1 - 2 of 2

People who viewed this also viewed

Knowledge Base

Tricentis Forum

Tricentis Downloads

All your Tricentis downloads in one place

Search

My Recent Downloads

☐ SAP SolEx Customer Certified

Product Line

All Product Lines

Tricentis Device Cloud

Tricentis LiveCompare

Tricentis Mobile Agent

Tricentis NeoLoad

Tricentis qTest

All Downloads

Tricentis Tosca Server 20

Product Line	Version	Patches	Product	Title	Build Date	Size in MBs
Tricentis Tosca 2024.2 LTS	2024.2	Tosca Server - 2024.2 - Tosca Server 202...	Tosca Server	Tricentis Tosca Server 2024.2.2 LTS.zip	2025-02-25	1337
Tricentis Tosca 2024.2 LTS	2024.2		Tosca Server	Tricentis Tosca Server 2024.2	2024-11-27	1338

Downloads 1 - 2 of 2

People who viewed this also viewed

Knowledge Base

Tricentis Forum

Downloads

Tricentis Tosca Server 2024.2.2 LTS.zip

331 MB - 331 MB of 1.3 GB, 12 secs left

Tricentis Tosca Server 2024.2.2 LTS.zip

Removed

Tricentis Tosca 2024.2.1 LTS (1).zip

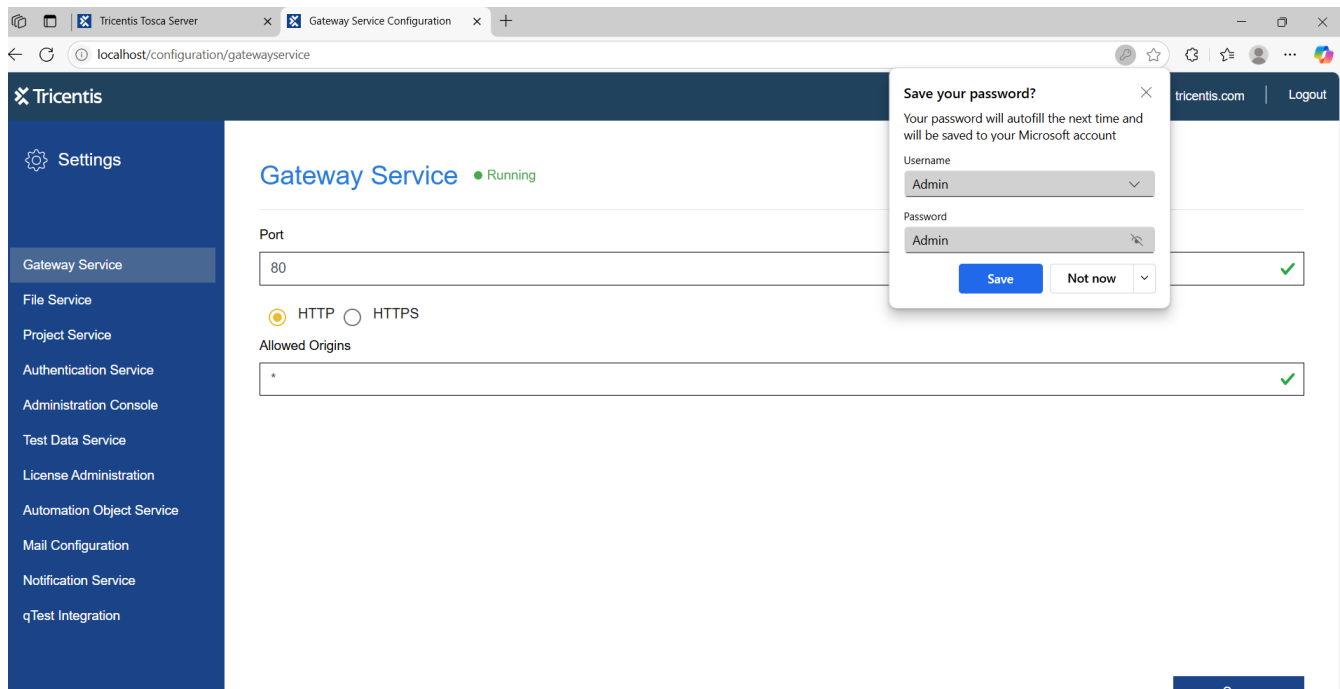
Removed

See more

Select the HTTP option button in the screen below. If 8080 port number is not available, try any other number

Default port for http : 80, https: 443

Default Logins for settings change after installation : Admin / Admin



After installation, launch the <http://localhost/> from browser
Click on Test Data Management

Follow below steps to install Tricentis Tosca Server
Double click on the exe file

Name	Date modified	Type	Size
Earlier this year			
Tricentis Tosca Server 2023.2.8 LTS	4/17/2025 8:44 PM	Application	1,156,593 ...
Tricentis Tosca Server 2023.2.8 LTS.exe.sha256	4/17/2025 8:44 PM	SHA256 File	1 KB

Tricentis Tosca Server Setup 2023.2.8

Prerequisites

Type

Review

Installation

Summary

The following prerequisites are required for Tricentis Tosca Server and will be installed during the setup if necessary. Checked items are already installed.

Prerequisites

✓	Microsoft .NET Framework 4.8	4.8.3928
✓	Microsoft .NET 8.0.12 - Windows Desktop Runtime (x64)	8.0.12
✓	Microsoft .NET 8.0.12 - Hosting bundle	8.0.12

2023.2.8.1105

Next

Tricentis Tosca Server Setup 2023.2.8

Prerequisites

Type

Review

Installation

Summary

The following will be installed:

Tricentis Tosca Server

Setup installer for Tricentis Tosca Server

Includes:

Gateway
License Administration
Authentication Service
Test Data Service
Automation Object Service
Distributed Execution
Product Info
File Service
Rest API Service
Notification Service
Interactive Testing Server
Project Service
Migration Service
Administration Console
Exploratory Testing Server
SAP Solution Manager API Service

Back

Next

Tricentis Tosca Server Setup 2023.2.8

Prerequisites

Type

Destination

Diagnostics

Service Configuration

File Service

Review

Installation

Summary

Install Location

Choose a destination folder. Please ensure you have proper file permissions.

Install Tricentis Tosca Server to:



C:\Program Files (x86)\TRICENTIS\Tosca Server

Browse

Set common application data path to:



C:\ProgramData\TRICENTIS\ToscaServer

Browse

Back

Next

×

Tricentis Tosca Server Setup 2023.2.8

Prerequisites

Type

Destination

Diagnostics

Service Configuration

File Service

Review

Installation

Summary

Tosca Server Diagnostics

Here is how it benefits you:

- Helps us build products that give you maximum benefit
- Enables us to better assist in problem analysis and resolution



Find our Data Policy at www.tricentis.com/legal-information

☐ Help improve Tricentis Tosca Server (TM) by sending anonymous usage statistics

Back

Next

Tricentis Tosca Server Setup 2023.2.8

Prerequisites

Type

Destination

Diagnostics

Service Configuration

File Service

Review

Installation

Summary

Tricentis Tosca Server shared Settings

☒ HTTP ☐ HTTPS

Port



The port is available.

Back

Next

Tricentis Tosca Server Setup 2023.2.8

Prerequisites

Type

Destination

Diagnostics

Service Configuration

File Service

Review

Installation

Summary

Install Location

Choose a destination folder. Please ensure you have proper file permissions.

Root directory to store files



C:\ProgramData\TRICENTIS\ToscaServer\FileService\File

Browse

Back

Next

Tricentis Tosca Server Setup 2023.2.8

Prerequisites

Type

Review

Installation

Summary

Ready to install

Summary

- ↓ Gateway
- ↓ License Administration
- ↓ Authentication Service
- ↓ Test Data Service
- ↓ Automation Object Service
- ↓ Distributed Execution
- ↓ Product Info
- ↓ File Service
- ↓ Rest API Service
- ↓ Notification Service
- ↓ Interactive Testing Server
- ↓ Project Service
- ↓ Migration Service
- ↓ Administration Console
- ↓ Exploratory Testing Server
- ↓ SAP Solution Manager API Service

Back

Install

Click YEs on windows popup

Tricentis Tosca Server Setup 2023.2.8

Prerequisites

Type

Review

Installation

Summary

Installing...

Preparing... 0%

Summary

Gateway

License Administration

Authentication Service

Test Data Service

Automation Object Service

Distributed Execution

Product Info

File Service

Rest API Service

Notification Service

Interactive Testing Server

Project Service

Migration Service

Cancel

Tricentis Tosca Server Setup 2023.2.8

Prerequisites

Type

Review

Installation

Summary

Install completed.

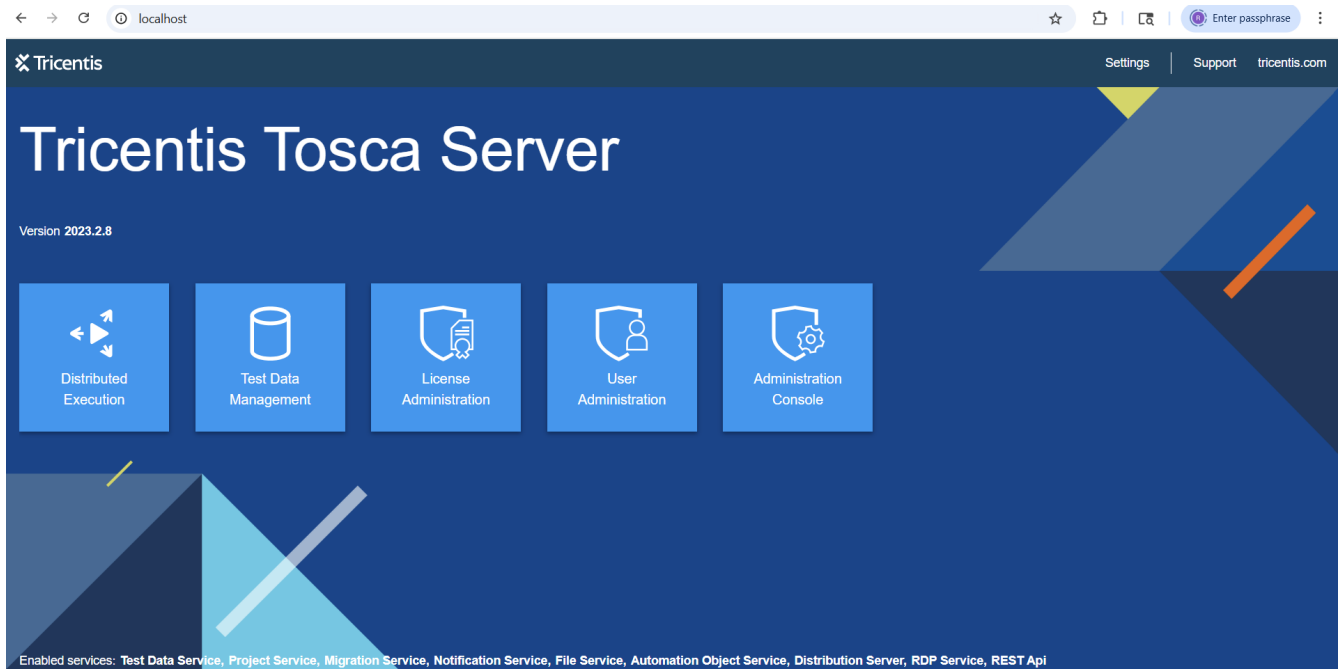
Summary

- ✓ Gateway
- ✓ License Administration
- ✓ Authentication Service
- ✓ Test Data Service
- ✓ Automation Object Service
- ✓ Distributed Execution
- ✓ Product Info
- ✓ File Service
- ✓ Rest API Service
- ✓ Notification Service
- ✓ Interactive Testing Server
- ✓ Project Service

Show Logs

- ☒ Open Tricentis Tosca Server in your browser
- ☒ Open Tricentis Tosca Server settings page in your browser

Finish

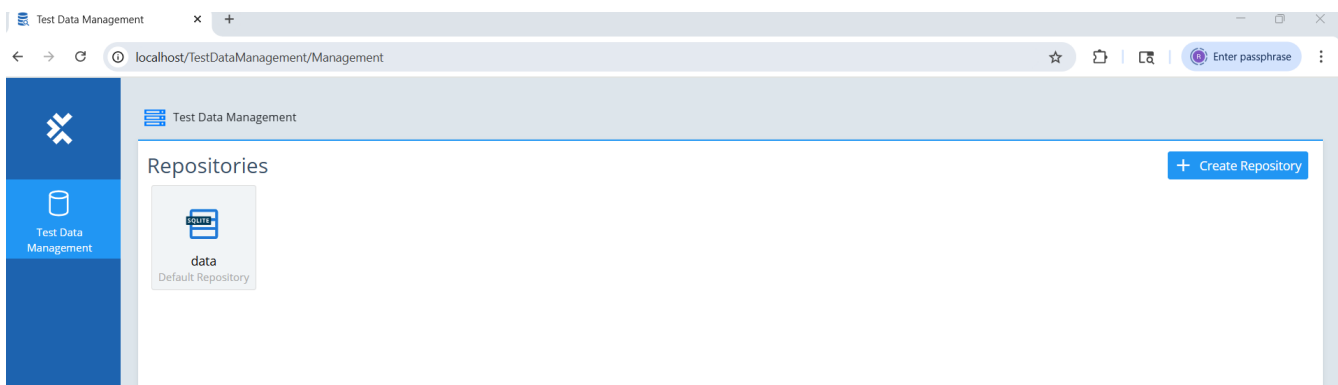


If we need to update the settings, click n settings, login : Admin / admin

<http://localhost/TestDataManagement/Management>

NOTE: Port number can be 8080

Click Create Repository and name it: AutomobileInsurance



Create Test Case

Add below test configuration parameters

TestDataEndpoint by default <http://localhost/testdataservice/>

OR depends on the port number

OR

<http://localhost:8080/testdataservice/>

<http://localhost:5001/testdataservice/>

TestDataRepository AutomobileInsurance

Working with Excel files using Open Excel Workbook, Define, range manipulation, close etc. Data Driven Testing with excel file

Task : Read data from existing excel file, specific range of values , write the value to a buffer variable, apply those values to the Vehicle Information page.

1. In your local drive, Create test data in excel file and save it , Example
AutomobileVehicleInsurance_AllPages_TestData.xlsx
2. In Tosca Commander : Create a new folder and add a test case.
3. Add step **TBox Open Excel Workbook** (from standard module / TBOX), add workbook name (any name) add the location of the file including file name
C:\Users\raoh2\OneDrive\Desktop\H2kInfosys\Tosca\AutomobileVehicleInsurance_AllPages_TestData.xlsx

Create New : False

4. Add **TBox Define Excel Range** , add workbook name (same name from above step), workbook name : VehicleData (from the excel file name), Range Name: Any name (VehicleTestDataCellRange), Data Range : Start Cell A2, End Cell H3 as per the excel file data that we want to use.
5. Add **TBox Excel Range Manipulation** from TBOX, type the Range Name as per the above step , VehicleTestDataCellRange ,
Under Data Table , click on Row and Select \$lastContentRow , in Value column, click on the blue down arrow button and add the property as RowNumber and variable name as TotalRows, Under action mode, select Buffer. Run the test case and verify the total number of records in excel file is matching with the value in TotalRows buffer variable

TBox Excel Range Manipu...				
	Range Name	VehicleTestDataCellRange	Input	String
	Data Table		Select	String
	\$lastContentRow	.RowNumber->TotalRows	Buffer	String
	<Cell>			String
	<Row>			String
	<Cell>			String
	<Col>			String

6. Select the **test case**, right click and click on Create folder, rename the folder to RepeatStep. On Create folder properties, click on Repetition property and add {B[TotalRows]} (This is how we can read the value from the buffer variable TotalRows, that was created in the above step.

Move this folder to under the step TBox Excel Range Manipulation.

Copy the step TBox Excel Range Manipulation, select the folder “RepeatStep” and paste.

Expand the folder, delete the first row which is \$lastContentRow RowNumber → TotalRows
Add a math function with the property “Repetition” which contains the number . if the excel file contains 10 rows, it start from 1, 2, 3.. 10

#{MATH[{Repetition}+1]}

Replace the cell with #1, #2 #3 etc.. for all the columns and value for each column can be something like Make, EngPerf, DOM, etc.. and select action mode as buffer so that the values will be stored in these buffer variables.

7. Select the folder “Repetition” , Add Step “ Enter Vehicle Data “ (NOTE : vehicle data related screen should be scanned and saved in Modules before we plan to add this step.

Add the buffer variable name to each column. Example for Make , value is {B[Make]}

8. Add “TBox Close Excel Workbook” , add the workbook name and path as mentioned in first step TBox Open Excel Workbook. Select the Save as False since we are not saving any data in excel.

All the above steps are covered in the below screenshot.

Details	Test configuration					
	Name	Value	ActionMode	DataType	WorkState	
▲	Data Driven Testing with Excel Fi...					
▲	Enter Vehicle Information Fr...				PLANNED	
▲	TBox Open Excel Workbook					
■	Workbook Name	TestData	Input	String		
■	Path	C:\Users\raoh2\OneDrive\Desktop\H2kInfosys\Tosca\AutomobileVehicleInsurance_AllPages_TestData.XLSX	Input	String		
■	Create New	False	Input	String		
■	Password			Password		
■	Open as Read-only			Boolean		
▲	TBox Define Excel Range					
■	Workbook Name	TestData	Input	String		
■	Worksheet Name	VehicleData	Input	String		
■	Range Name	VehicleTestDataCellRange	Input	String		
▲	Data Range		Select	String		
■	Start Cell	A1	Input	String		
■	End Cell	H3	Input	String		
▲	TBox Excel Range Manipu...					
■	Range Name	VehicleTestDataCellRange	Input	String		
▲	Data Table		Select	String		
▲	\$_lastContentRow	RowNumber → TotalRows	Buffer	String		

Details						Properties		
Test configuration								
Name	Value	ActionMode	DataType	WorkState		Name	Value	
TBox Excel Range Manipu... Range Name VehicleTestDataCellRange Input String						RepeateStep /TestCases/Dat a Driven Testing with Excel File/Enter Vehicle Information From Excel/Repeate Step		
Data Table SlastContentRow RowNumber → TotalRows Buffer String						NodePath		
<Cell> String						CreatedBy Admin		
<Row> String						CreatedAt 5/28/2025 8:12:30 PM		
<Col> String						ModifiedBy Admin		
RepeateStep						ModifiedAt 5/28/2025 8:38:40 PM		
TBox Excel Range Man... Range Name VehicleTestDataCellRange Input String						HasMissingRefere... False		
Data Table #[MATH]([Repetit... Select String						Uniqueld 3a1a2b8e-f933 -4fa2-391f-5a1 a7e98b096		
#1 Make Buffer String						Repetition [B[TotalRows]]		
#2 EngPerf Buffer String								
<Cell> String								
<Row> String								
<Col> String								
Enter Vehicle Data Make B Make Input String								
[kW] B EngPerf Input String								
MM/DD/YYYY String								
Number of Seats String								
Fuel Type String								
License Plate Number String								
Annual Mileage String								
Next » String								
TBox Close Excel Workbo... Workbook Name TestData Input String								
Path C:\Users\raoh2\OneDrive\Desktop\H2kInfosys\Tosca\AutomobileVehicleInsurance_AllPages_TestData.XLSX Input String								
Save False Input String								
Save Path String								
Save As Type String								

Details						Properties		
Test configuration								
Name	Value	ActionMode	DataType	WorkState		Name	Value	
<Cell> String								
<Row> String								
<Col> String								
Enter Vehicle Data Make B Make Input String								
[kW] B EngPerf Input String								
MM/DD/YYYY String								
Number of Seats String								
Fuel Type String								
License Plate Number String								
Annual Mileage String								
Next » String								
TBox Close Excel Workbo... Workbook Name TestData Input String								
Path C:\Users\raoh2\OneDrive\Desktop\H2kInfosys\Tosca\AutomobileVehicleInsurance_AllPages_TestData.XLSX Input String								
Save False Input String								
Save Path String								
Save As Type String								

Install SQL Server Database and SQL Server Management Studio (SSMS)

Download SQL Server database

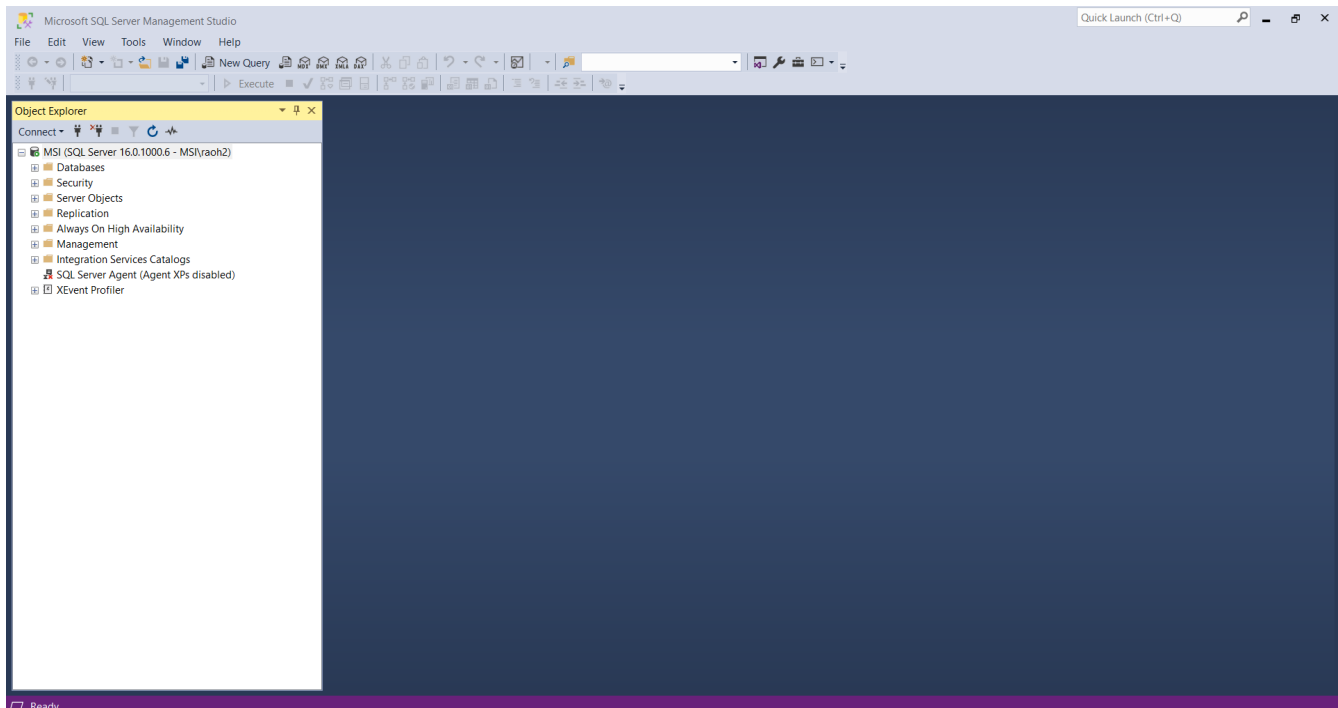
<https://www.microsoft.com/en-us/sql-server/sql-server-downloads>

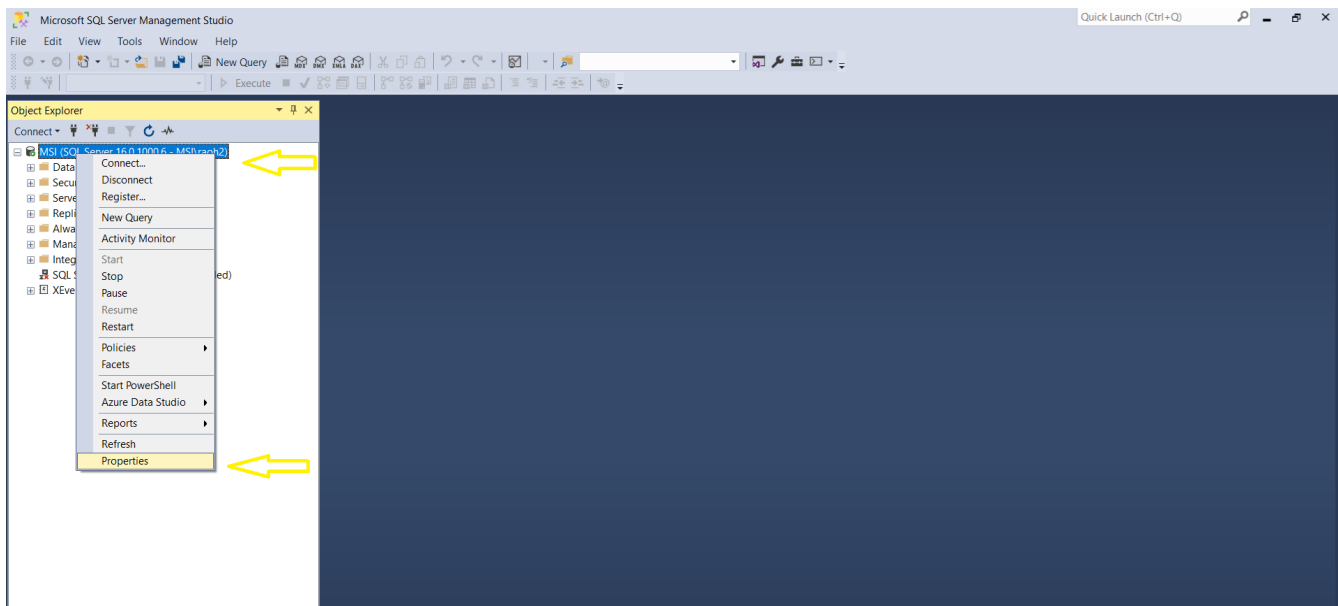
After downloading the file, click on the executable file example: SQL2022-SSEI-Dev . Select Basic and follow the instructions and install.

Install SQL Server database. And at the end of the installation, it will prompt you to install SQL Server Management Studio, Click on Install SSMS and follow the steps to install .

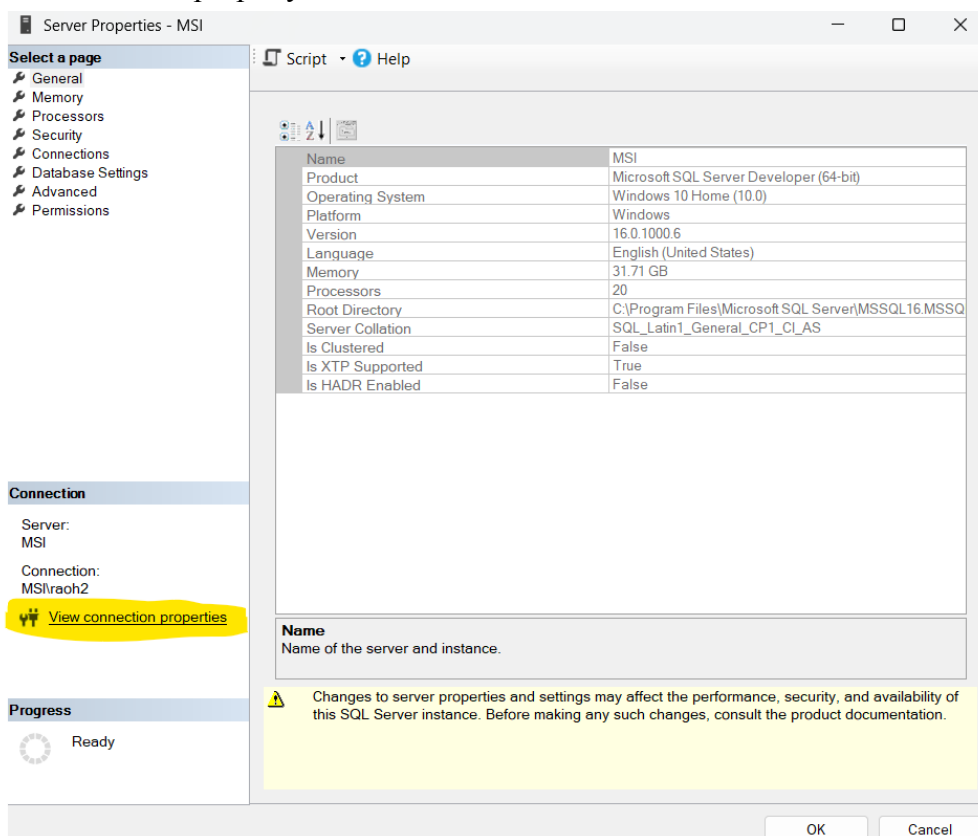
From above installation steps, SQL server name , usually your computer name , example MSI , dont enter any password.

Once you install SSMS, search for SQL Server management studio from your windows taskbar and launch SSMS .

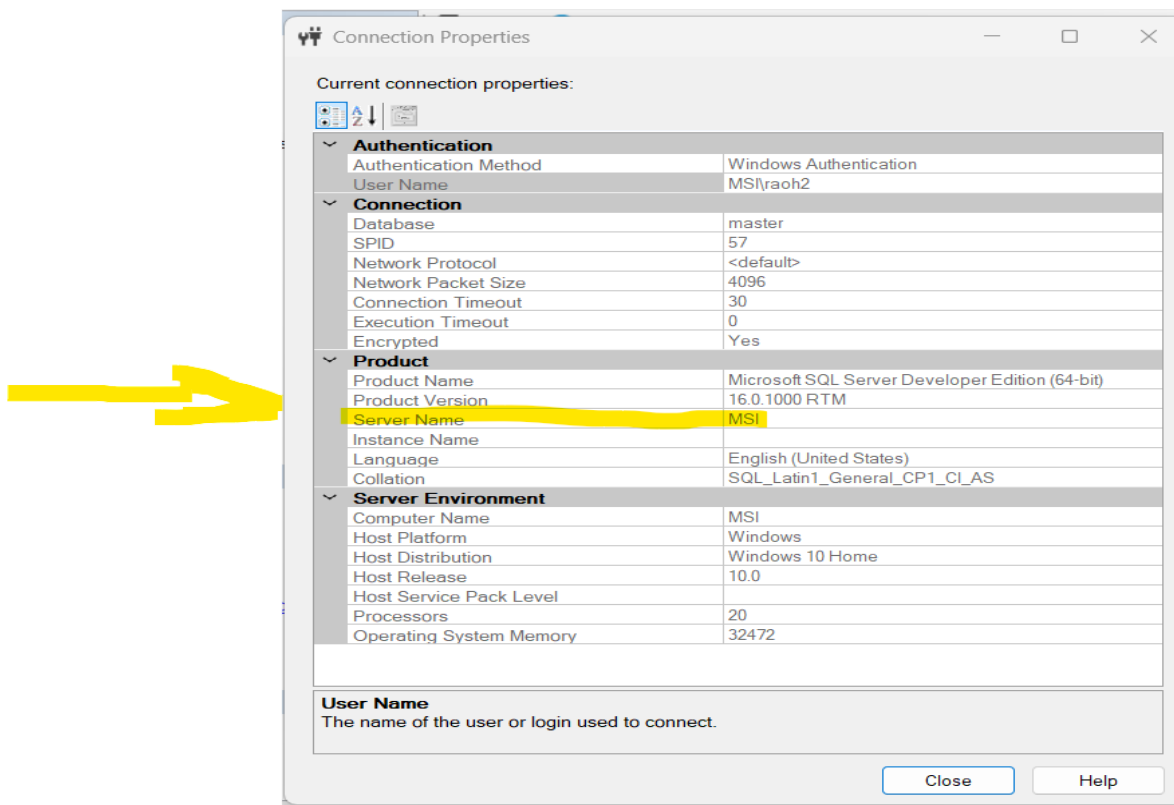




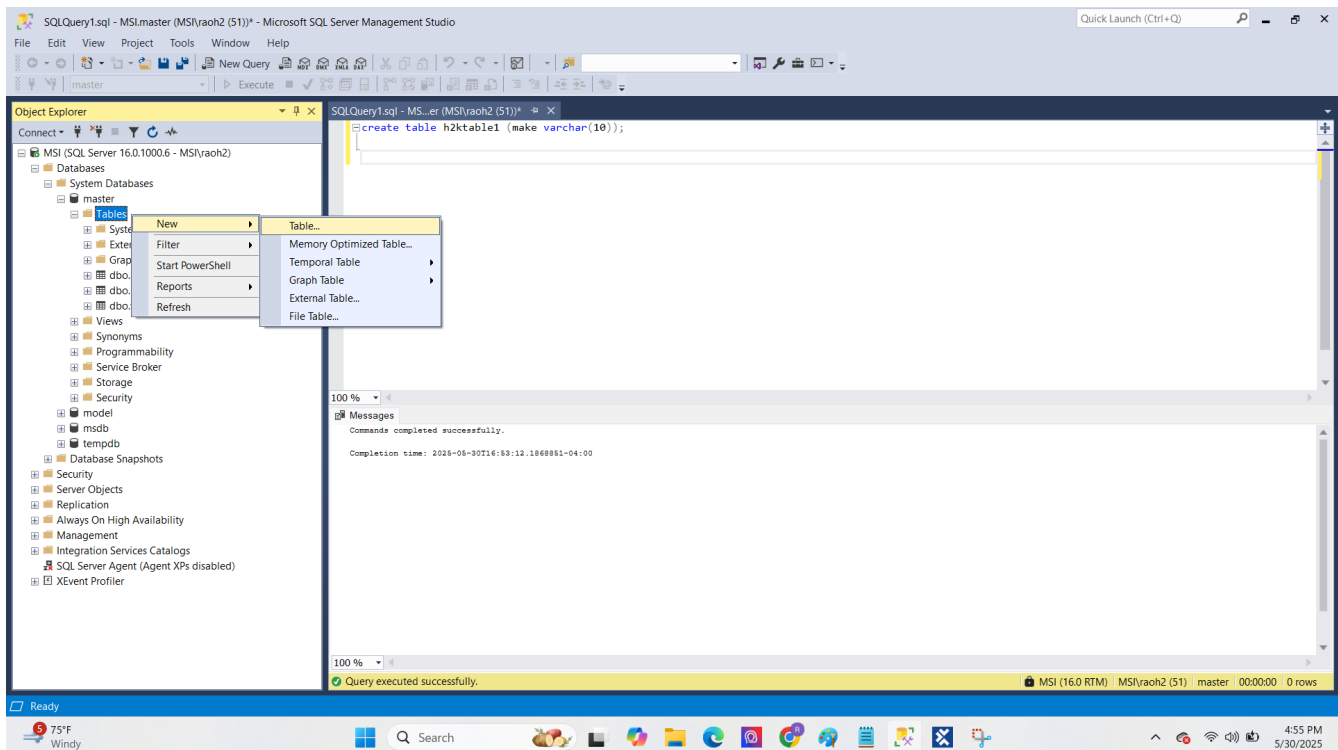
Click on View connection property



Find the server name from below screen



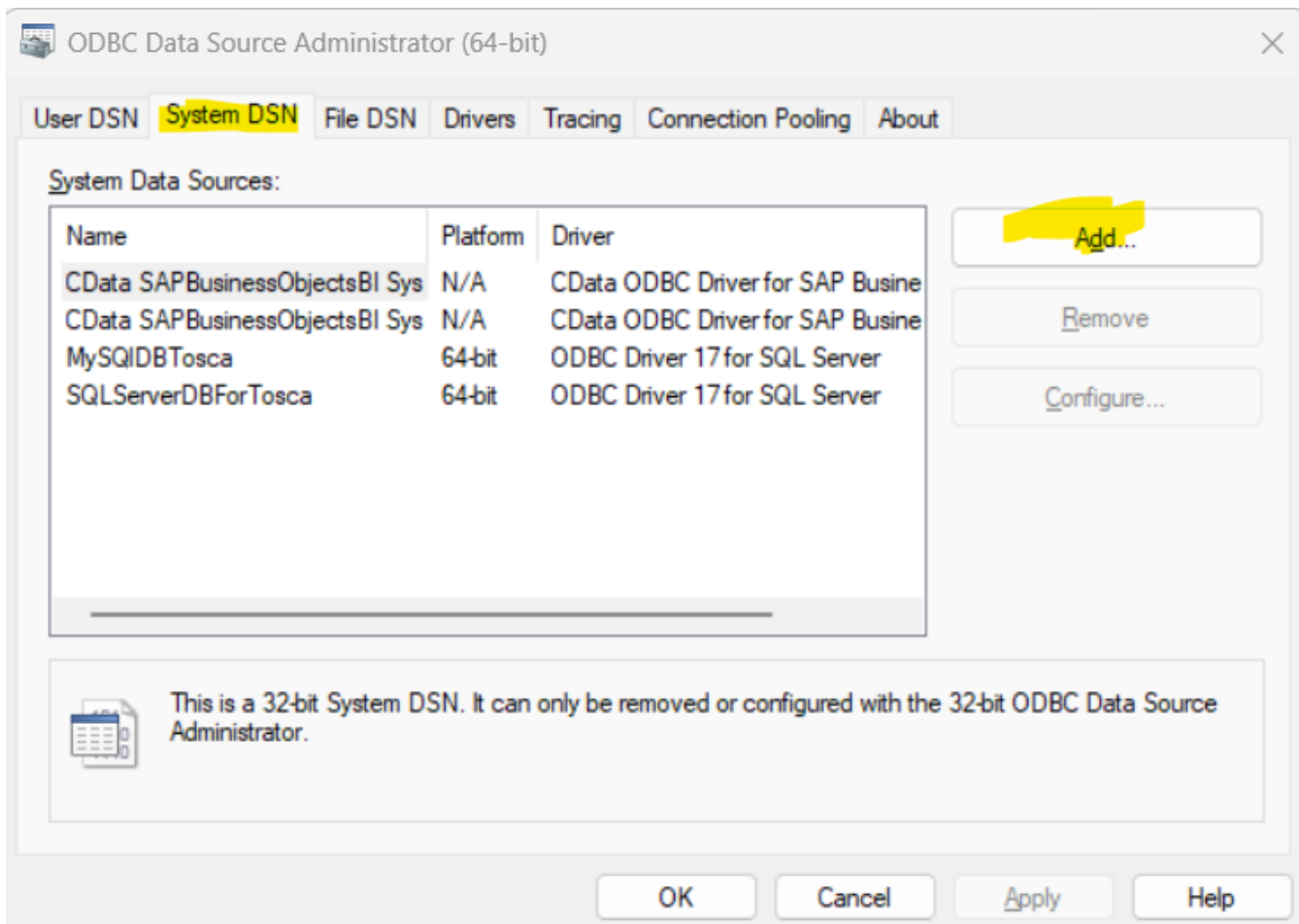
Create a new table VehicleInfo

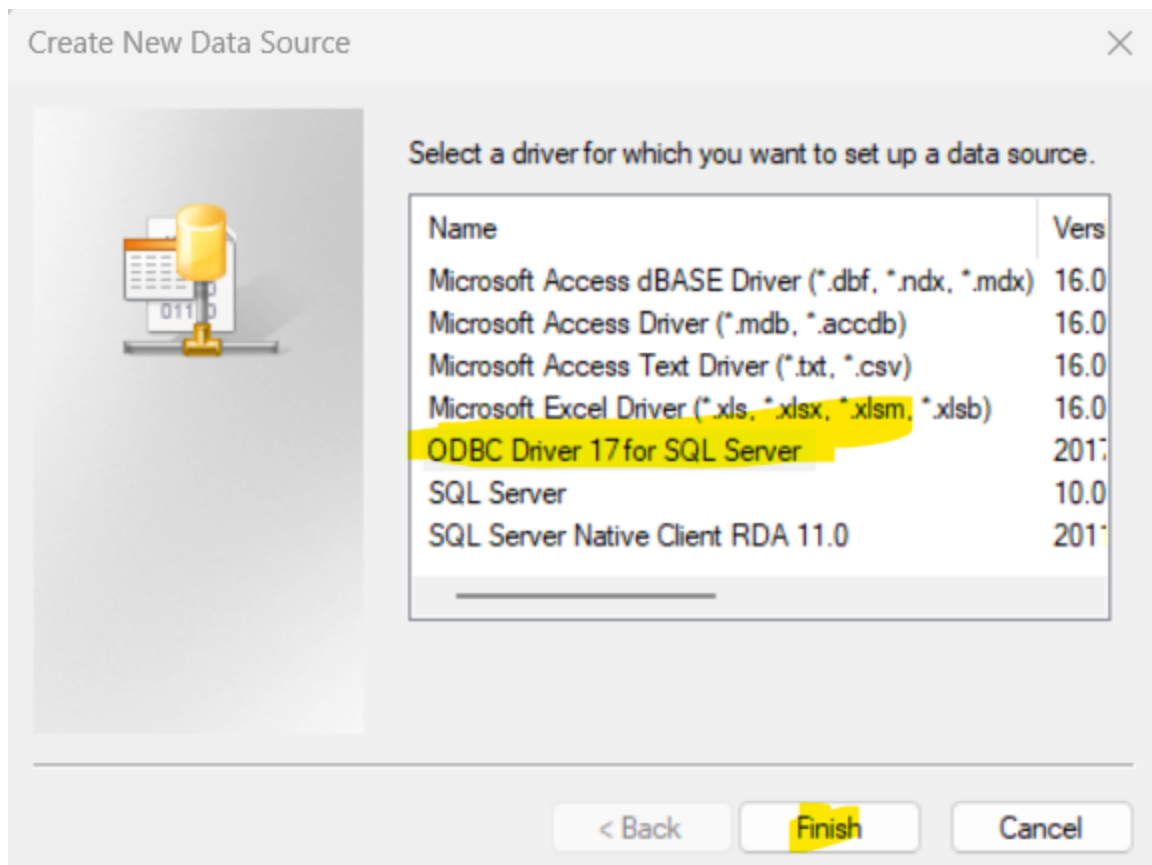


How to create DSN using ODBC Driver for SQL Server

Step 1: Create DSN from ODBC Drivers

To connect Tosca to a SQL Server database, an ODBC driver connection must be created via the ODBC Data Source (64-bit). This can be accessed by searching for "ODBC Database source" in the Windows taskbar. In the ODBC Data Source window, navigate to the "System DSN" tab and click "Add". Select "ODBC Driver 17 for SQL Server" in the "Create New Data Source" window and click "Finish". In the subsequent "Create New Data Source to SQL Server" window, specify a name, a description for the connection, and the name of the SQL Server that will be used.





Server name can be found from your SQL Server Management Studio by clicking on settings .

Create a New Data Source to SQL Server

This wizard will help you create an ODBC data source that you can use to connect to SQL Server.

What name do you want to use to refer to the data source?

Name:

How do you want to describe the data source?

Description:

Which SQL Server do you want to connect to?

Server:

Finish Next > Cancel Help

Click Next on this below screen

Create a New Data Source to SQL Server



How should SQL Server verify the authenticity of the login ID?

- ☒ With Integrated Windows authentication.
SPN (Optional):
- ☐ With Azure Active Directory Integrated authentication.
- ☐ With SQL Server authentication using a login ID and password entered by the user.
- ☐ With Azure Active Directory Password authentication using a login ID and password entered by the user.
- ☐ With Azure Active Directory Interactive authentication using a login ID entered by the user.
- ☐ With Azure Managed Service Identity authentication.
- ☐ With Azure Service Principal authentication.

Login ID:

Password:

< Back Next > Cancel Help

Click Next

Create a New Data Source to SQL Server



☐ Change the default database to:

(Default) ▾

Mirror server:

SPN for mirror server (Optional):

☐ Attach database filename:

☒ Use ANSI quoted identifiers.

☒ Use ANSI nulls, paddings and warnings.

Application intent:

READWRITE ▾

☐ Multi-subnet failover.

☒ Transparent Network IP Resolution.

☐ Column Encryption.

Enclave Attestation Info:

Keystore Configuration...

☐ Use FMONLY metadata discovery.

< Back Next > Cancel Help

Click Finish

Create a New Data Source to SQL Server ✕



☐ Change the language of SQL Server system messages to:
(Default) ▼

☐ Use strong encryption for data.

☐ Trust server certificate.

☒ Perform translation for character data.

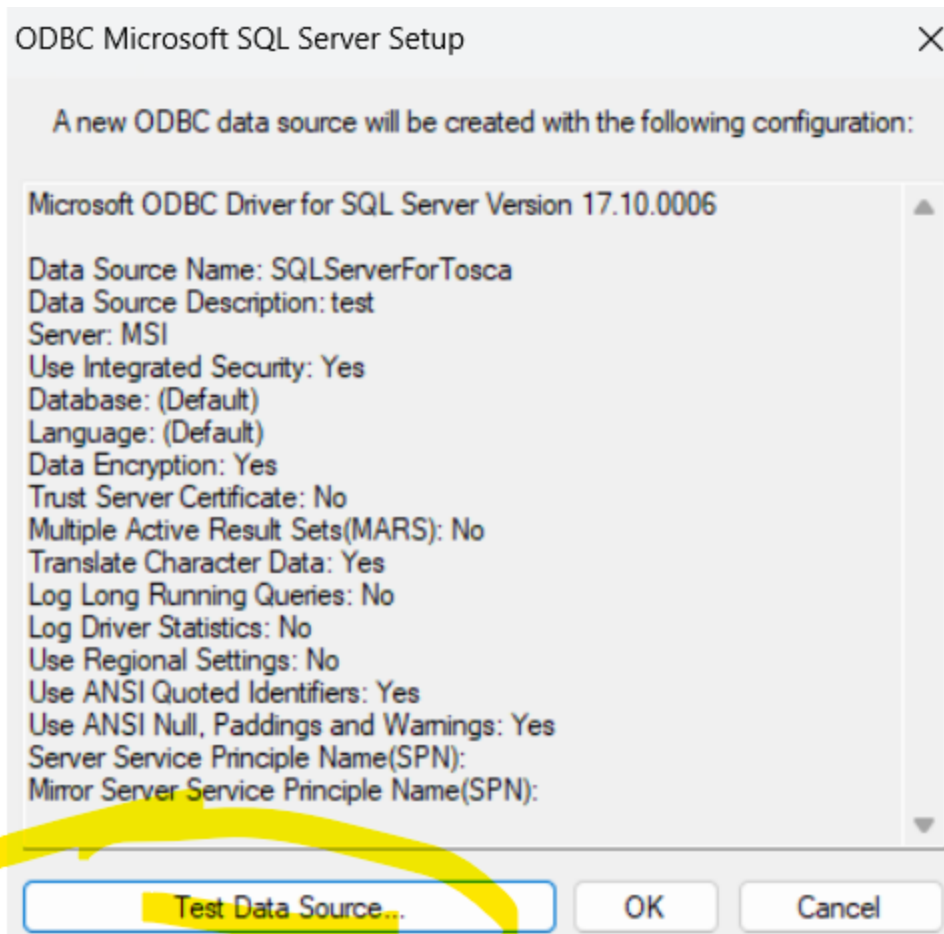
☐ Use regional settings when outputting currency, numbers, dates and times.

☐ Save long running queries to the log file:
C:\Users\raoh2\AppData\Local\Temp\QUERY.LOG Browse...
Long query time (milliseconds): 30000

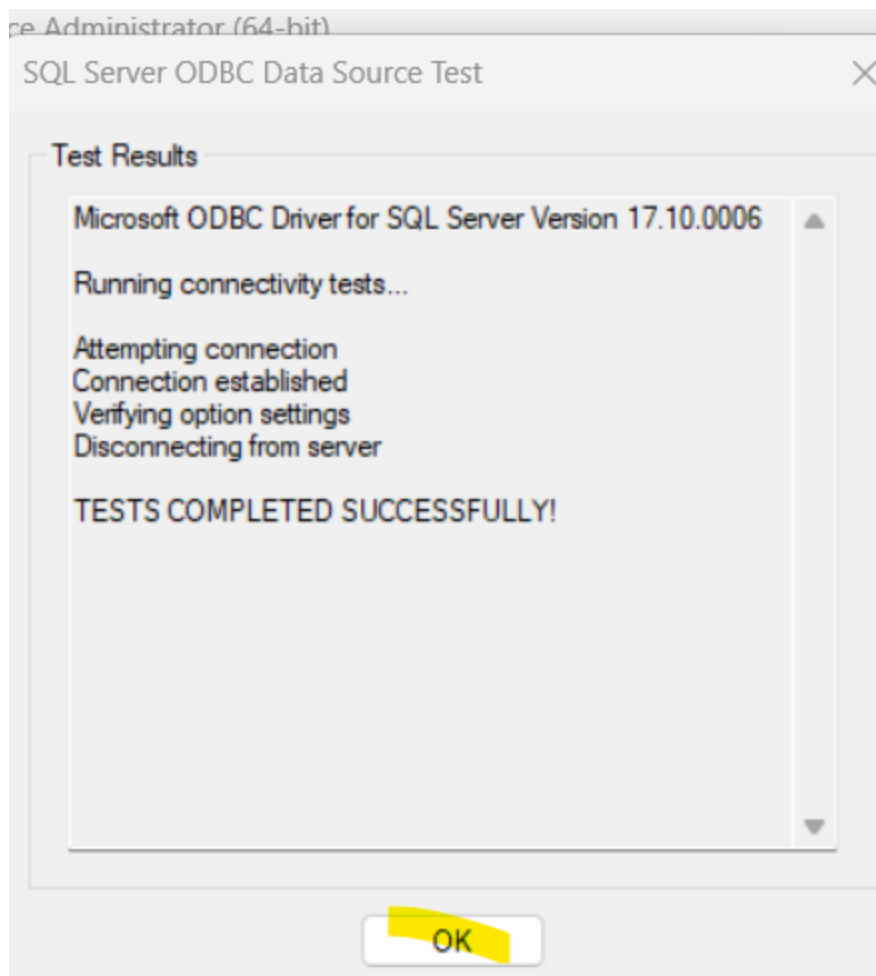
☐ Log ODBC driver statistics to the log file:
C:\Users\raoh2\AppData\Local\Temp\STATS.LOG Browse...
Connect retry count: 1
Connect retry interval (seconds): 10

< Back Finish Cancel Help

Click Test data source.

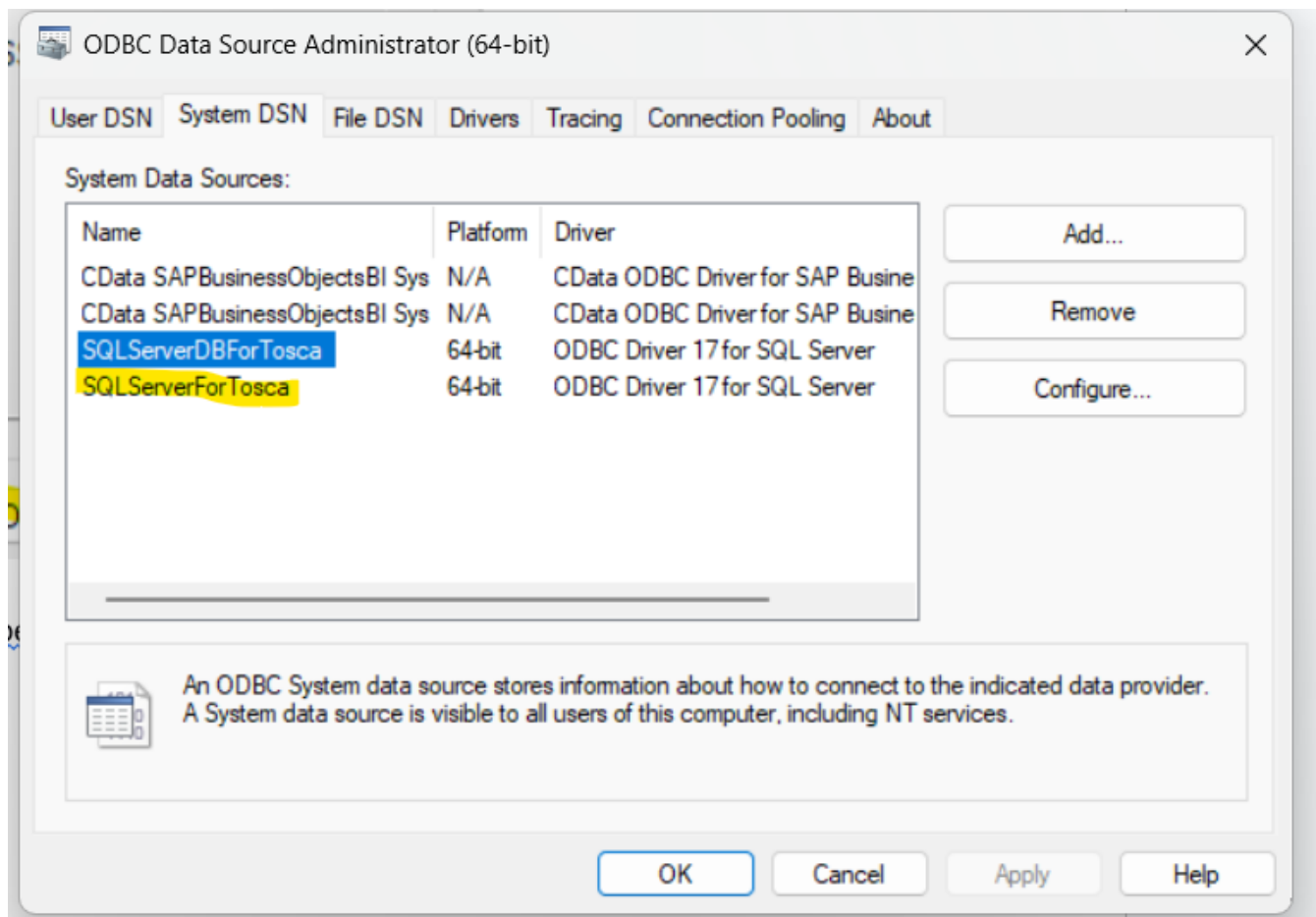


Click OK



Click OK again and you can see below screen.

The ODBC : SQLServerForTosca is created.

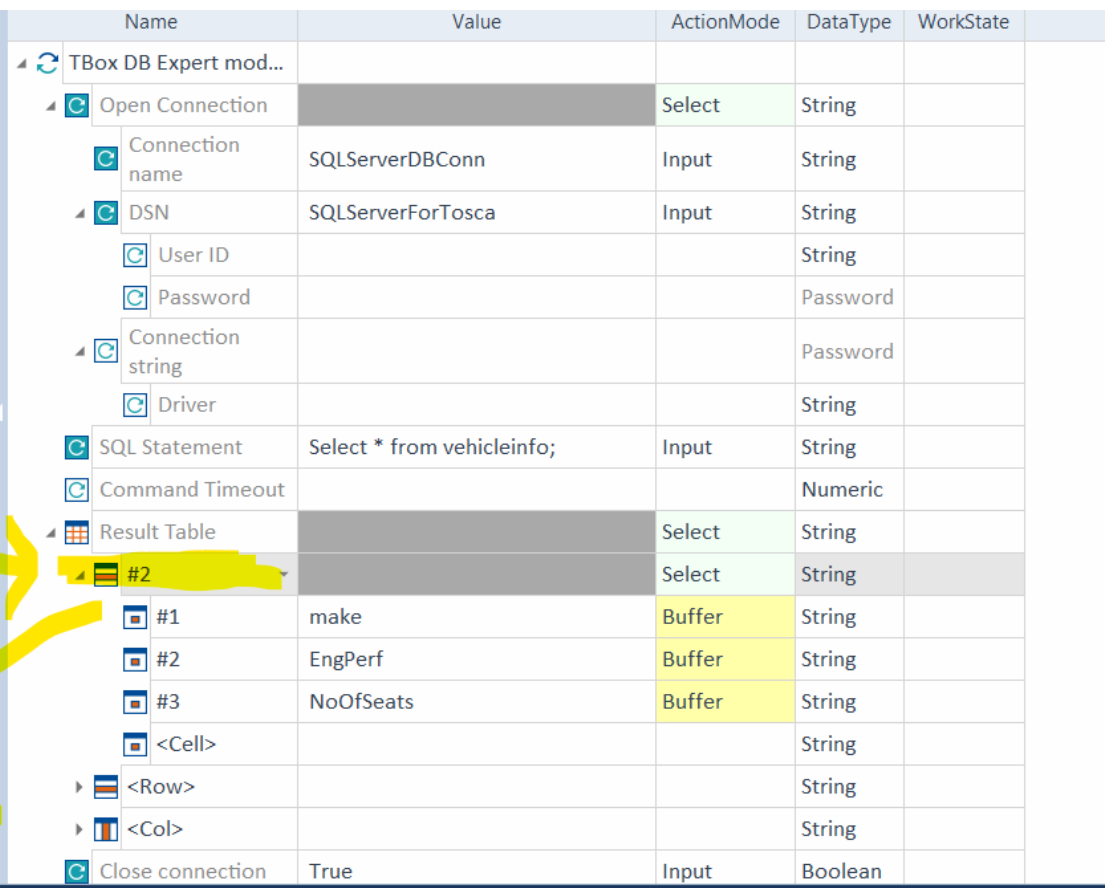


How to connect to Database in Tosca Commander

To automate database testing with Tricentis Tosca:

1. **Configure ODBC:** Create a DSN name using ODBC drivers, specifying the server/host name.
2. **Open Connection:** In Tosca Commander, within a test case, use the **TBOX DB Open Connection** module.
3. **Execute Queries:** Employ the **TBOX DB Run SQL Statement** module to run your SQL queries.
4. **Close Connection:** Utilize the **TBOX DB Close Connection** module to terminate the database connection.

1. Create a folder from the Test case tab. Create a test case, add **TBox DB Expert module**
2. Add the connection name, any name, DNS , name from the odbc connection name that we created. Write sql statement : Select * from vehicleinfo;
3. Result table : Row : #1 means record start from the column header, #2 means first row. \$1 means start from the first data record.. Values returned by **#1 and \$2 are the same.**
4. #1, #2 etc.. in cell represents the column names of the table, instead of #1, #2 etc.. We can also write the actual column name of the table.
5. Under the Value column of the Test case, add the buffer name and select action mode as buffer. The values are stored in the buffer names. Example shown in the below screenshot.



Name	Value	ActionMode	DataType	WorkState
TBox DB Expert mod...				
Open Connection		Select	String	
Connection name	SQLServerDBConn	Input	String	
DSN	SQLServerForTosca	Input	String	
User ID			String	
Password			Password	
Connection string			Password	
Driver			String	
SQL Statement	Select * from vehicleinfo;	Input	String	
Command Timeout			Numeric	
Result Table		Select	String	
#2		Select	String	
#1	make	Buffer	String	
#2	EngPerf	Buffer	String	
#3	NoOfSeats	Buffer	String	
<Cell>			String	
<Row>			String	
<Col>			String	
Close connection	True	Input	Boolean	

Requirements - Test cases - Execution List

https://documentation.tricentis.com/tosca/2320/en/content/requirements/rqm_introduction.htm

- 1) **Create Requirement** : Right click on requirement, create requirement folder, select folder, create **requirement set**, select Requirement Set, right click and create Requirement .
<https://documentation.tricentis.com/tosca/2320/en/content/requirements/requirements.htm>
- 2) **Link requirements to test case: Create test cases** based on the requirements. Now drag and drop each test case to the corresponding requirement . A link will be created on the test case under the requirement.
https://documentation.tricentis.com/tosca/2320/en/content/requirements/link_reqs_to_test_cases.htm
- 3) **Link Requirement with Execution List** : Drag and drop the execution list to the requirement set.
https://documentation.tricentis.com/tosca/2320/en/content/requirements/requirement_sets_in_execution_section.htm
- 4) **Project Status** : Depends on the “Work State” (Planned, In_Work, Completed) of the test case, “**Execution State %**“ will be updated on each Requirement
https://documentation.tricentis.com/tosca/2320/en/content/requirements/project_state.htm

Execution List:

Create a new execution list. Drag and drop the Test case folder / test cases to the Execution list.

From the Execution list, drag and drop the List to the requirement folder.

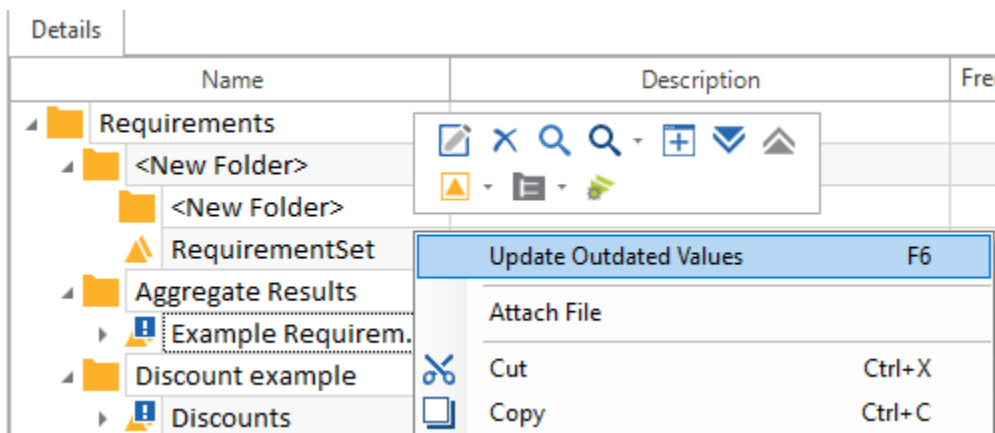
Update values manually

By default, the setting [AutoCalculateRequirements](#) is turned **Off**. This means that you need to manually trigger the update of Requirement values.

Anytime you make a change to a RequirementSet, or to a TestCase that is linked to a RequirementSet, a blue exclamation mark on the RequirementSet icon  indicates that values need to be updated.

To trigger a specific update calculation, right-click the RequirementSet and select **Update Outdated Values** from the context menu.

To update all outdated values at once, select  **Update Outdated Values** from the dynamic menu **Requirements**.



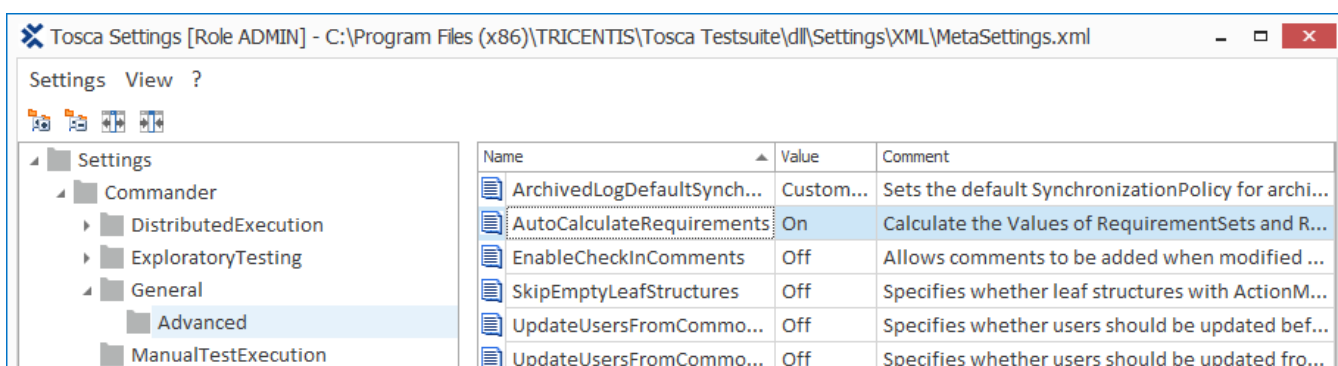
Update Requirement values manually

Update values automatically

You can also configure Tricentis Tosca to automatically update values in the **Requirements section**.

To do so, enable the setting **AutoCalculateRequirements** under **Project->Settings->Commander->General->Advanced**.

Restart Tricentis Tosca for your changes to take effect.



Tricentis Tosca synchronizes changes and automatically updates values in the **Requirements** section.

API Automation Testing using Tricentis Tosca

Overview of APIs, JSON, GET, PUT, POST, DELETE methods

Generate Service Access Key

<https://webservice.toscacloud.com/training/>

Swagger URL for API documentation

<https://webservice.toscacloud.com/api/ui/index.html>

URI / to Scan APIs from Tosca

<https://webservice.toscacloud.com/api/docs/v1/>

2nd Example: SOAP Services with XML

<http://demowebshop.tricentis.com/Plugins/Misc.WebServicesCustomer/Remote/NopService.svc>

<https://webservice.toscacloud.com/Calculator.svc>

3rd Example:

<https://petstore.swagger.io/>

JSON file

<https://petstore.swagger.io/v2/swagger.json>

Sample API services

<https://www.postman.com/postman/published-postman-templates/documentation/ae2ja6x/postman-echo?ctx=documentation>

Tosca Document reference

https://documentation.tricentis.com/tosca/2420/en/content/engines_3.0/api/api_engine.htm

https://support-hub.tricentis.com/open?id=manual&lang=en&path=%2Ftosca%2F2310%2Fen%2Fcontent%2Ftosca_commander%2Fscan_module_non_ui.htm&product=tosca&sessionRotationTrigger=true&type=product_manual&version=2023.1

Overview of APIs

API : Application Programming Interface

Postman :

Example [Starbucks.com](https://www.starbucks.com)

Coffeeshop APIs

URI:

A **URI** (Uniform Resource Identifier) is a formal system for uniquely identifying resources and consists of two types: URLs (Uniform Resource Locator) and URNs (Uniform Resource Name).

API Hosting Server

Create Read Update Delete the data , CRUD operations using API Methods : Post Get Put Delete

What is JSON?

- JSON stands for JavaScript Object Notation. It's a data format used to represent structured data in a text-based way.
-
- It's widely used in web applications for exchanging data between clients and servers.
-
- JSON data is composed of key-value pairs and arrays, making it easy to parse and work with.

Sample JSON API call

JSON

```
{
  "description": "string",
  "id": 0,
  "name": "string"
}
```

Sample JSON ARRAY

```
{
  "data": [
    { "type": "tags", "id": "2" },
    { "type": "tags", "id": "3" }
  ]
}
```

---- another json array example

```
{
  "links": {
    "self": "http://example.com/articles"
  },
  "data": [{
    "type": "articles",
    "id": "1",
    "attributes": {
      "title": "JSON:API paints my bikeshed!"
    }
  }, {
    "type": "articles",
    "id": "2",
    "attributes": {
      "title": "Rails is Omakase"
    }
  }
}]
}
```

HTTP Error codes

200

400

500

How to Scan APIs using Tricentis Tosca?

Launch Tricentis Tosca commander, create workspace. Click on API Testing. OR from module, right click and click Scan - API

Click on the URI, enter the URI <https://webservice.toscacloud.com/api/docs/v1/>

All the API Services will be created

Explore the API scanner , Request, response buttons on API scanner.

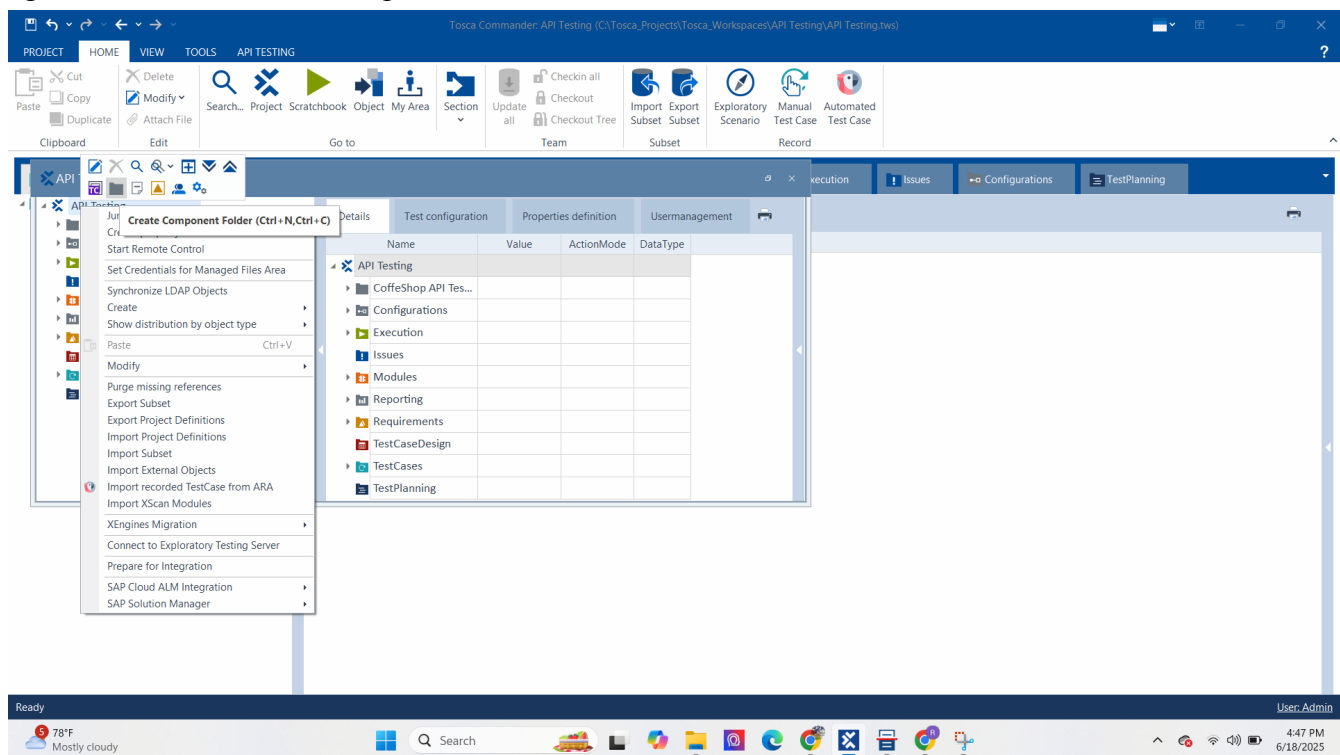
Provide Request payload data and send request / click run button .

Review the response payload.

Export the APIs to Tosca Commander

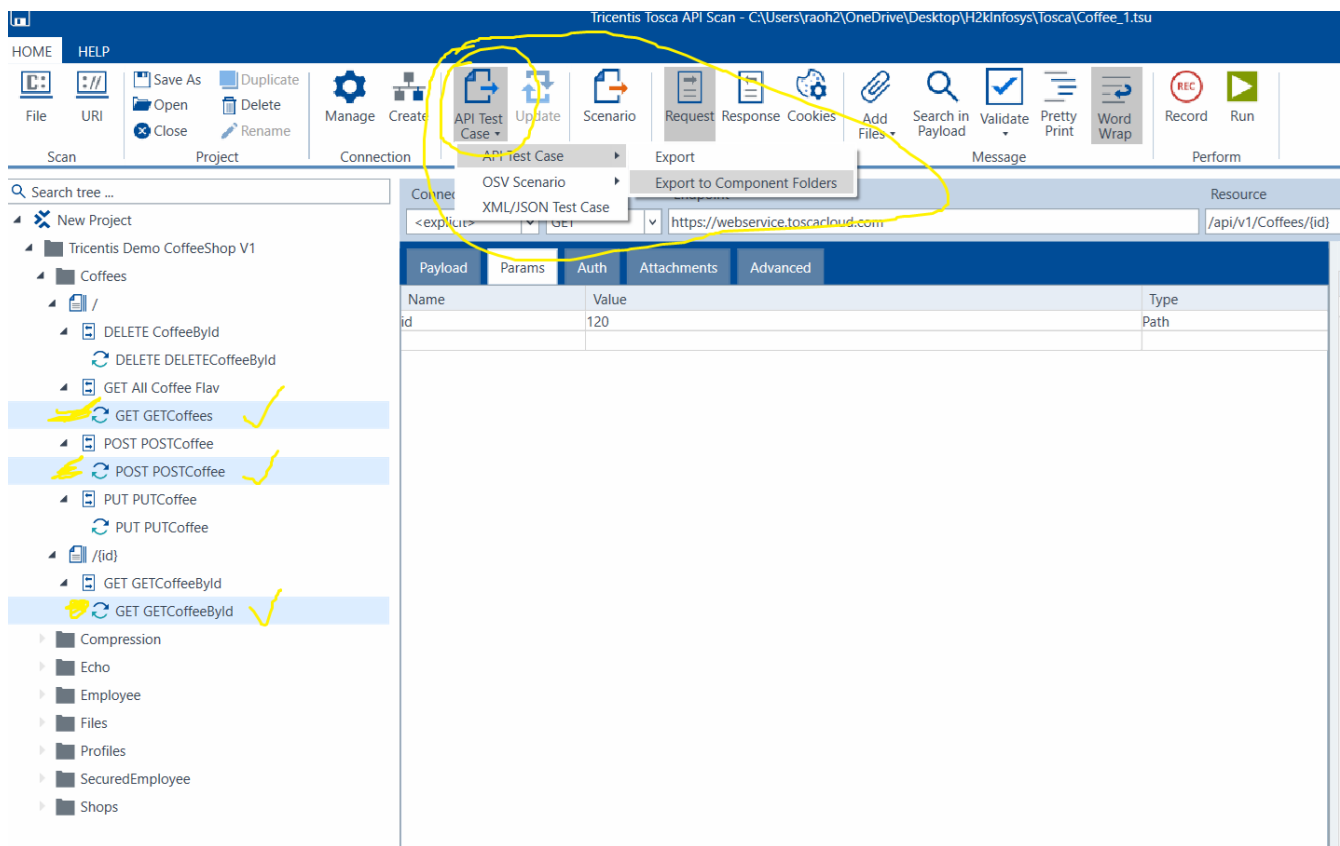
After testing each API service, we can export those API services to Tosca Commander.

From the Tosca Commander, On HOME tab, Click on Project, Select the Workspace / project name and right click and click on “Component folder” next to TC.

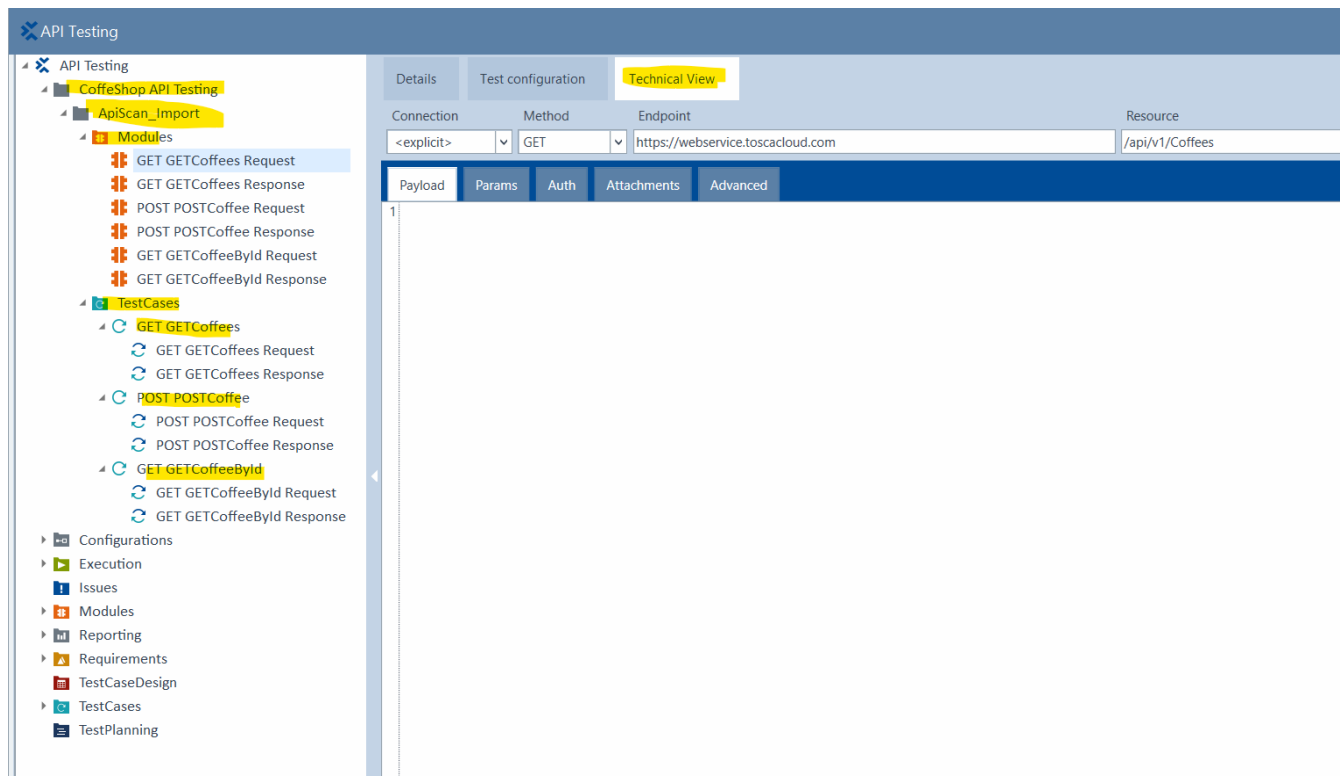


Select the Component folder, go back to the API scanner, select the required API services and click on “API Test Case, == > API Test Case ==> Export to Component Module.”

Selected API Services will be exported under the component folder.



After Exporting, the APIScan_Import folder will be created under the selected component folder and Modules and Test cases are created, for each API Service, One request and one response created.

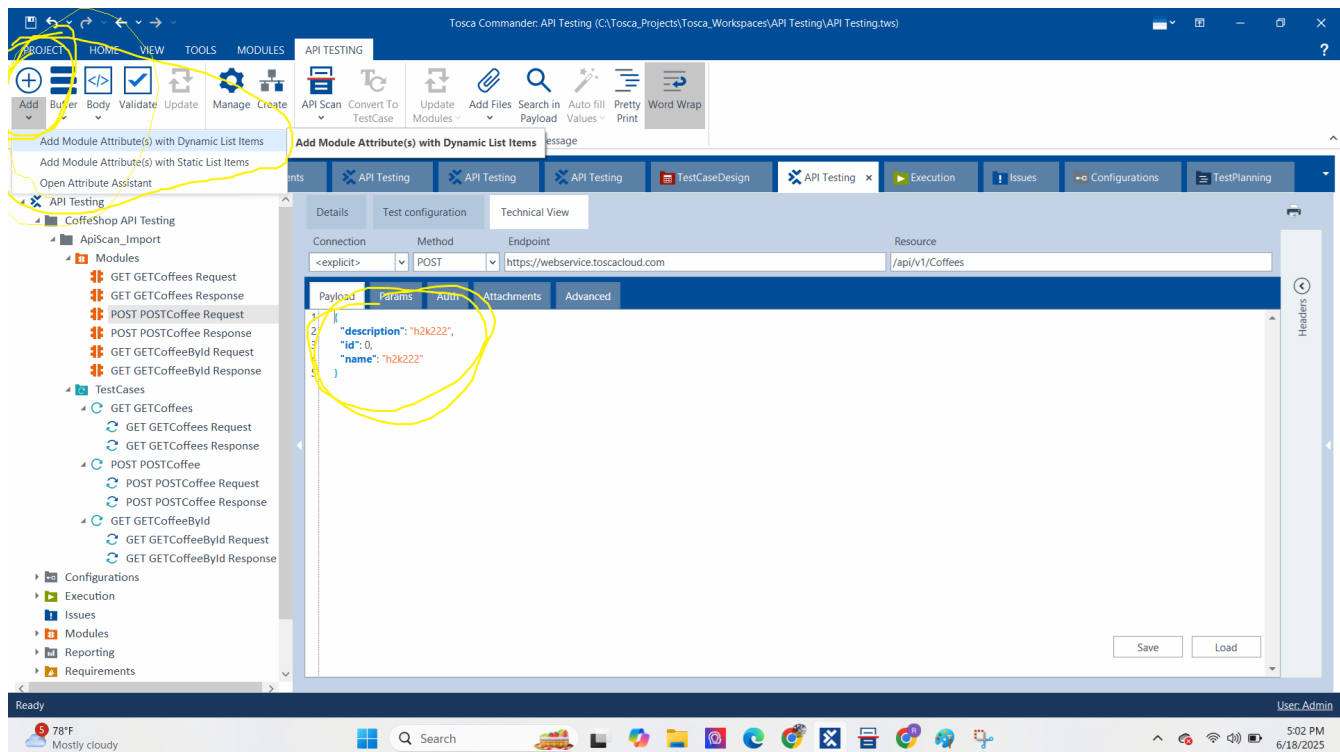


Running test cases from API Component module.

How to Create Module Attributes

https://documentation.tricentis.com/tosca/2420/en/content/engines_3.0/api/api_lists_dynamic.htm

1. Module Attributes with Dynamic List Items
2. Module Attributes with Static List Items
3. Open Attribute Assistant



Auto Fill

https://documentation.tricentis.com/tosca/2420/en/content/engines_3.0/api/api_clipboard.htm

Buffer values to verify response payload

May-June-2025-Batch					
API Automation Testing for Coffeeshop					
ApiScan_Import					
Modules					
04 Delete CoffeeById Request					
id					
04 Delete CoffeeById Response					
StatusCode					
01 Create New Flav Request					
description					
id					
name					
01 Create New Flav Response					
description					
id					
name					
03 Get Coffee By Id Request					
id					
03 Get Coffee By Id Response					
description					
id					
name					
TestCases					
Create Verify Delete Coffe Flav					
Step 1.1 Create New Flav Request					
description					
id					

Details	Test configuration				
Name	Image	TechnicalID	TypeInfoDescription	ValueRange	
Modules					
04 Delete CoffeeB...					
id					
04 Delete CoffeeB...					
StatusCode					
01 Create New Fla...					
description					
id					
name					
01 Create New Fla...					
description					
id					
name					
03 Get Coffee By I...					
id					
03 Get Coffee By I...					
description					
id					
name					

May-June-2025-Batch					
API Automation Testing for Coffeeshop					
ApiScan_Import					
Modules					
TestCases					
Create Verify Delete Coffe Flav					
Step 1.1 Create New Flav Request					
description					
id					
name					
Step 1.2 Create New Flav Response					
description					
id					
name					
Step 2.1 Get Coffee By Id Request					
id					
Step 2.2 Get Coffee By Id Response					
description					
id					
name					
Step 3.1 Delete CoffeeById Request					
id					
Step 3.2 Delete CoffeeById Response					
StatusCode					
Automobile Insurnace Application					
Configurations					
Data Driven Testing - TDS					

Details	Test configuration				
Name	Value	ActionMode	DataType	WorkState	
TestCases					
Create Verify Delete Coffe Flav					
Step 1.1 Create New Flav Request					
description					
id					
name					
Step 1.2 Create New Flav Response					
description					
id					
name					
Step 2.1 Get Coffee By Id Request					
id					
Step 2.2 Get Coffee By Id Response					
description					
id					
name					
Step 3.1 Delete CoffeeById Request					
id					
Step 3.2 Delete CoffeeById Respo...					
StatusCode					

JSON Operations:

https://documentation.tricentis.com/tosca/1600/en/content/standard_subset/engines_3.0/json.htm

JSON Schema Validation

https://documentation.tricentis.com/tosca/2320/en/content/standard_subset/engines_3.0/json.htm#json_schema_validation

SampleJSON.json

```
{
  "description": "H2kinfosys",
  "name": "h2k"
}
```

Sample Schema File SampleJSON-Schema.json

```
{
  "description": "string",
  "name": "string"
}
```

Recording Messages

To Capture all the API services from UI

Multi User Workspace , Check In / CheckOut , Branches

Select Multi User Workspace repository : SQLite

- 1) Create New Repository Main repository
- 2) Create New Workspace

Admin : Credentials (User id : admin / pwd admin)

Main Branch : Master

Similar to GitHub , we can check out the required test cases or modules etc.. make changes and check in.

Check Out

Check In

Create New Branch :

https://documentation.tricentis.com/tosca/2420/en/content/tosca_commander/creating_branches.htm

Tricentis Tosca Distributed Execution DEX Server Configuration, Agents

NOTE: Tosca DEX to end of life

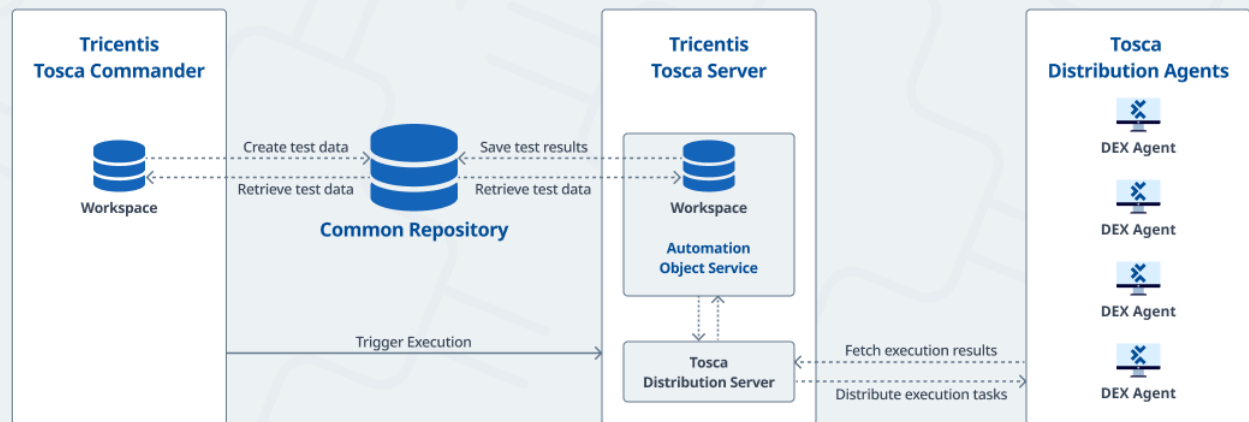
<https://support-hub.tricentis.com/open?number=NEW0001523&id=post>

https://documentation.tricentis.com/tosca/2420/en/content/distributed_execution/distributed_execution_overview.htm

https://documentation.tricentis.com/tosca/2320/en/content/distributed_execution/setup_with_aos_https.htm



Distributed Execution with Automation Object Service



🧠 1: What is Tosca Distributed Execution?

- A smart way to speed up automated testing
- Runs tests across multiple machines—your network PCs, VMs, or the cloud
- Keeps your computer free and responsive

🚀 2: Why Use Distributed Execution?

- ✅ **Faster testing:** Multiple machines run tests at the same time
- 💻 **No interruptions:** Your machine isn't tied up
- 💡 **Better resource usage:** All available machines are put to work

👥 3: Who Can Use It?

- Only available in multi-user workspaces

- Everyone shares a common repository for test cases and results
-

4: Main Components of Distributed Execution

1. User Machines

- Where tests are created and test runs are triggered

2. Tosca Distribution Server

- Uses Automation Object Service (AOS)
- Manages test distribution
- Has its own Tosca workspace
- Connects to the shared repository

3. Agent Machines

- Actually run the tests
 - No need for local Tosca workspaces
 - Send results back via AOS
-

5: How It All Works Together

Tosca Commander / Workspace → Distribution Server → Agent Machines → Common Repository

- You create and trigger tests
- Server sends them out to available agents

- Agents run tests and report back
 - Everything is saved in one shared location
-

✓ 6: Key Benefits Recap

- Test faster without blocking your system
- Ideal for large or complex test suites
- Great for collaborative teams

Install Tosca Server

Configure

Tosca Server : Dex / configuration , user pwd : admin / admin

Prerequisite : Version of Tosca commander and Tosca Server should be the same.

Tosca Commander, Tosca Server, Tosca Agent Machine should be able to talk to each other. For practice purposes, we are installing all these in our local machine / laptop / desktop.

User Machine (Tosca Commander) =====> DEX Server Machine(Tosca Server) =====> Agent Machines (virtual servers)

Steps to execute test cases on DEX

1. Install same version of Tricentis Tosca Server and Tosca Commander (as part of Tosca commander installation, Agent will be installed automatically)
2. Create a common repository and add all the test cases in Multi User workspace env.
3. Create a workspace in Tosca Commander.
4. Update AOS settings on Tosca Server for Distributed execution
5. Add the Repository name and workspace name in AOS settings in Tosca Server.
6. Start the agent and make sure of the configuration settings. The Monitor Icon is green.
7. Update the settings on Tosca Commander : Settings →Commander->Distributed Execution->Monitor URL & Server
Last one on Settings Tricentis Services : <http://localhost/> (or the IP Address of Tosca Server)
8. From common repository, Under Test Execution, Create folder, , create configuration, add the executing list which contains test cases and connect the agents, Execute the test cases

Step 2: Create new repo and workspace, on create admin user tab, uid/pwd : Admin / Admin

Repository Name : MayJune2025

Step 3: Create new workspace with existing repository : MayJune2025

New Workspace: H2kInfosysValidation

Login / Pwd: Admin / Admin

Step 4: Tosca Server : AOS Settings

<http://localhost/>

Click on settings on Tosca Server dashboard

Login with : UID/PWD Admin / Admin

Click on Automation Object Service

Click on Add new

Enter the Root name as the Common Repository name : MayJune2025

Enter the Workspace name : H2kInfosysValidation

UserName , Pwd: Admin / Admin (Created while creating new repository on Tosca Commander)

Click on Save Button

Click Ok on “Restarting Services” window

Click on Tricentis Icon on top left corner to open the Tosca Server Dashboard, click on refresh on browser.

If the Icons are still not enabled / blue color, Open Services from Windows search on the task bar.

On Search Screen, click on ConsoleService , start

Step 5: Start Agent

Agents are installed as part of the Tosca commander installation.

C:\Program Files (x86)\TRICENTIS\Tosca Testsuite\DistributedExecution

ToscaDistributionAgent.exe

Right click and run as admin

Green monitor icon should be added on right hand corner of the Task bar

Click on Green monitor , click on configuration

Verify server details, ip address , workspace etc..

Step 6: Update Settings on Tosca Commander

Settings → Commander->Distributed Execution->Monitor URL : <http://localhost/Monitor/>

Server : <http://localhost/DistributionServerService/ManagerService.svc>

Last one on Settings **Tricentis Services** : <http://localhost/> (or the IP Address of Tosca Server)

Step 7: Executing Test cases on Common Repository

Open the common repository : MayJune2025

Under Execution Tab, Right click on Configuration , Click on Refresh Agents.

No Of Agents should be updated to 1 , since we started one agent from a local machine.

Expand Any, RDP, SupportsClassic , Local Machine / Agent name should be available .

Check out the Test Events

Create a new test event under Test Events.

Under the new Test events , drag and drop “Any” from the Configuration , Drag and drop the required execution list from the Execution Tab.

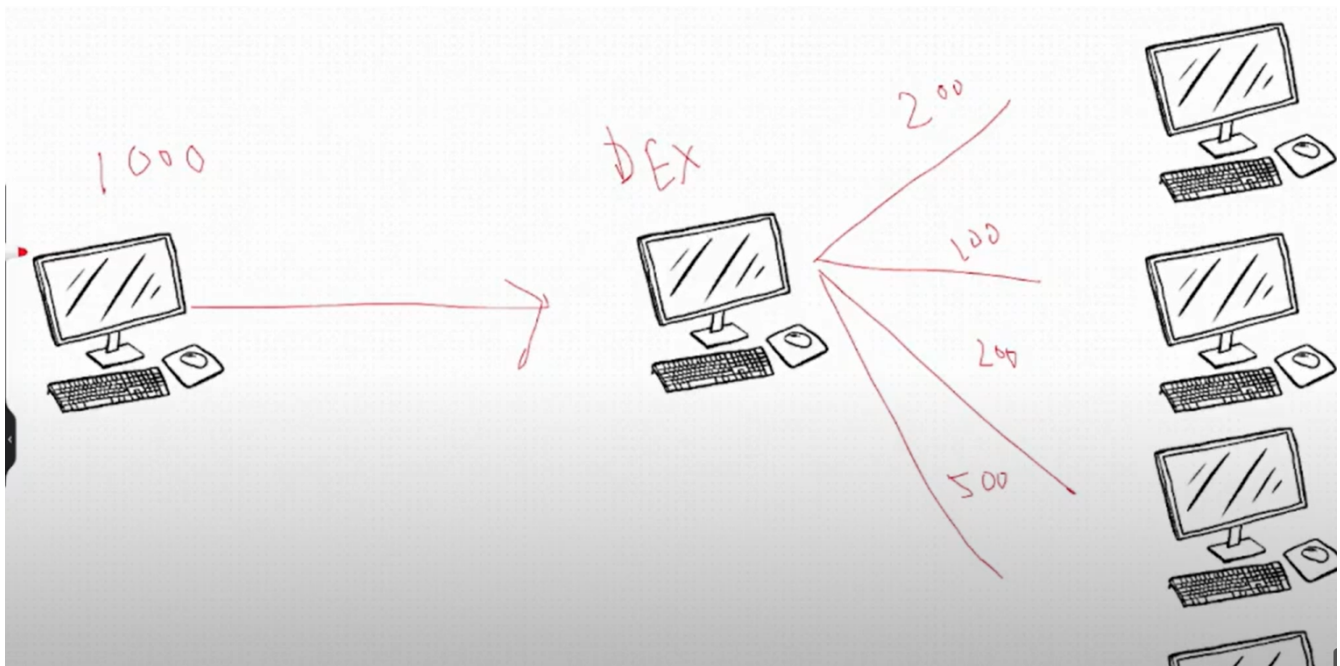
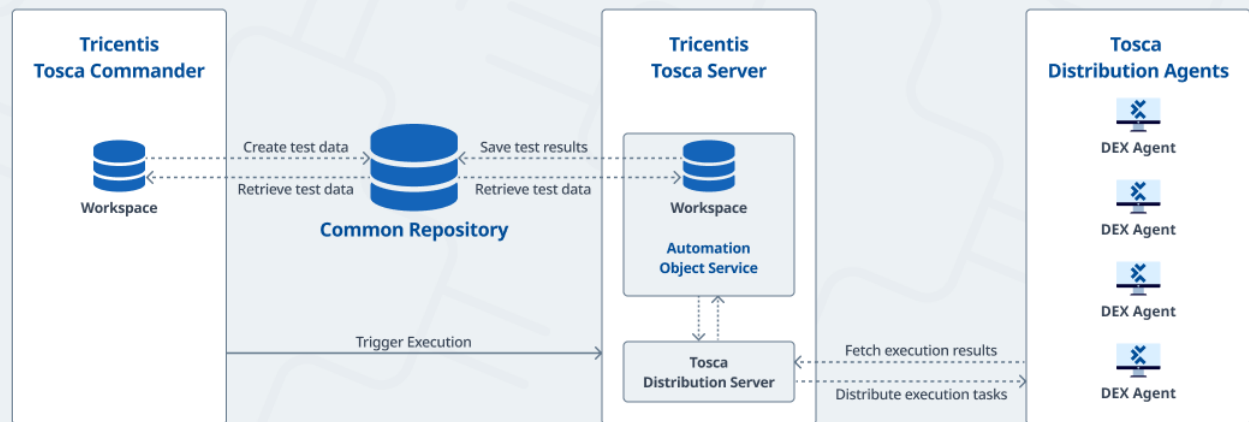
Refresh the Agent on Configuration.

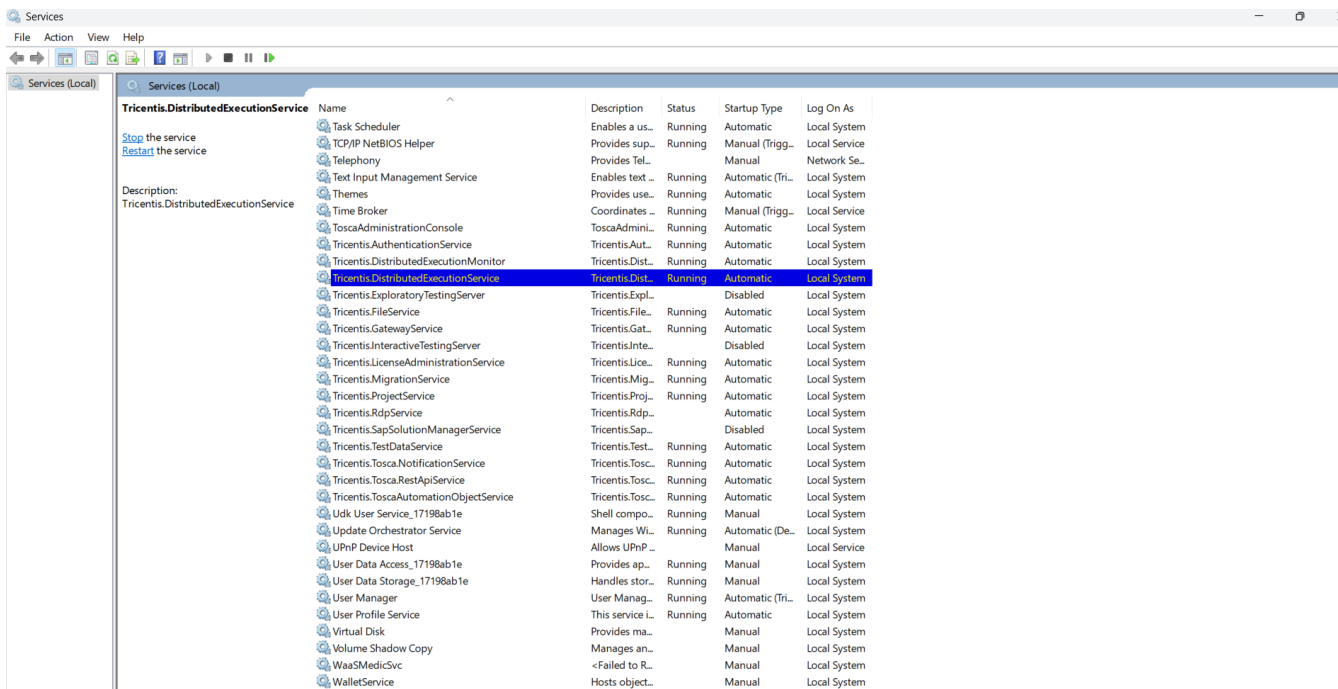
Select the “Test Event Folder” and right click, click on Execute Now

Step 8: Open the Tosca Server : dashboard, click on Distributed Execution , and monitor the test execution status. Agent View for status of Agents, Execution View for status of test case execution .



Distributed Execution with Automation Object Service





ERROR: Tosca Server might not be working and all the services are down

From Windows Services:

1. Open **Services** from the Widows **Start** menu.
2. Find **Consul.Service** in the list.
3. Right-click on the service name and select **Restart**.

Step 1:

Create a new repository and a workspace under multi user workspace ,
Example: Repository: MyMainRepo Workspace : MyDexWorkSpace

Create a test case to launch a website and close the browser.

Drag and drop the test cases to under execution tab, rename the newly creed execution list to
DexTestEventsForAutoInsu

Drag and drop this DexTestEventsForAutoInsu under TestEvents

Step 2:

Create a new workspace using existing repository Example: Repository MyMainRepo
New Workspace name : MyAOSworkspace

Project Settings on Tosca Commander :

Commander ==> Distribute Execution ==> Monitor URL ==> localhost:80/Monitor

Server End Point address :

<http://localhost/DistributionServerService/ManagerService.svc>

On settings page, last setting on bottom of the page : **Tricentis Services** : Server end point address :

<http://localhost:80/>

Step 3:

Open Tosca server localhost:80

Click on settings on top right corner, Admin / Admin

Click on Automation Object Service AOS :

Click on add new , for add new workspace

Repository type

Project Root name is the repository name : MyMainRepo

Workspace name : MyAOSworkspace

User / pwd ; Admin / Admin

Click on save

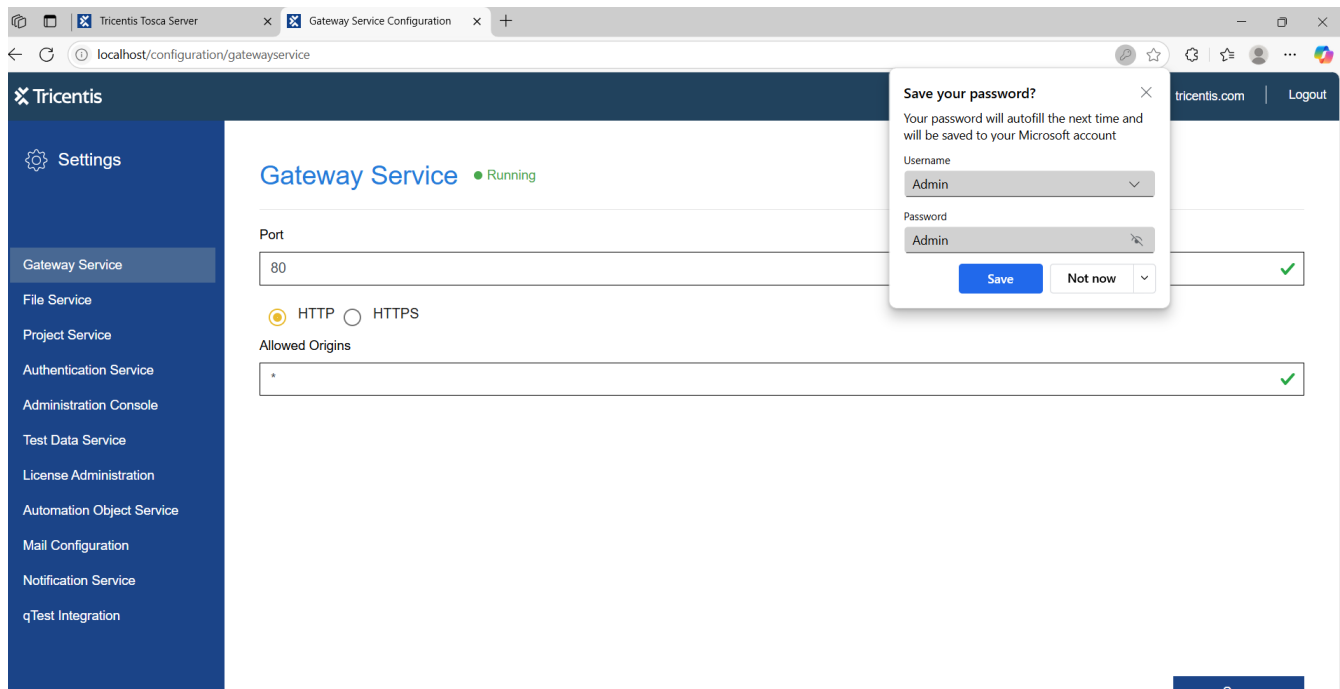
Restarting services window , click ok

Click on the Tricentis Icon on top left corner to open the Tosca Server Dashboard : <http://localhost/>

After restarting services, Tosca server might be down , The Follow below steps and make sure

From Windows Services:

4. Open **Services** from the Windows **Start** menu.
5. Find **Consul.Service** in the list.
6. Right-click on the service name and select **Restart**.



Step 4:

Creating Agents:

From C:\Program Files (x86)\TRICENTIS\Tosca Testsuite\DistributedExecution

Find ToscaDistributionAgent

Click on ToscaDistributionAgent , run as admin to start the agent

Agent monitor should show on right hand corner of the windows task bar in green color

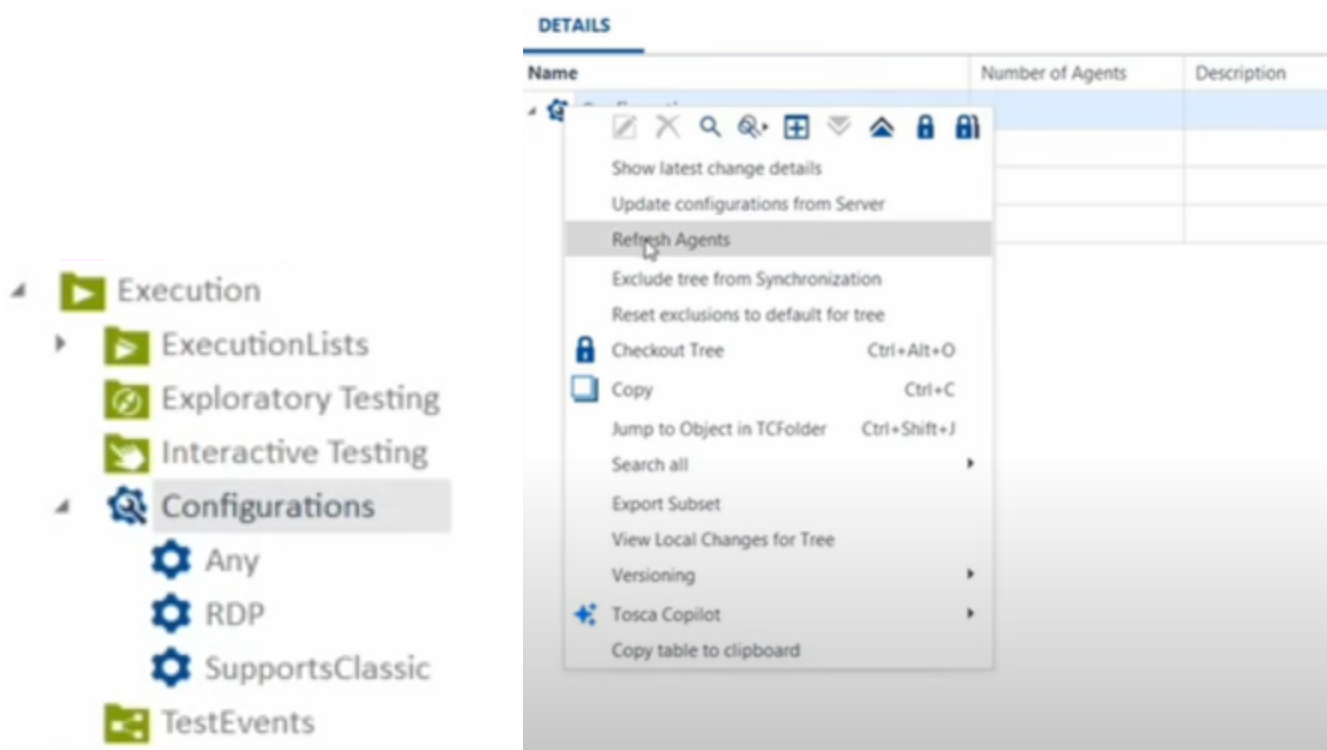
If we need to update the configuration of agent, click on config on the agent from the task bar

Step 5: From Tosca commander

Open the Main Repository

Click on project, execution folder, configuration folder, Any , RDP, Supports classic,

Right click on the configuration folder, click on Refresh agents.



Create a folder under Test Events

Drag and drop Any from configuration folder to the Test Events folder.

Drag and drop the execution list folder under the Any in Test Events folder

DETAILS		
Name	Number of Agents	Description
TestEvents		
DexDemo	1	
Any	0	
Demo Webshop Register		

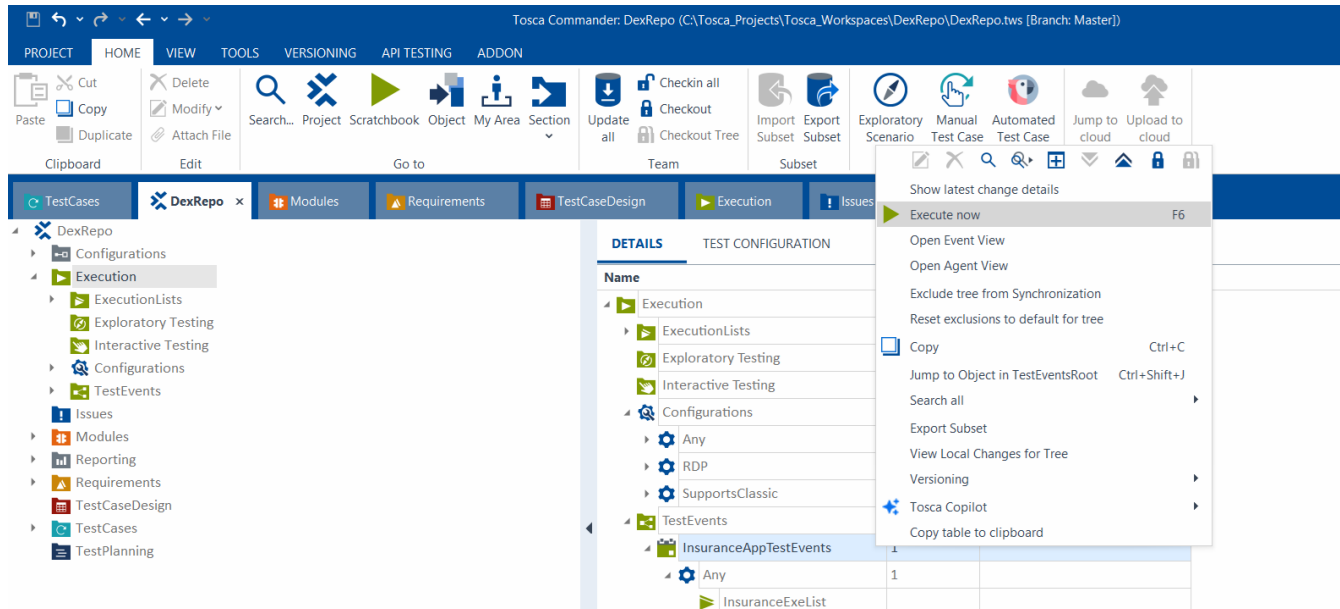
Step 5:

Open the Tosca Server URL : localhost:80 localhost

Click on Dex Distributed Execution

View the Agent view and Event view

From Tosca Commander, Under Test Events, Right click on the folder that we created and Execute Now



Observe the Agent view and Events view.

Agent view : Executing Icon should show one since we are running one agent.

Event View : can view the results of the execution.

Refresh the browser on Tosca Server , for Distributed Execution screen ,

Click on Events View for the test execution status.

localhost/Monitor/EventManager

"Tricentis Automation Extension for Chrome" started debugging this browser

Event View

11 Events

0 0 1 9 1

State ↓	Name	Actual Result	Start Time (EDT)	Creator
	InsuranceAppTestEvents	1 Passed	Jun 25, 2025, 06:14 PM	Admin
	InsuranceAppTestEvents	1 no Result	Jun 25, 2025, 06:05 PM	Admin
	<New TestEvent>	1 Canceled	Jun 25, 2025, 04:50 PM	admin
	InsuranceAppTestEvents	1 Canceled	Jun 25, 2025, 05:42 PM	Admin
	InsuranceAppTestEvents	1 Canceled	Jun 25, 2025, 05:37 PM	Admin

Execute Test cases from Command Prompt tshell.exe

Execute Test cases using APIs form Tosca execution client github

Jenkins Install, Launch Jenkins, Schedule Jobs on Jenkins to Execute Tosca Test Cases

We can execute the Tosca test cases from the command prompt and also using API calls with Tosca Execution Client.

How to execute Tosca Test cases from the command prompt?

https://documentation.tricentis.com/tosca/2310/en/content/tosca_commander/tshell_commands.htm

https://documentation.tricentis.com/tosca/2310/en/content/tosca_commander/tshell.htm

NOTE: Replace the workspace name, location and the Execution List folder name
Below example is for a single user workspace . If it is a multi user workspace,add a checkout.

On your local drive ,Create Script.tcs or any .tcs file, in this example ,I created Script.tcs under C:\Tosca_Projects\TCSHELL with below info .

```

jumpnode "/Execution/ExecutionLists/H2kExe"
task Run
save

```

Execute below command from command prompt.

```
"C:\Program Files (x86)\TRICENTIS\Tosca Testsuite\ToscaCommander\tcshell.exe" -workspace  
"C:\Tosca_Projects\Tosca_Workspaces\JenkinsTest\JenkinsTest.tws" -login "" ""  
"C:\Tosca_Projects\Script.tcs"
```

We can also execute this Tosca Command - TC Shell script from Jenkins.

How to execute the test cases using “Tosca Execution Client“

Open the github url and copy the text from Tosca_execution_client.ps1

<https://github.com/Tricentis/ToscaExecutionClient>

https://documentation.tricentis.com/tosca/2420/en/content/continuous_integration/tosca_execution_clients.htm?Highlight=tosca_execution_client.ps1

Login to Tosca Server, click on user administration and click on my account, click on Next to Tokens, Click on Add New . Client ID and Client Secret Code will be generated.

Set-ExecutionPolicy Bypass -Scope Process -Force &

```
"C:\Tosca_Projects>tosca_execution_client.ps1" -toscaServerUrl "http://localhost/" -events  
["H2kExe"] -projectName "JenkinsTest" -clientId "DDjZp5pRj0WfwH_bR5KzXQ" -clientSecret  
"bg-HMwJjkmnFqS7FAyhsAIYQ3IsNdz0yRAF_0VR17EQ"
```

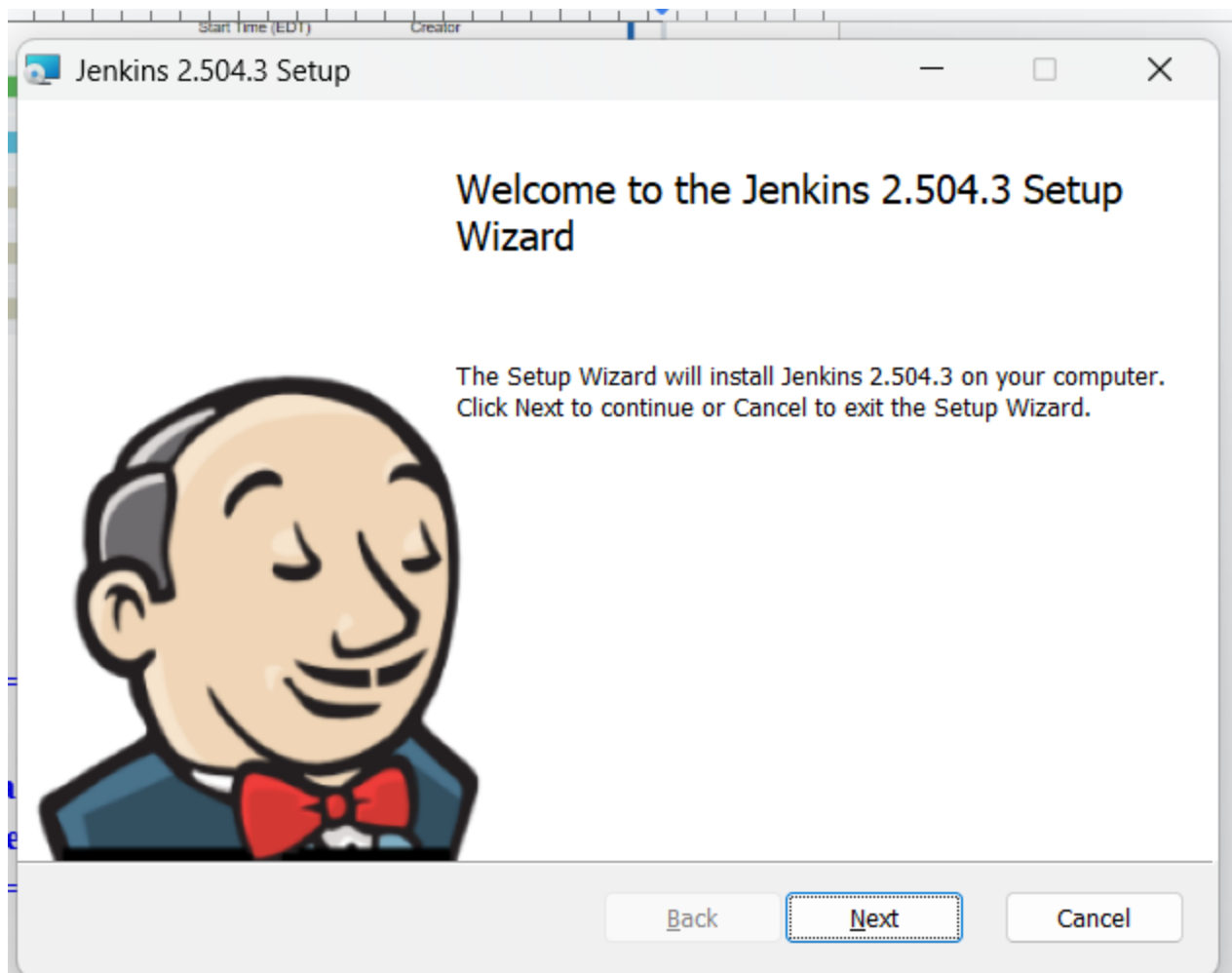
cd to the folder where tosca_execution_client.ps1 is located.

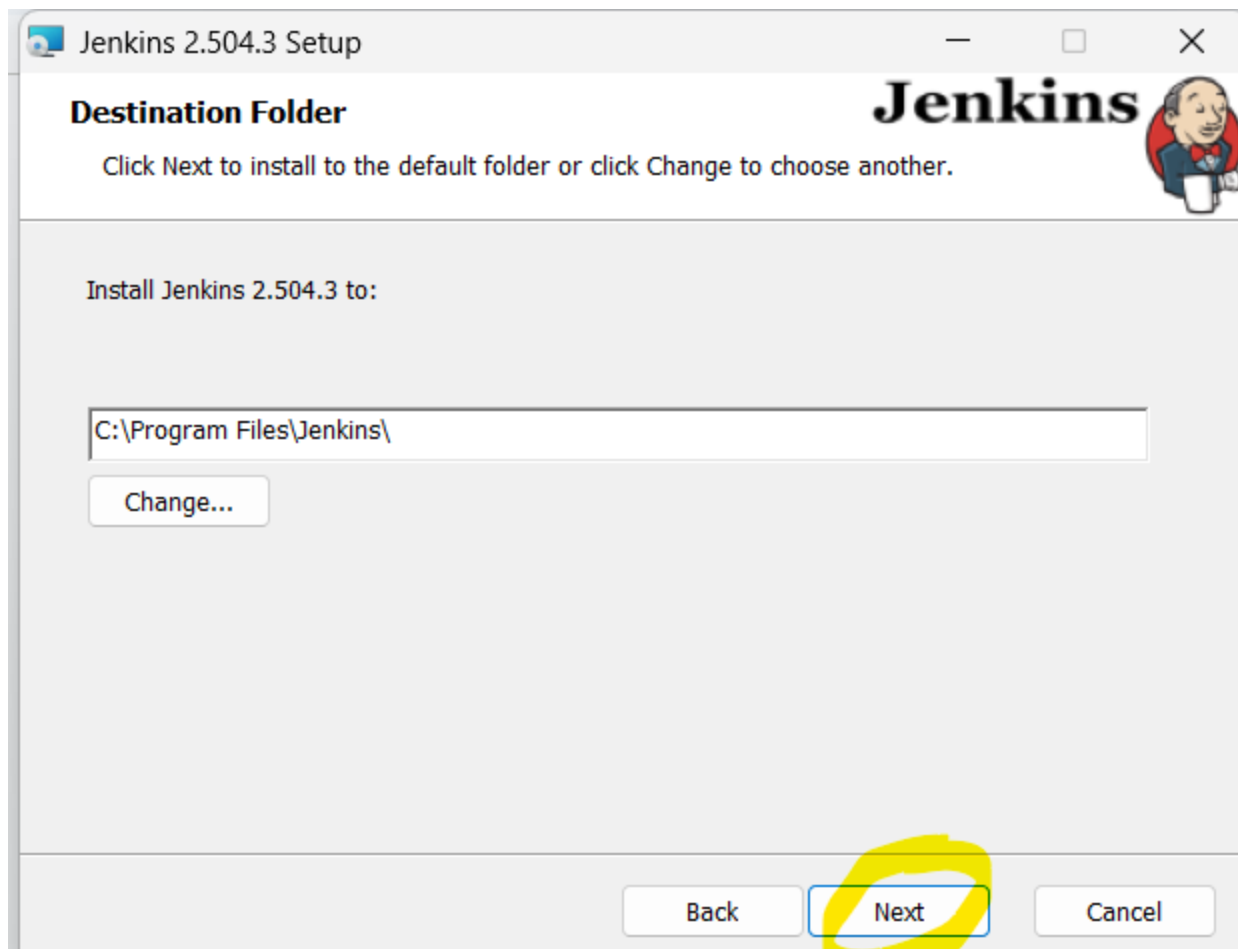
```
powershell -ExecutionPolicy Bypass -File "C:\Tosca_Projects\tosca_execution_client.ps1"  
-toscaServerUrl "http://localhost/" -events ["DexTestEventsForH2kinfosysValidation"]  
-projectName "MayJune2025" -clientId "XE538JM0CEaGNWLECM0Jw" -clientSecret  
"MFuYI7a0pEWDcHELTkdySg_wudMOVHMKGeNGOiFK3WTg"
```

Step 1: Download and Install Jenkins

<https://www.jenkins.io/download/>

Click on Windows, jenkins file will be downloaded, double click on Jenkins file to install





Jenkins 2.504.3 Setup

Service Logon Credentials

Enter service credentials for the service.

Jenkins 2.504.3 installs and runs as an independent Windows service. To operate in this manner, you must supply the user account credentials for Jenkins 2.504.3 to run successfully.

Logon Type:

☒ Run service as LocalSystem (not recommended)

☐ Run service as local or domain user:

Account:

Password:

Test Credentials

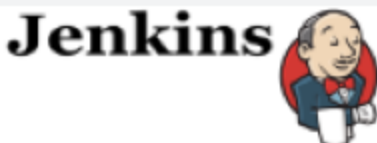
Back Next Cancel

Click on Test port and make sure the default port 8080 is available . If not enter any other port number and click on test port to validate the port number is available.

Jenkins 2.504.3 Setup

Port Selection

Choose a port for the service.



Please choose a port.

Port Number (1-65535):

Test Port

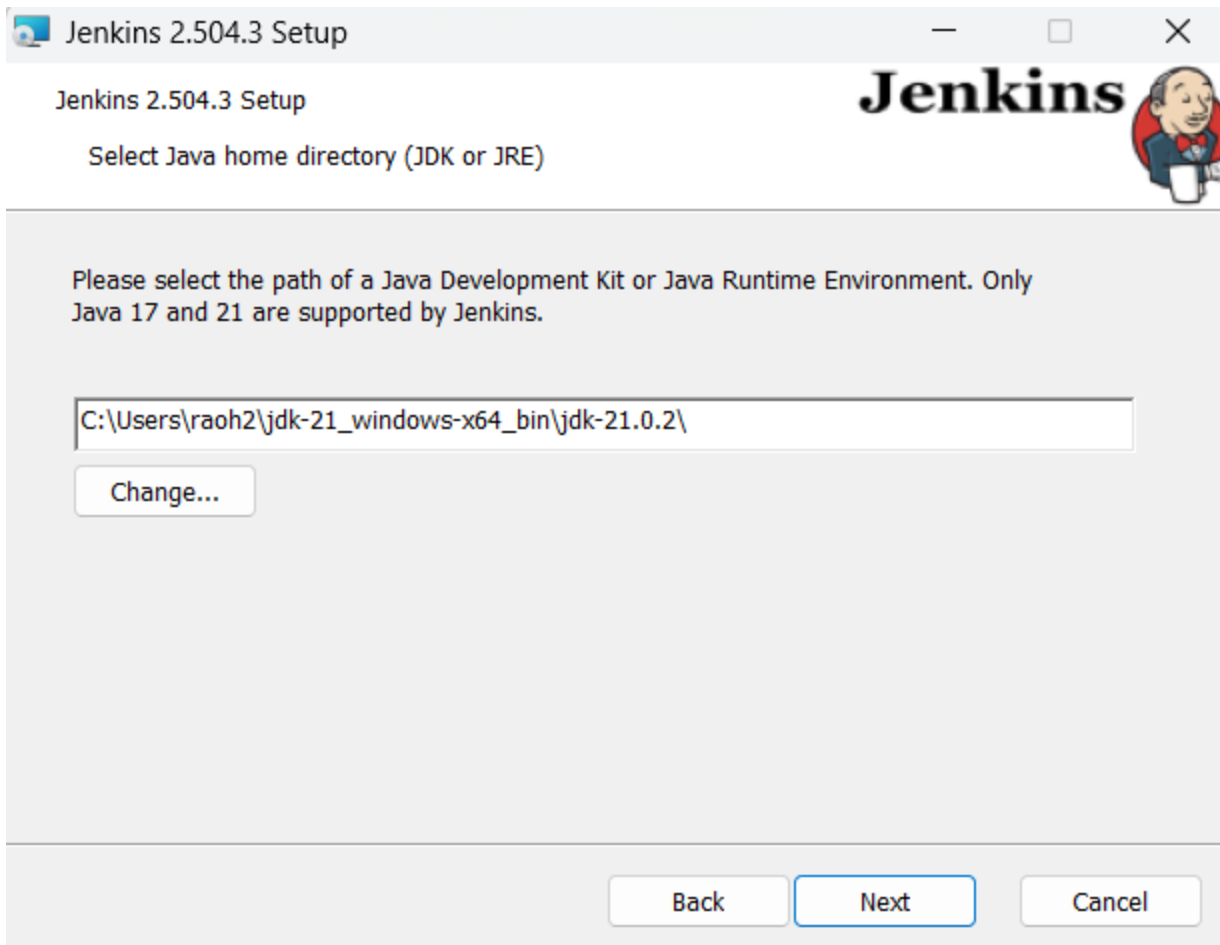


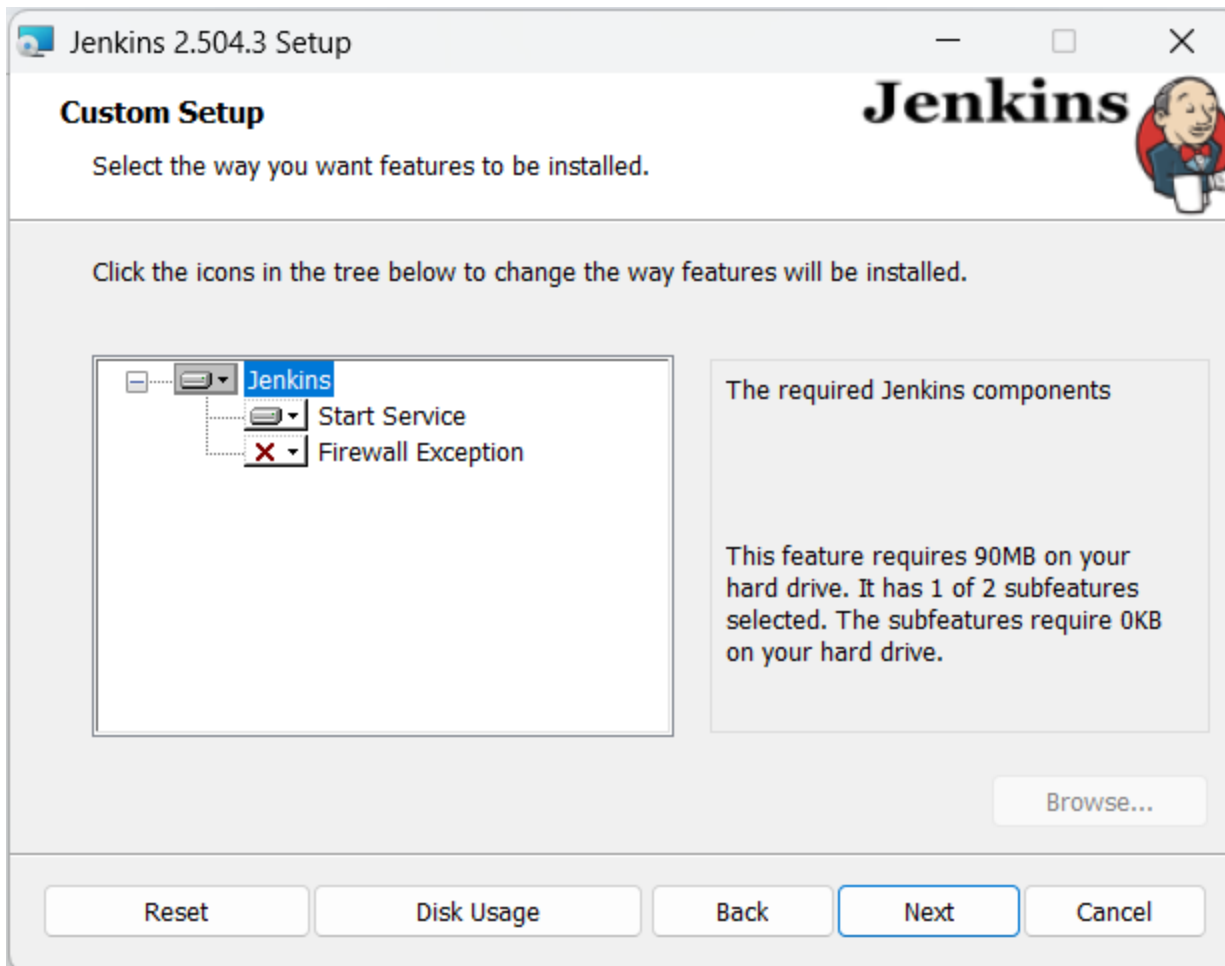
It is recommended that you accept the selected default port.

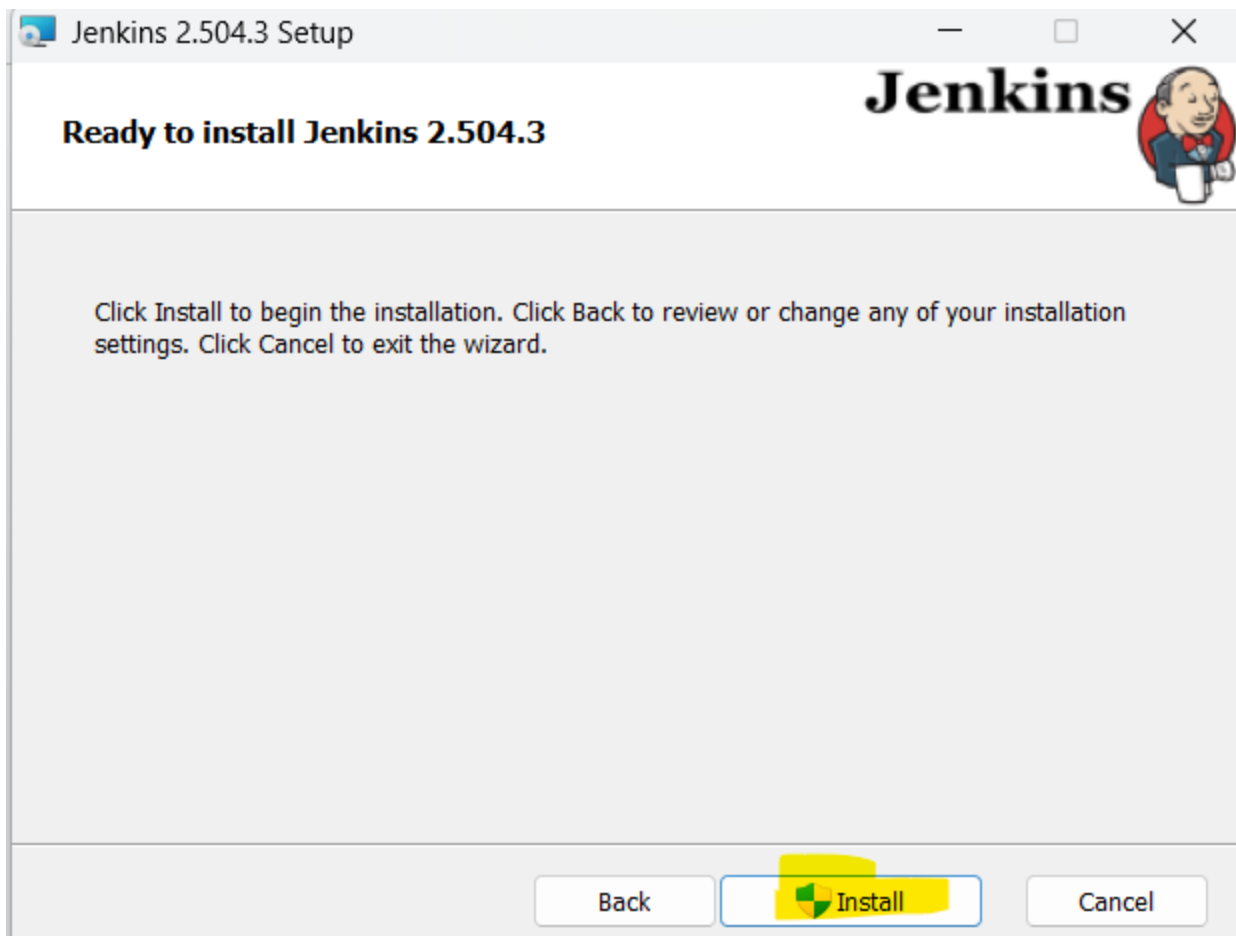
Back

Next

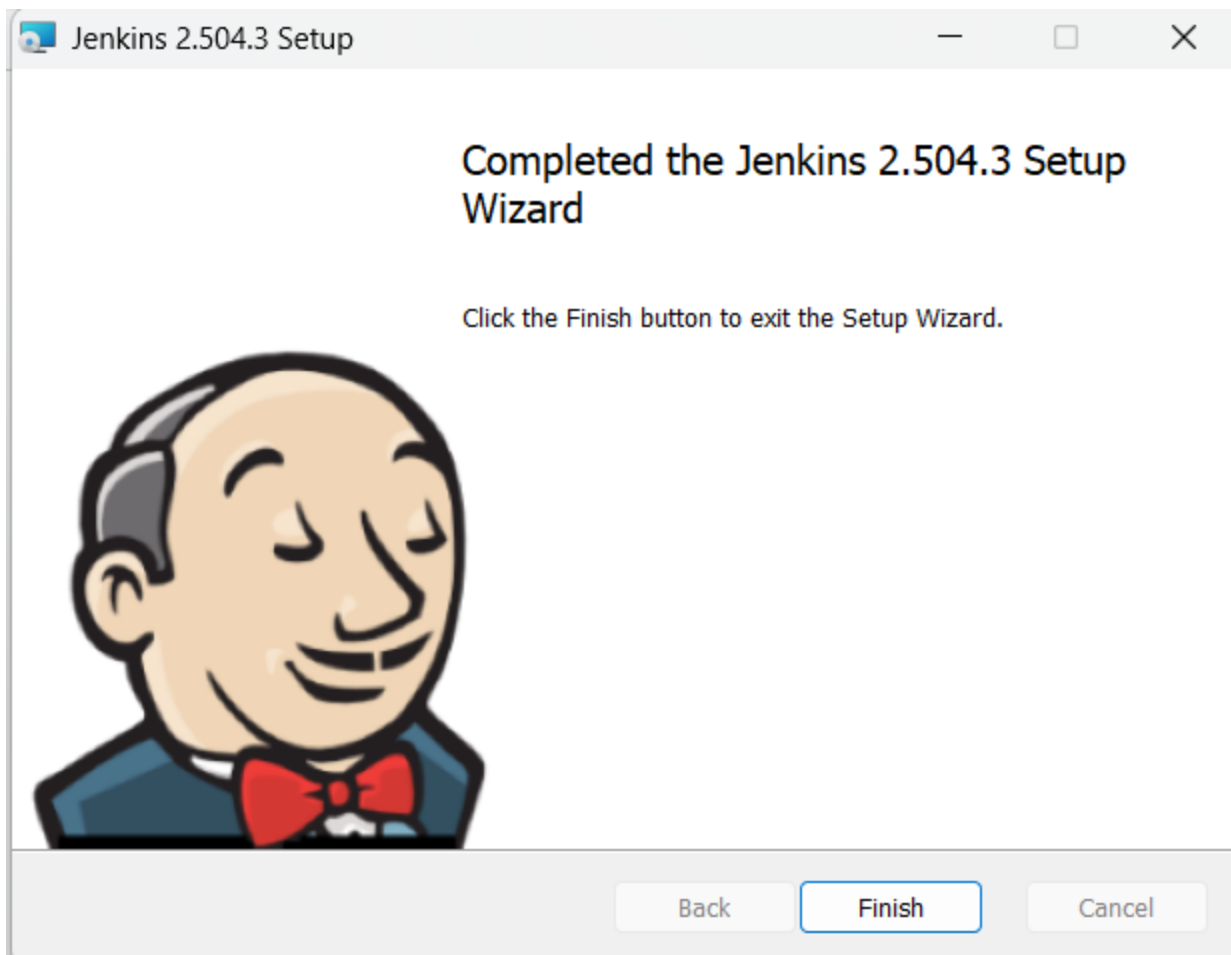
Cancel



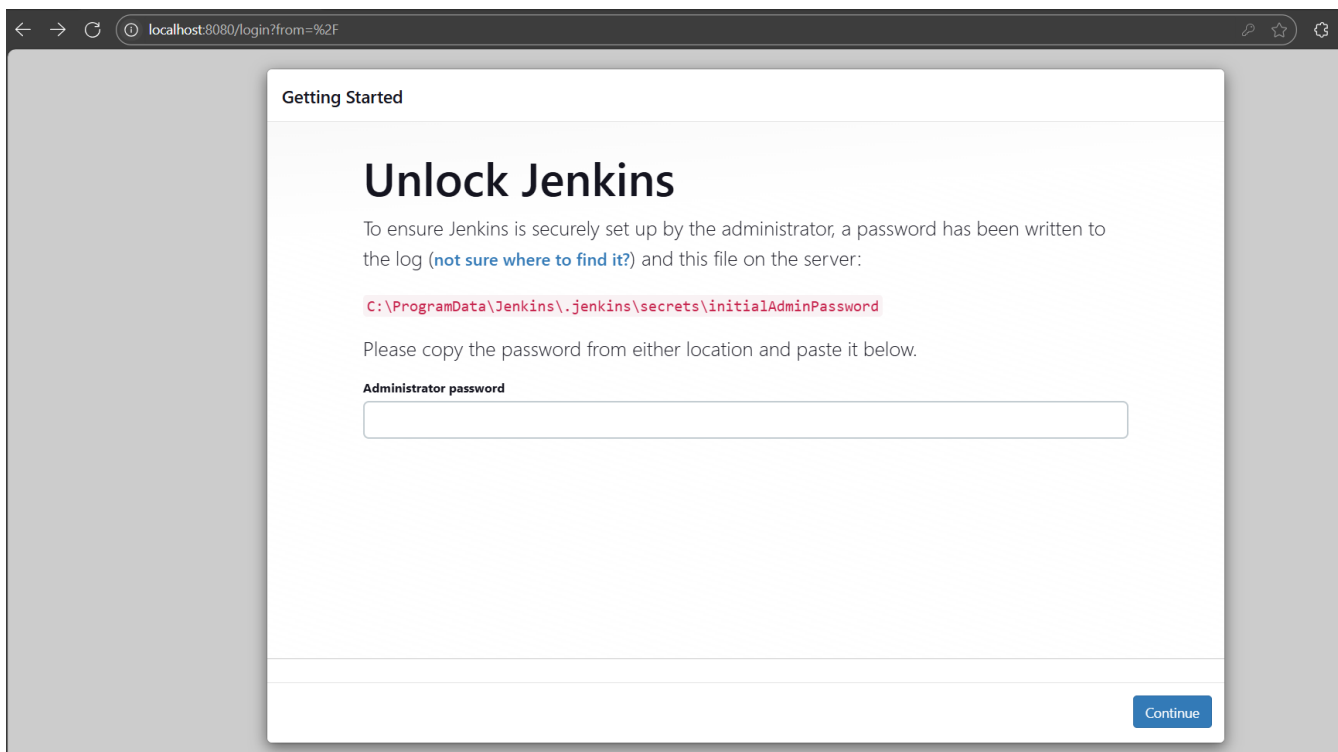




If any popup window prompts for allow the application, click Yes to allow.



Launch browser with localhost:8080

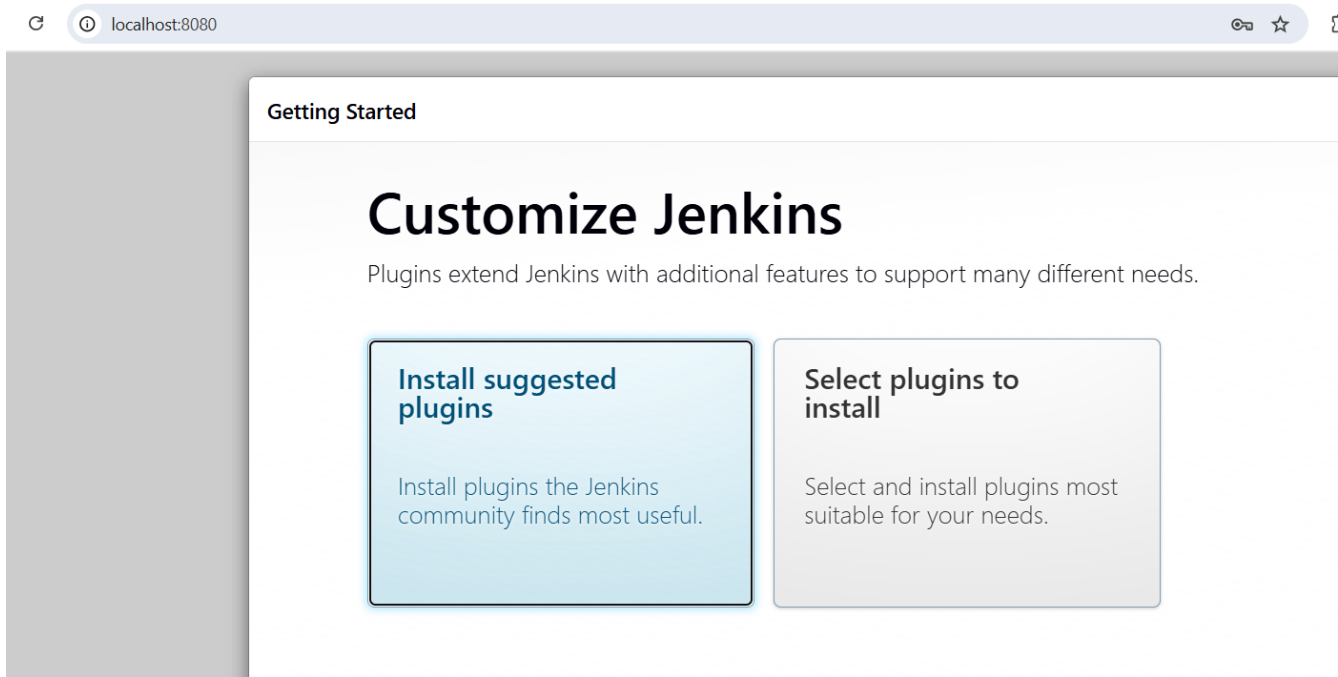


Open the file and copy the password

`C:\ProgramData\Jenkins\.jenkins\secrets\initialAdminPassword`

Uid: admin and the password is copied from the above file.

After Login , click on suggested plugins



Setup new uid and password as admin / admin (or any of your choice)

OR continue as admin

I created admin / admin for easy to remember

A screenshot of the Jenkins 'Create First Admin User' form. The form has the following fields: 'Username' with 'admin' entered, 'Password' with '.....', 'Confirm password' with '.....', 'Full name' (empty), and 'E-mail address' (empty). At the bottom, there is a 'Skip and continue as admin' link and a 'Save and Continue' button. The Jenkins version '2.504.3' is displayed in the bottom left corner.

Save and finish

The screenshot shows the Jenkins 'Getting Started' page for 'Instance Configuration'. The browser address bar shows 'localhost:8080'. The page title is 'Getting Started' and the main heading is 'Instance Configuration'. Under 'Jenkins URL:', there is a text input field containing 'http://localhost:8080/'. Below this, explanatory text states that the Jenkins URL is used for absolute links and that the default value is 'not saved yet'. At the bottom right, there are two buttons: 'Not now' and 'Save and Finish'. The 'Save and Finish' button is highlighted with a yellow circle.

Getting Started

Instance Configuration

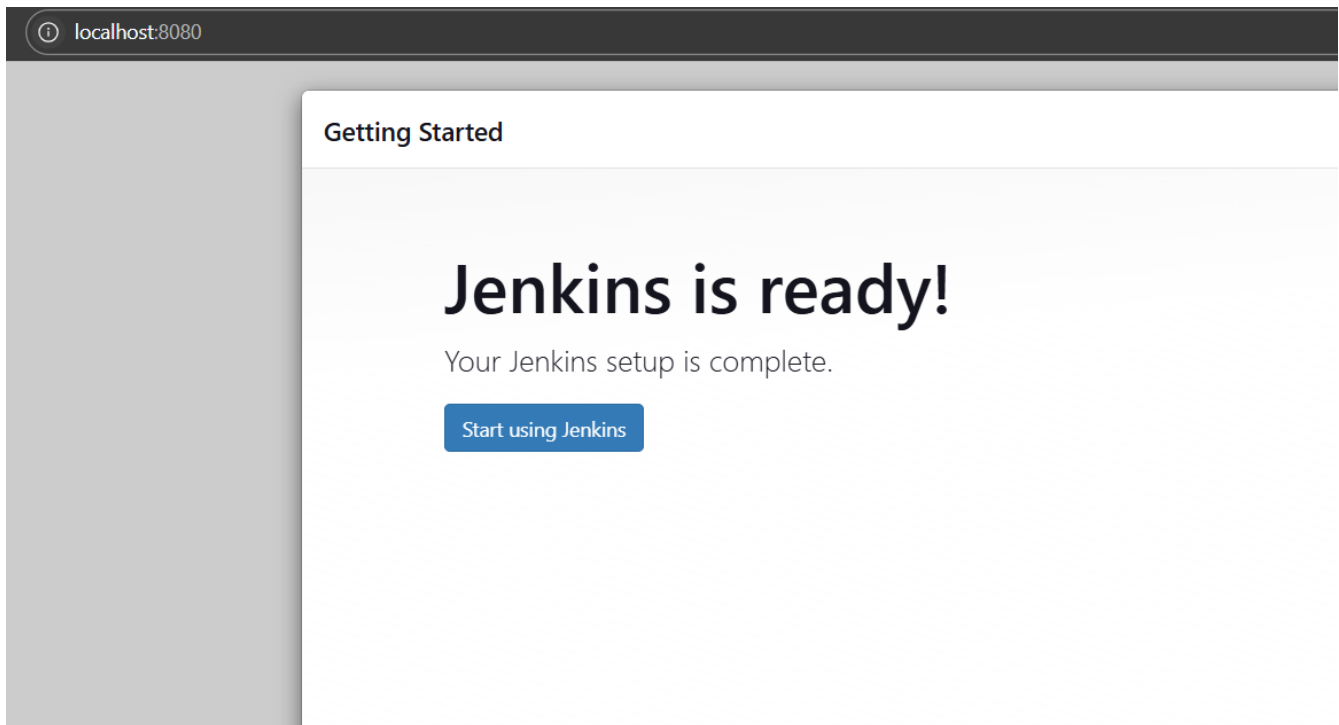
Jenkins URL:

The Jenkins URL is used to provide the root URL for absolute links to various Jenkins resources. That means this value is required for proper operation of many Jenkins features including email notifications, PR status updates, and the BUILD_URL environment variable provided to build steps.

The proposed default value shown is **not saved yet** and is generated from the current request, if possible. The best practice is to set this value to the URL that users are expected to use. This will avoid confusion when sharing or viewing links.

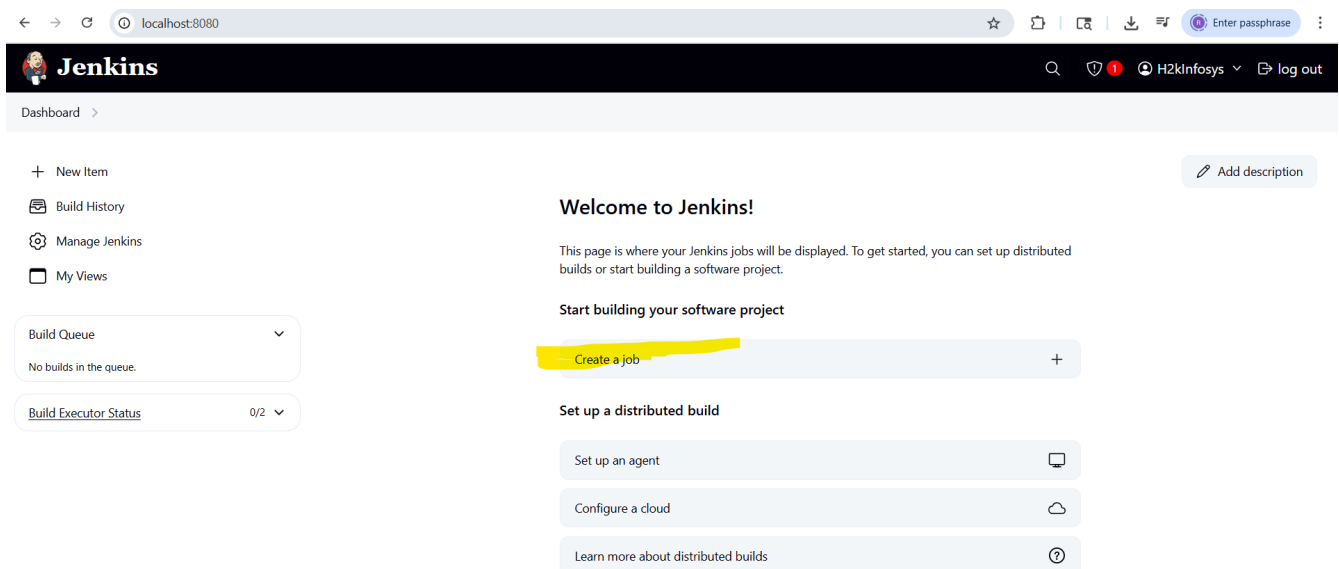
Jenkins 2.504.3

Not now **Save and Finish**



<http://localhost:8080/>

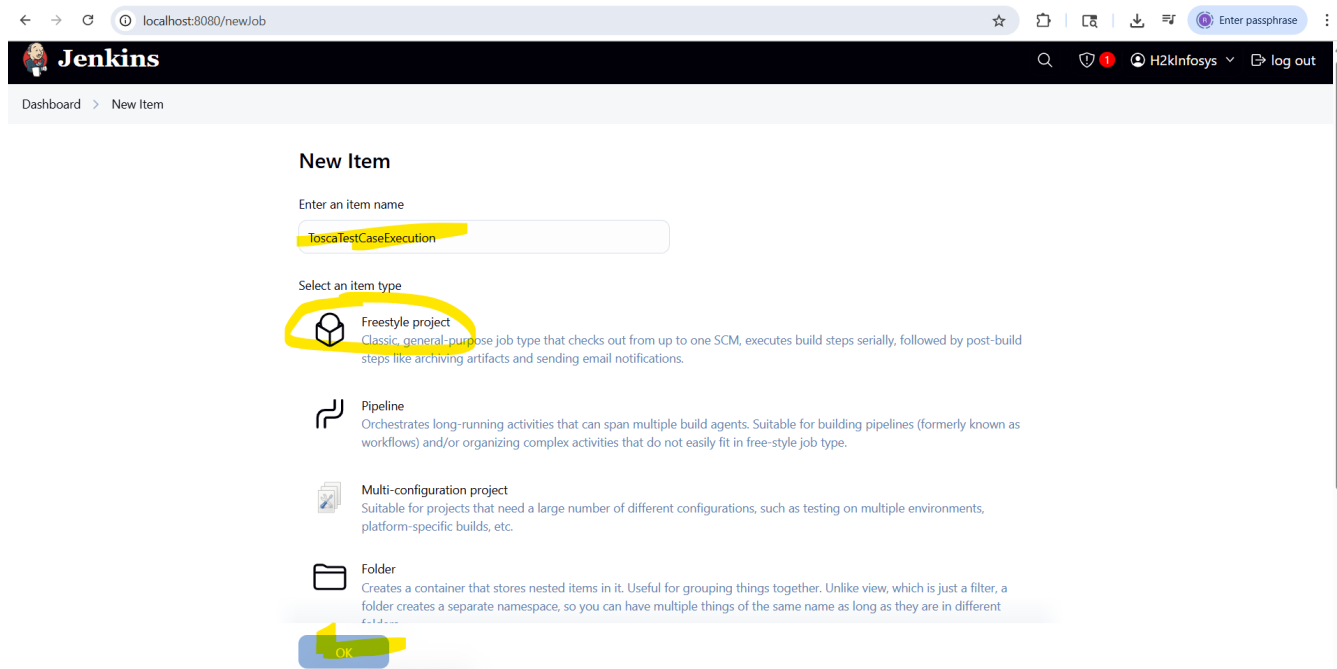
Click on create new job



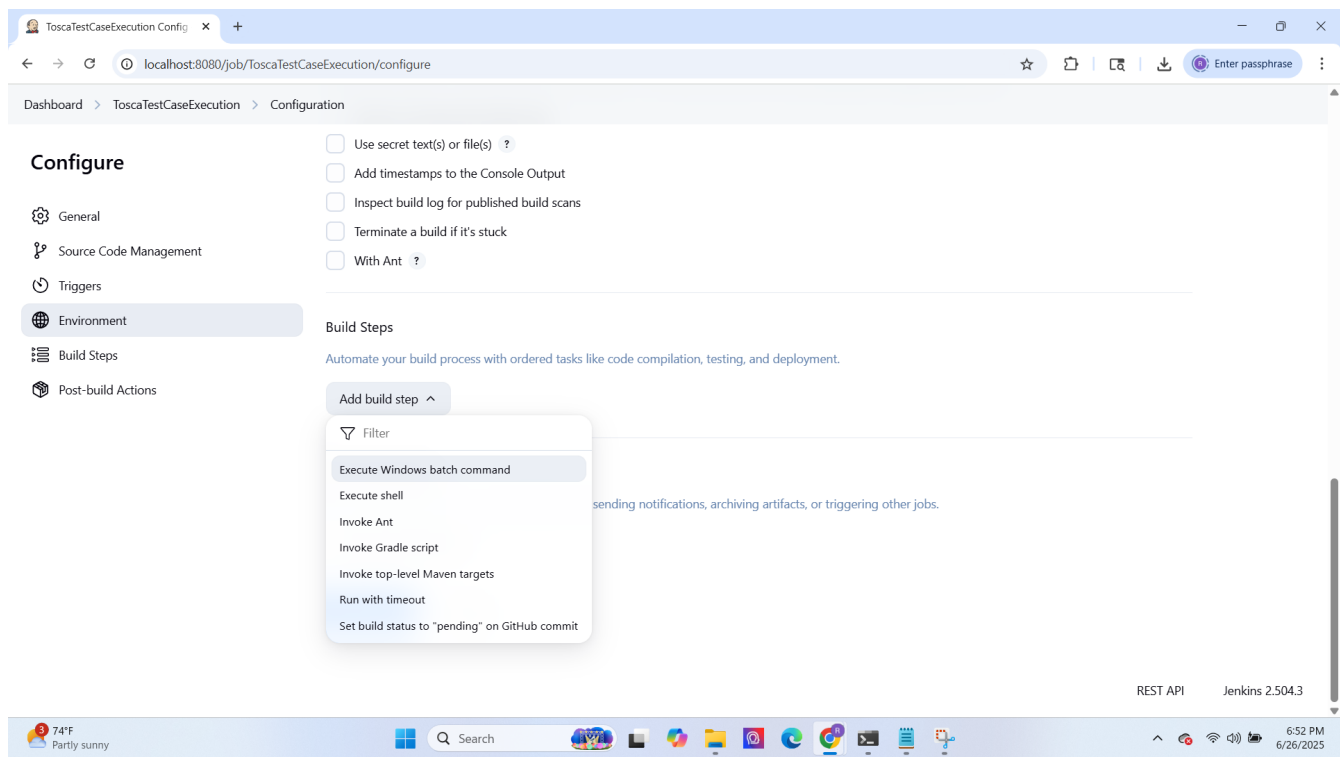
Enter the New Item name (any name)

I want to execute Tosca Regression Test cases , Name it as ToscaTestCases

Select FreeStyle Project and Click OK



Next screen, scroll down and click on Build Steps, select Execute Windows Batch command



Copy and paste below text and update the Workspace name and related folder

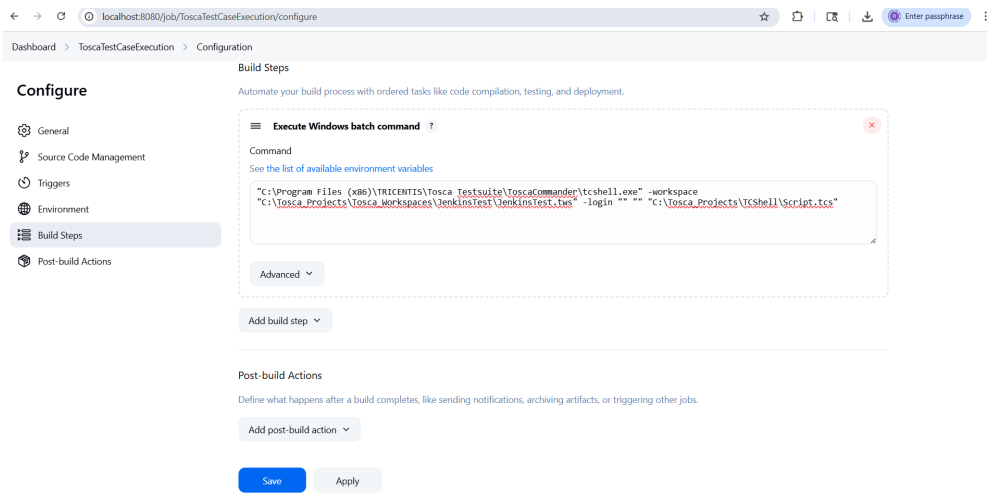
Below example is for a single user workspace . If it is a multi user workspace,need to add a checkout statement in this script.

```
"C:\Program Files (x86)\TRICENTIS\Tosca Testsuite\ToscaCommander\tcshell.exe" -workspace
"C:\Tosca_Projects\Tosca_Workspaces\JenkinsTest\JenkinsTest.tws" -login "" ""
"C:\Tosca_Projects\TCShell\Script.tcs"
```

On your local drive ,Create C:\Tosca_Projects\TCShell\Script.tcs file with below info

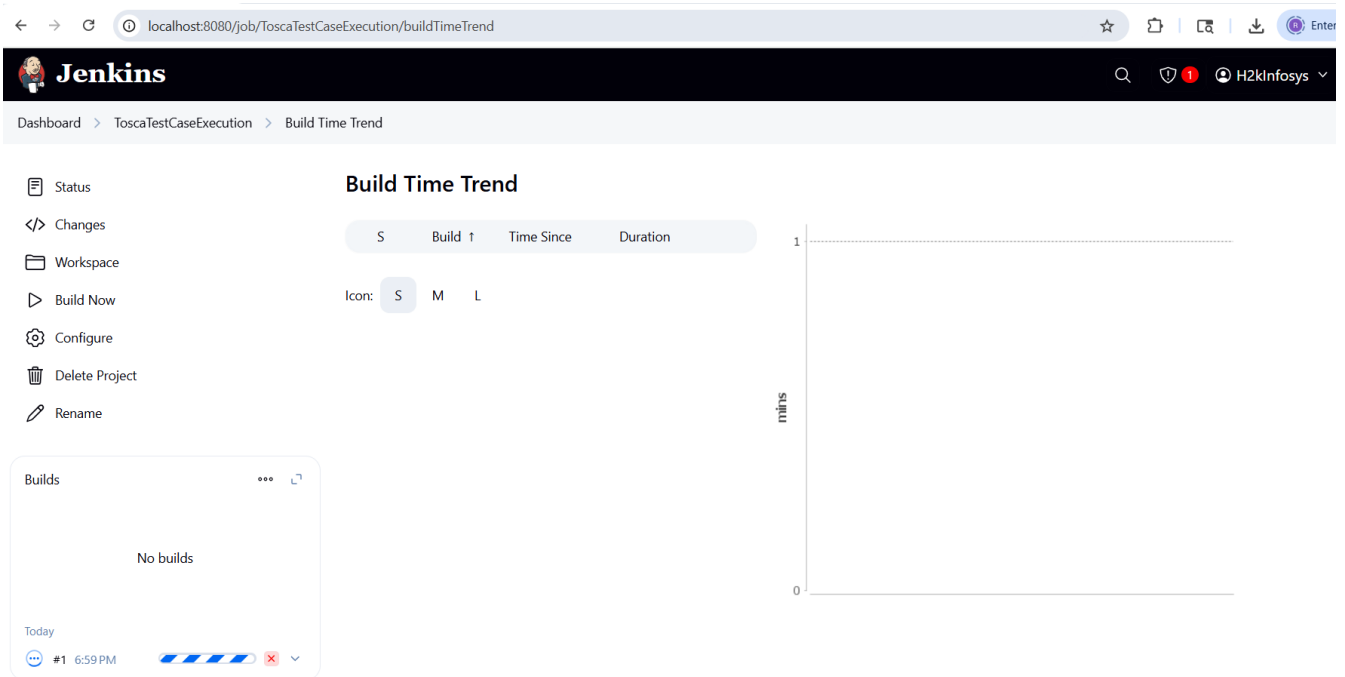
NOTE: replace H2kExe with the name of the execution list folder that you are planning to execute to run the test cases.

```
jumpnode "/Execution/ExecutionLists/H2kExe"
task Run
save
```



++++

Click on Build Now



Click on Down Arrow on Green Tick mark and click on Console Output to view the results.

ToscaTestCaseExecution Build 1

localhost:8080/job/ToscaTestCaseExecution/buildTimeTrend

Jenkins

Dashboard > ToscaTestCaseExecution > Build Time Trend

Status

Changes

Workspace

Build Now

Configure

Delete Project

Rename

Build Time Trend

S Build 1 Time Since Duration

Icon: S M L

mins

1

0

Builds

No builds

Today

#1 6:59 PM

Changes

Console Output

Edit Build Information

Delete build '#1'

Timings

REST API Jenkins 2.504.3

74°F Partly sunny

Search

7:00 PM 6/26/2025

localhost:8080/job/ToscaTestCaseExecution/1/console

Jenkins

Dashboard > ToscaTestCaseExecution > #1 > Console Output

Status

Changes

Console Output

Edit Build Information

Delete build '#1'

Timings

Console Output

Download Copy View as plain text

```

Started by user H2kInfosys
Running as SYSTEM
Building in workspace C:\ProgramData\Jenkins\workspace\ToscaTestCaseExecution
[ToscaTestCaseExecution] $ cmd /c call C:\WINDOWS\TEMP\jenkins10093036734051197933.bat

C:\ProgramData\Jenkins\workspace\ToscaTestCaseExecution>"C:\Program Files (x86)\TRICENTIS\Tosca Testsuite\ToscaCommander\tcshell.exe" -
workspace "C:\Tosca_Projects\Tosca_Workspaces\JenkinsTest\JenkinsTest.tws" -login "" "" "C:\Tosca_Projects\TShell\Script.tcs"
Checking License ... License found!
Run started with TBoxRunTaskExecutor
Verifying required licenses ...
Execution Running ...
Processing execution results ...
Run finalizing ...
Run finished

C:\ProgramData\Jenkins\workspace\ToscaTestCaseExecution>exit 0
Finished: SUCCESS

```

Configure QTest and Import Test Cases

Make sure you have QTest installed and QTest Server URL is available.

Configure and Connect to QTest

https://documentation.tricentis.com/tosca/2420/en/content/qtest_integration/qtest_integration_intro.htm

On Tosca Server, Settings,

Configure QTest integration

You can [integrate Tosca and QTest](#). This allows you to manage your test objectives and releases in QTest, with the results from your automated tests in Tosca.

For detailed information on all setup steps, check out "[Set up the integration](#)".

To get started, fill out these settings:

Property	Description
qTest URL	Enter the URL to your QTest instance.
API Key	Enter your QTest API key. To get the API key, open your QTest project and select  Download QTest Resources . Then go to the Integration with Tosca section and copy the API token.
Activate QTest Integration	Activate the toggle button.

Best Practices , Errors & Solutions

Best Practices:

https://documentation.tricentis.com/tosca/2420/en/content/best_practices/best_practices_overview.htm

Error : Synchronization failed. Please increase your synchronization settings.

Solution: Increase your synchronization settings. Click on the project, click on TBOX Sync and update the numbers. OR Restart Tosca Commander. Close all the applications / browsers etc.

ERROR: Tosca Server might not be working and all the services are down

From Windows Services:

7. Open **Services** from the Windows **Start** menu.
8. Find **Consul.Service** in the list.
9. Right-click on the service name and select **Restart**.

https://documentation.tricentis.com/tosca/2420/en/content/tosca_server/server_service_health_check.htm#:~:text=Right-click%20on%20the%20service%20and%20select%20Restart.

Q&A

Tosca Certification

<https://www.tricentis.com/blog/certified-automation-specialist-tosca>

How to execute the same test case in different browsers at same time?

How to run the test case in headless mode?

Sample Resume

Self healing , Buffer, XB, Waittime, synchronization, Recovery senacior,, sting, math functions, UI testing, api testing, Db testing ,excel TDS, DEX, Jenkins, Tricentis Tosca Server configuration, CI/CD pipeline

- Worked in an Agile environment, attending standup meetings, providing status.
- Reviewed all manual test cases, identified regression test scenarios for automation.
- Developed Hybrid Automation Test framework in Tricentis Tosca with the help of Tosca Commander , Modules, Test Cases, Requirements, Execution etc.
- Using the Advanced XSCAN window, identified duplicate controls, and made them unique by selecting multiple properties. Using ReScan, added new controls to the existing module.
- Applied Self-Healing properties to identify the control if any label etc. updated and avoid failures.
- Implemented Tosca Recovery Scenarios and Clean up Scenarios to reduce the test case failures.
- Implemented Data Driven testing by connecting the excel files, Database, DO Loop, While Loop etc.
- Using Random functions, Math and String functions, Buffers, XB (Exclude Buffers) generated random data while executing data driven test cases.
- Created branches for team members and merged the branches to a common repository.
- Developed and executed API Automation test cases, Validated POST, GET, PUT, DELETE methods, validated JSON request and response payload using Tricentis Tosca.
- Implemented Parallel test execution with the help of DEX env, by updating settings on Tosca Server, and Tosca Commander.
- Distributed test case execution to multiple virtual machines by launching multiple agents.
- Worked with DevOps engineers and provided required information to execute the Tosca automation scripts as part of the CI/CD pipeline.
- Generated authentication token (Client ID , Client Secret) for Tosca Execution Client.
- Validated Tosca automation test results and created defects, reported to developers.

Please write a valuable review

<http://www.facebook.com/H2KInfosysLLC>

<https://goo.gl/oiqSd6>

<https://www.trustpilot.com/review/h2kinfosys.com>

<https://tinyurl.com/h2kinfosysgooglereview>