

Working with Test Cases

Automatic Script Detection

Run-Time Options

Global Setup and Tear Down

Debugging - Commenting Test Cases in Scenario Editor

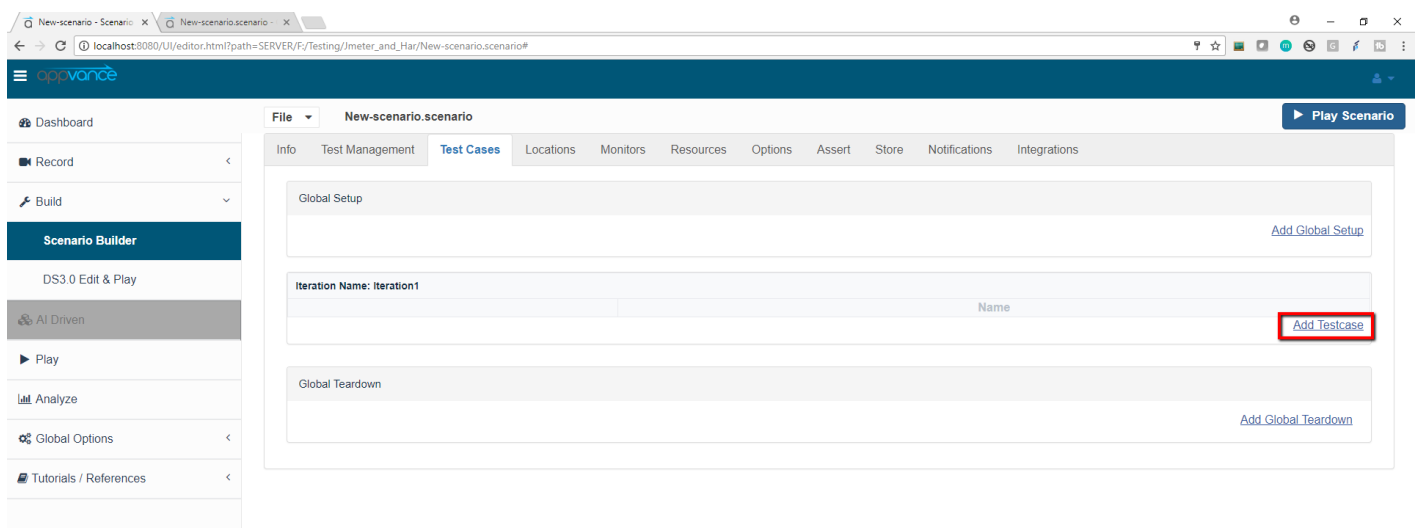
Activate only last execution failures

The **Test Cases** panel is the place where you set up one or more test steps, options for each test step, Distribute users across multiple locations, Set Runtime options like Ramp-up, Ramp-down, Offset, Interval time and Sleep time, and Set Think time options.

This example runs a single test (`AddCompany.ds`) with no setup or teardown.

The Scenario Builder allows for sophisticated scenarios. Unified Test Platform scenarios consist of one or more Test Cases, each of which can contain one or more test scripts. Each Test Case can have setup and teardown scripts. In addition, the entire scenario can have a global setup and global teardown scripts.

To create a Test Case go to the Test Case panel and click 'Add Test Case' including



The setup, teardown, and test elements in the scenario all point to scripts. Appvance IQ knows how to run many kinds of scripts, everything from Appvance Designer Script and Selenium through to JRuby and Perl.

The Scenario Builder shows options for the script that is appropriate to the script type. For example, you get to choose a browser to run an Appvance Designer Script. You get to specify a

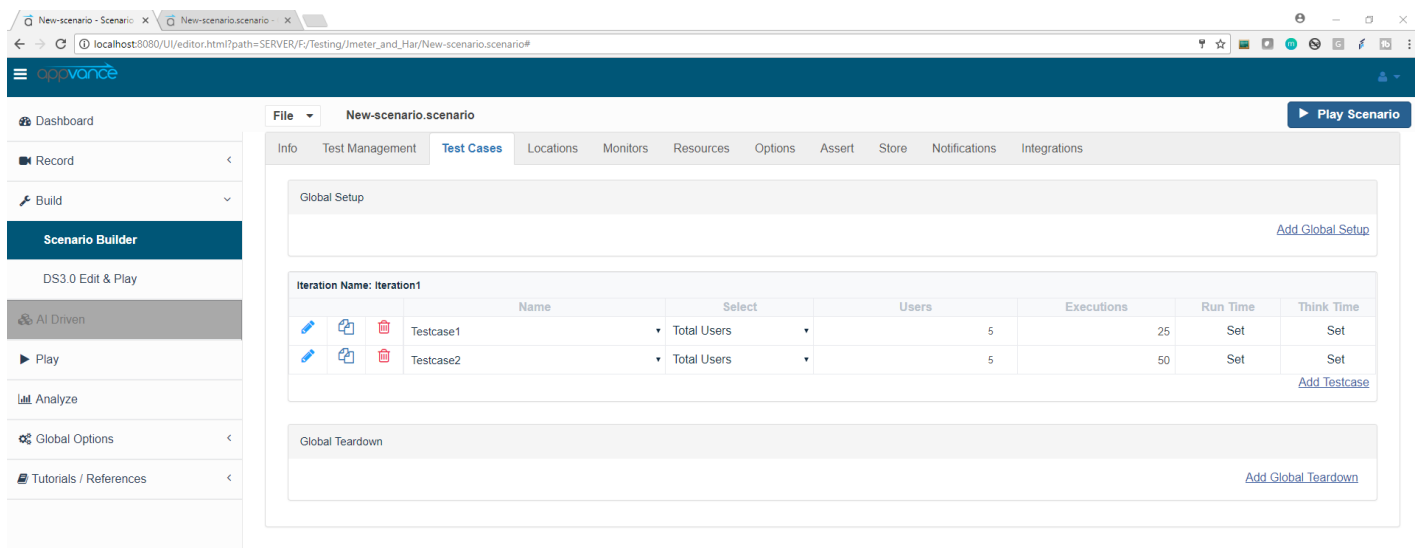
class name for a Java language script.

Automatic Script Detection

In the Test Case panel, when the user clicks on Add a Test Case execution, the user can just browse the file and the script type will be automatically detected and will be auto-populated for the user.

For example:

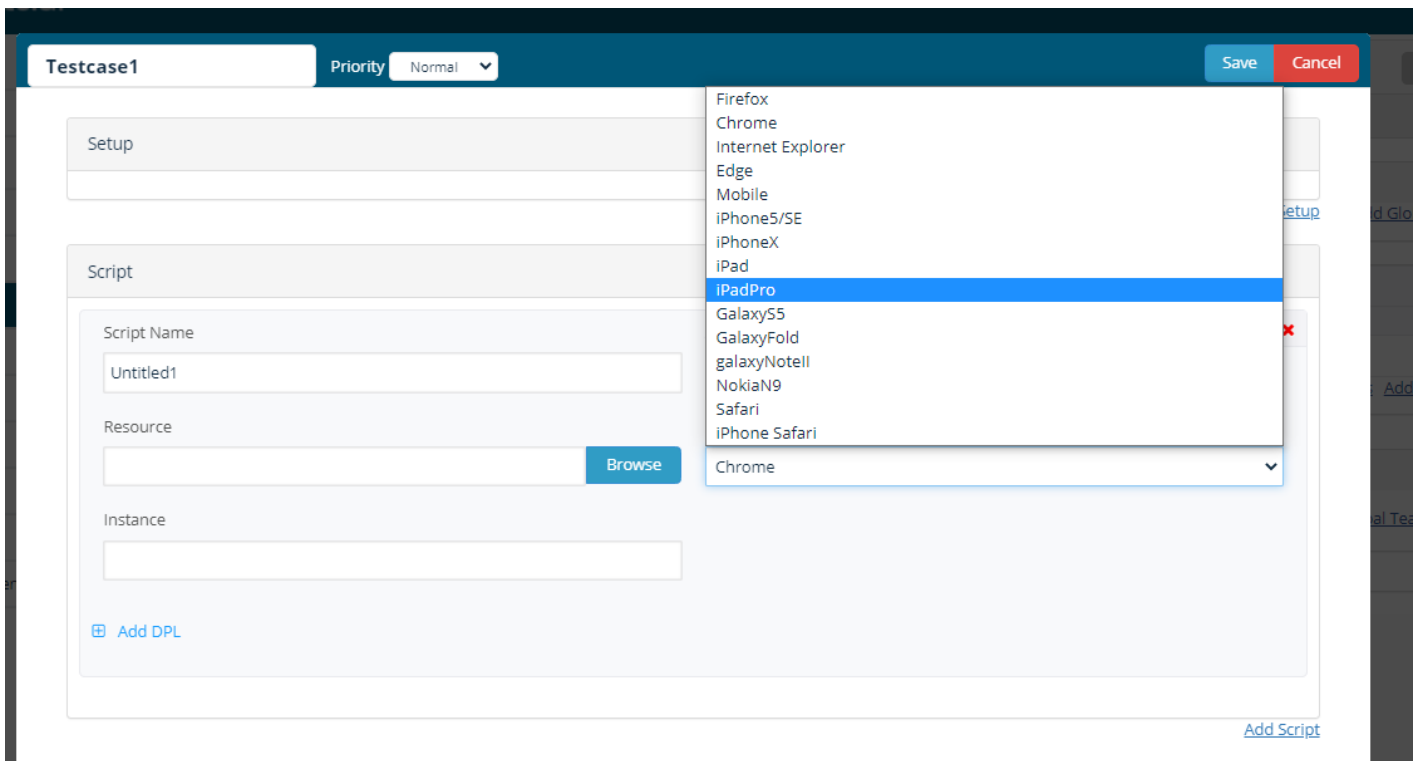
- If a .jmx file is browsed, it will automatically detect that it is a Jmeter script and the test type will be populated.
- If the file extension is a .ds extension it will detect that is a Designer Script and the test type will be populated to the Designer Script.
- .sah will be Sahi scripts etc.



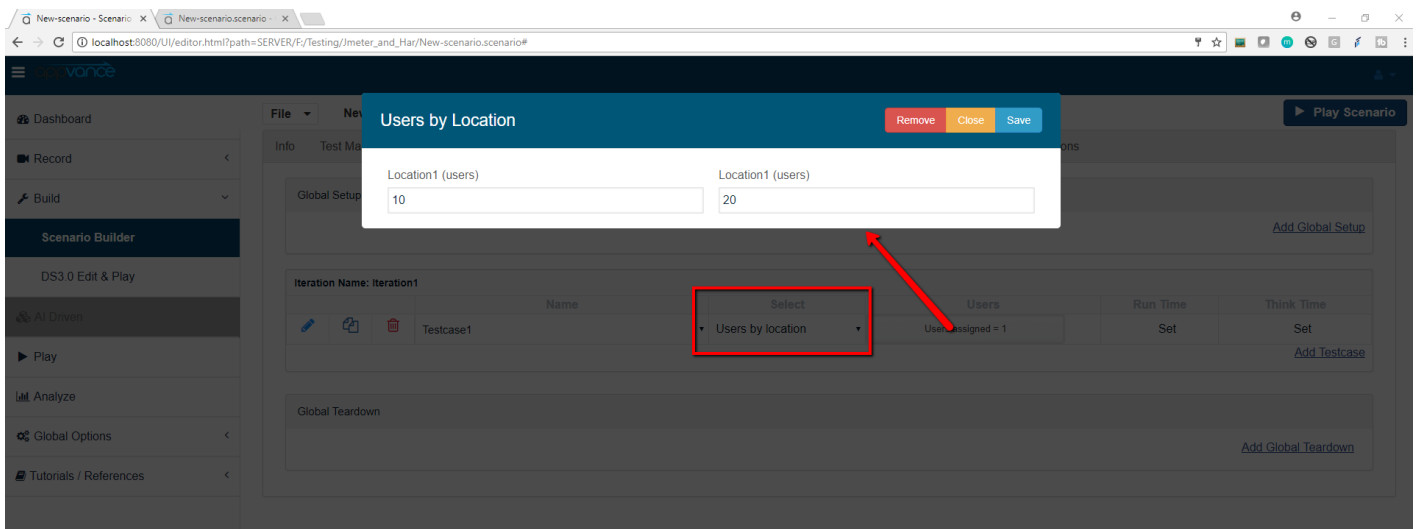
The screenshot displays the Appvance Scenario Builder web application. The interface includes a sidebar on the left with navigation options: Dashboard, Record, Build, Scenario Builder (highlighted), DS3.0 Edit & Play, AI Driven, Play, Analyze, Global Options, and Tutorials / References. The main content area is titled 'New-scenario.scenario' and features a 'Play Scenario' button. Below the title bar, there are tabs for 'Info', 'Test Management', 'Test Cases' (active), 'Locations', 'Monitors', 'Resources', 'Options', 'Assert', 'Store', 'Notifications', and 'Integrations'. The 'Test Cases' tab shows a 'Global Setup' section with an 'Add Global Setup' link. Below this is a table for 'Iteration Name: iteration1' with columns for Name, Select, Users, Executions, Run Time, and Think Time. The table contains two rows: 'Testcase1' and 'Testcase2', both with 'Total Users' selected, 5 users, 25 and 50 executions respectively, and 'Set' for Run Time and Think Time. An 'Add Testcase' link is at the bottom right of the table. Below the table is a 'Global Teardown' section with an 'Add Global Teardown' link.

Iteration Name: iteration1						
	Name	Select	Users	Executions	Run Time	Think Time
  	Testcase1	Total Users	5	25	Set	Set
  	Testcase2	Total Users	5	50	Set	Set

From Scenario Editor, you will have the option to choose chrome responsive modes as playback options including other normal browsers.

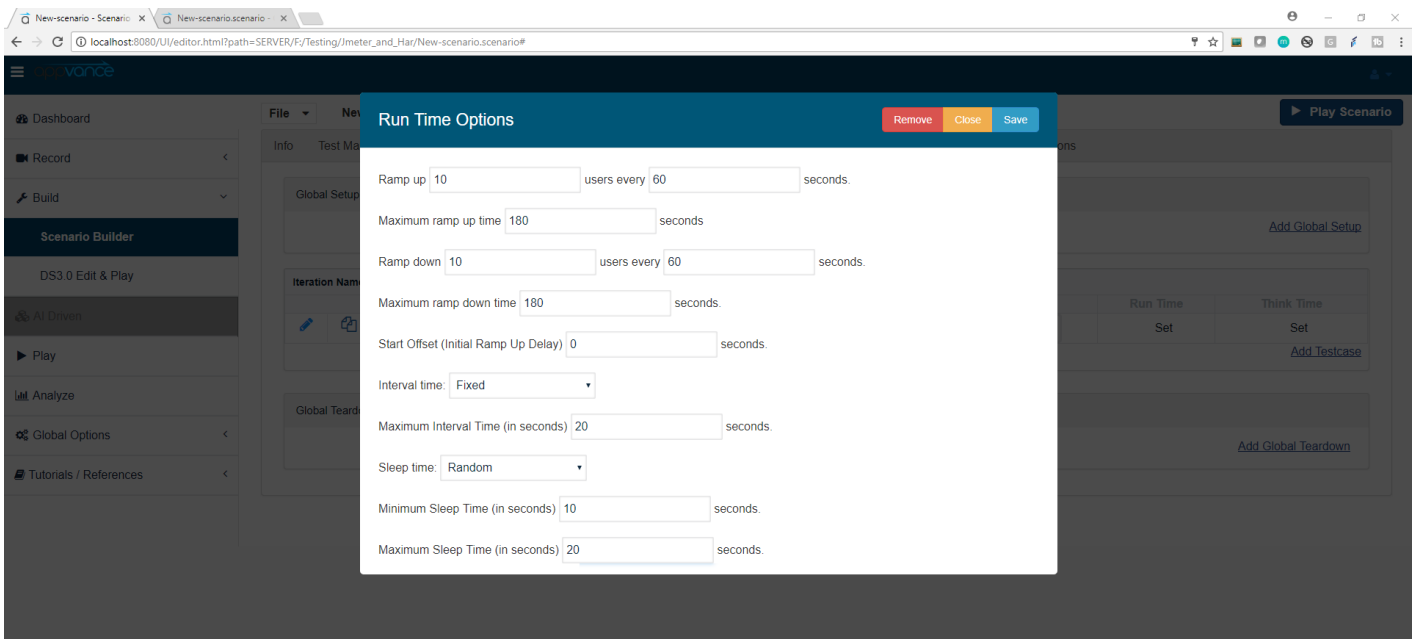


Users: Users can be divided among multiple locations per Test Case, Example: As seen in the below screen-shot, from the 'Select' drop-down Select the option 'Users by location' Users by Location' pop-up will be displayed where users can provide the number of users per location.

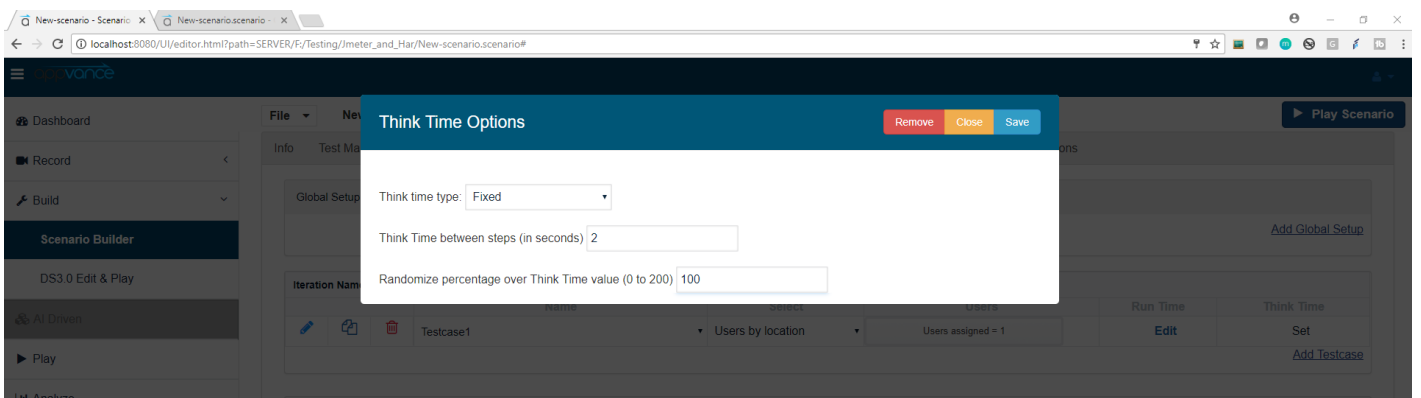


Run-Time Options

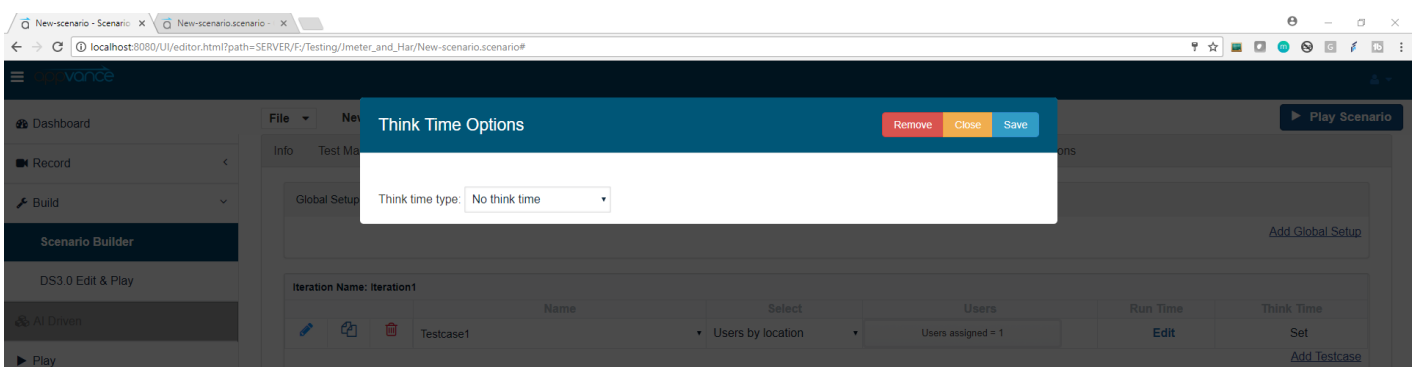
Below are different Run Time Options available.



Fixed think time: The user can provide a fixed think time in seconds. With this user can also provide value for the option 'Randomize percentage over Think Time value (0 to 200) to randomize the think times.



No Think time: No waits statements are executed



For Functional Scenario Test type: Think time options should be provided in the Options Panel.

Also please make sure the option "Report Think Times (Appvance Designer Script only)" is checked as seen in the below screenshot.

Scenario Options

- ☐ Summary steps does not include substeps details.
- ☐ Dynamic IP Address
- ☒ Enable logging (Output, Error, Request, Response, Logs)
 - Save logs for 10 % of transactions
 - ☒ Always save logs (failures only)
 - ☒ Filter response content
 - Don't save response contents greater than 64 Kb
- Sleep Time between test case execution (in seconds) Zero
- Think Time between steps Fixed
 - Fixed Think Time (in seconds) 1
 - Randomize percentage over Think Time value (0 to 200%) 0
- Interval Time between test case execution (in seconds) Zero
- Ramp up 0 users every 0 seconds
- Maximum ramp up time (in seconds) 0
- Ramp down 0 users every 0 seconds
- Maximum ramp down time (in seconds) 0
- ☒ Report Think Times (Designer Script only)
- ☐ Add Think Times to steps

Command Line Options

Random Think Time:

Functional Test				Details
Iteration1				102817
localhost				Working
Testcase1 - 1				102817
1) Untitled1				102817 msec
1.1) wait random 5000 to 6000 ms				5464 msec
1.2) navigateTo('http://demosite.appvance.com')				4800 msec
1.3) wait random 5000 to 6000 ms				5641 msec
1.4) click(link('Bags'))				2689 msec
1.5) wait random 5000 to 6000 ms				5057 msec
1.6) click(link('Mugs'))				4846 msec
1.7) wait random 5000 to 6000 ms				5689 msec
1.8) click(link('Clothing'))				3642 msec
1.9) wait random 5000 to 6000 ms				5208 msec
1.10) setValue(searchbox(0) 'Ruby')				192 msec
1.11) wait random 5000 to 6000 ms				5380 msec
1.12) click(submit(0))				2525 msec
1.13) wait random 5000 to 6000 ms				5841 msec
1.14) click(span('Ruby on Rails Baseball Jersey'))				6238 msec
1.15) wait random 5000 to 6000 ms				5158 msec
1.16) click(radio(3))				54 msec
1.17) wait random 5000 to 6000 ms				5299 msec
1.18) click(submit('Add To Cart'))				11172 msec
1.19) wait random 5000 to 6000 ms				5217 msec

In the controller window, observe that user-added WAIT will not be changed, new WAIT statements will only be added to steps that do not already have one.

Fixed Think Time:

New-scenario.scenario
Tue Oct 17 17:31:18 IST 2017 [Edit Scenario](#)

Status: Execution done | Iteration: Iteration1 | Succeeded: 1 | Failed: 0 | [Play Scenario](#)

Output | **Live Reports** | Test case(s) Executions | Testnode | Active Virtual Users | Validation Suite | Autocorrelation

Functional Tree Depth: Steps

Functional Test		Details
Iteration1		102817
localhost		Working
Testcase1 - 1		102817
1) Untitled1		102817 msec
1.1) wait random 5000 to 6000 ms		5464 msec
1.2) navigateTo("http://demosite.appvance.com")		4800 msec
1.3) wait random 5000 to 6000 ms		5641 msec
1.4) click(link("Bags"))		2689 msec
1.5) wait random 5000 to 6000 ms		5057 msec
1.6) click(link("Mugs"))		4846 msec
1.7) wait random 5000 to 6000 ms		5689 msec
1.8) click(link("Clothing"))		3642 msec
1.9) wait random 5000 to 6000 ms		5208 msec
1.10) setValue(searchbox(0) "Ruby")		192 msec
1.11) wait random 5000 to 6000 ms		5380 msec
1.12) click(submit(0))		2525 msec
1.13) wait random 5000 to 6000 ms		5841 msec
1.14) click(span("Ruby on Rails Baseball Jersey"))		6238 msec
1.15) wait random 5000 to 6000 ms		5158 msec
1.16) click(radio(3))		54 msec
1.17) wait random 5000 to 6000 ms		5299 msec
1.18) click(submit("Add To Cart"))		11172 msec
1.19) wait random 5000 to 6000 ms		5217 msec

Appvance Designer Script (using Convert to Think Time) with Fixed and Random Think Time options:

Create a scenario with Appvance Designer Script that has user added WAIT statement which is converted to Think Time and the Think Time option in Scenario Builder is set to Fixed / Random. Fixed and Random will change script WAIT (User added WAIT) statement which is converted to Think Time and add preceding WAIT statements to all steps that do not already have one. **For Example:** Consider an Appvance Designer Script that has user added WAIT statement which is converted to Think Time. Create a Scenario with this Appvance Designer Script and set Fixed to think time option “1000” milliseconds or Random Think Time option to “500 to 600” milliseconds. Save and play the scenario.

Appvance Designer Script:

File Options **demositeappvance.ds** [Record] [Play] [Stop] [Pause] [Step Over]

No	Action	Accessor	Value	Description
1	navigateTo		"http://demosite.appvance.com"	
2	click	link("Bags")		
3	click	link("Mugs")		
4	click	link("Clothing")		
5	setValue	searchbox(0)	"Ruby"	
6	wait		1000	
7	click	submit(0)		
8	click	in("Ruby on Rails Baseball Jersey")		
9	click	radio(3)		
10	click	submit("Add To Cart")		
11	click	submit("Checkout")		

Context menu options: Cut, Copy, Paste below, Delete, Set Breakpoint, Add New Step Below, **Convert to Thinktime** (highlighted), Comment, Undo, Refactor.

Random Think Time:

- 2 Setup calls sharing an instance in DS3, they will share browser

Script Name

Global Setup1

Script Type

Test Designer

Resource

GlobalSetup1.js

Browse

Browser

Firefox

Instance

globalBrowser

Add DPL

Add Argument

Script Name

Global Setup2

Script Type

Test Designer

Resource

GlobalSetup2.js

Browse

Browser

Firefox

Instance

globalBrowser

Add DPL

Add Argument

Test Case setup and Tear-down:

- Any scenario can have multiple global setups and tear-down calls.
 - A failing Test Case setup will fail the scenario
 - A failing Test Case tear-down won't fail the scenario
 - Test Case setup and Test Case tear-down can share instances.
 - In the case of Test Designer, if a Test Case setup and Test Case tear-down share an instance, the Browser will be kept open during the iteration, EVEN if the reuse browser on every execution is not set
- Command Line

Test Designer

☐ Fail step when getting failing http status codes

☒ Reuse browser every execution

☐ Take Snapshot on failure

- A script with one setup, 2 scripts, and one tear-down, sharing the instance:

Setup

Setup Name

Setup

Setup Type

Test Designer

Resource

Setup.js

Browse

Browser

Firefox

Instance

localBrowser

Add DPL

Add Setup

Script

Script Name

Untitled1

Script Type

Test Designer

Resource

Test1.js

Browse

Browser

Firefox

Instance

localBrowser

Add DPL

Add Script

Script

Script Name

Untitled1

Script Type

Test Designer

Resource

Test2.js

Browse

Browser

Firefox

Instance

localBrowser

Add DPL

Add Script

Teardown

Teardown Name

Teardown

Teardown Type

Test Designer

Resource

Teardown.js

Browse

Browser

Firefox

Instance

localBrowser

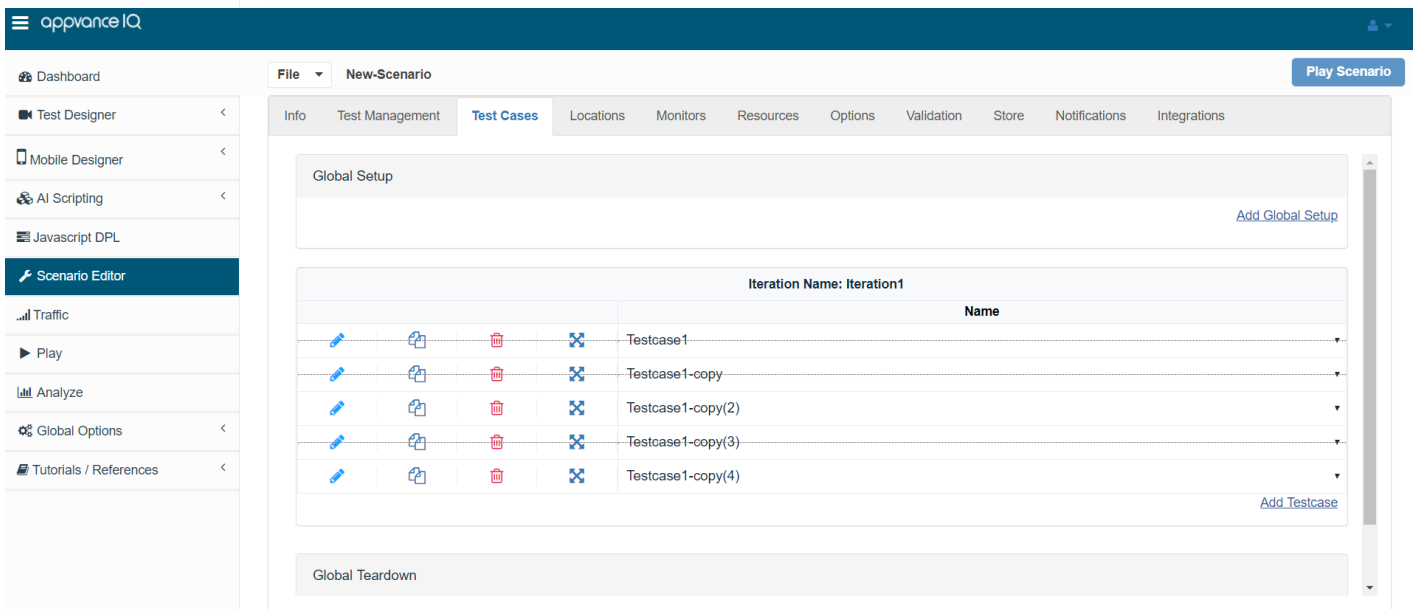
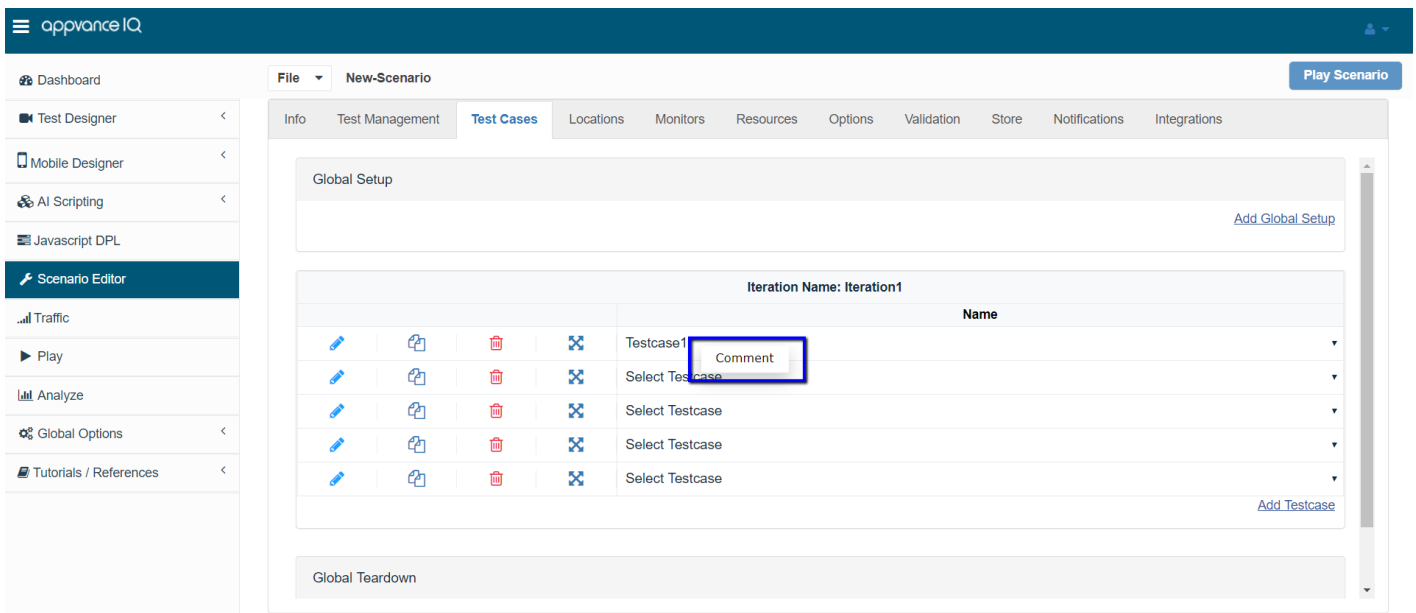
Add DPL

Add Teardown

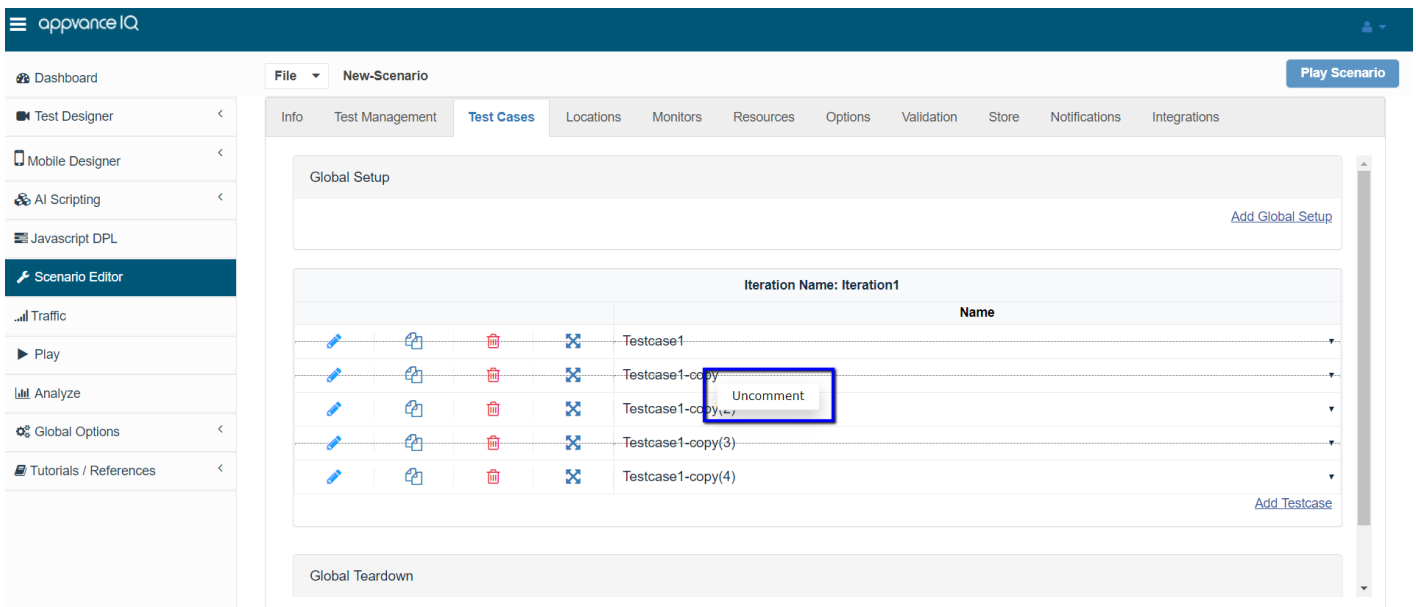
Debugging - Commenting Test Cases in Scenario Editor:

For debugging purposes, it is always a good option to comment out Test Cases in Scenario Editor, that way all those commented Test Cases would not run

1. To do so, open the scenario in the scenario editor with all the Test Cases defined
2. Right-click any of the Test Cases that you want to comment on
3. Click Comment



Similarly, for un-commenting, right-click the commented Test Case and click uncomment to uncomment the selected Test Case.



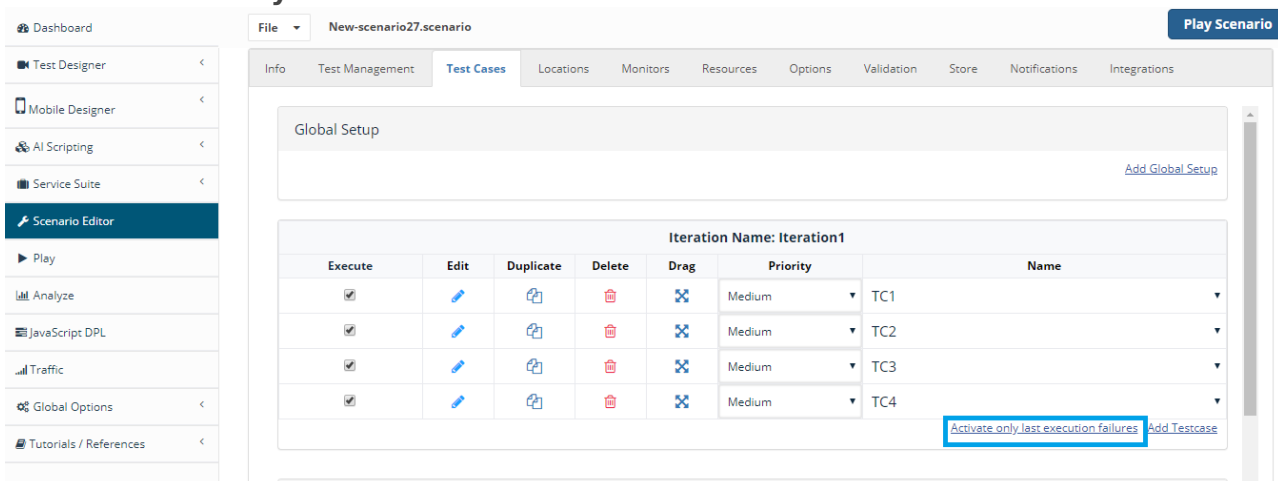
Activate only last execution failures

Given a Scenario with 100 Test Cases, upon executing them, 95 of them passed and 5 of them failed (Random Test Cases) due to several reasons, data issues, application issues, application not responding due to multiple threads running in parallel to run the Test Cases, etc. What if there was an option to execute only those failed 5 Test Cases from Scenario itself instead of checking them in Test Designer or creating a new scenario with only those 5 Test Cases (after identifying them), Activate only the last execution failures makes life easier, in this case, this option just picks those failed Test Cases from the last run, and runs only them to provide a result.

Once a scenario is executed and encountered failures, for you to enable the option follow the below steps;

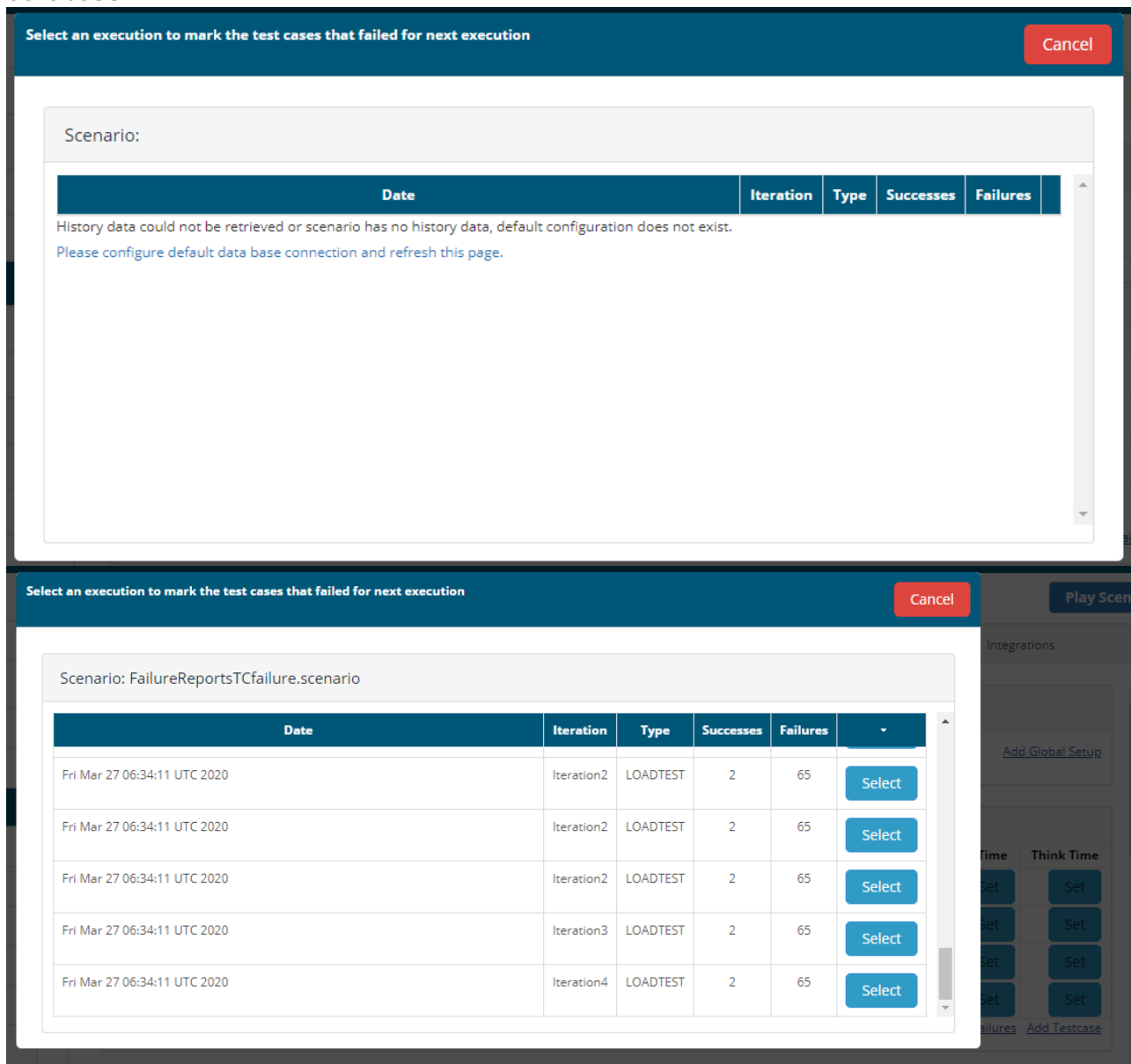
Note: This is more of a functional test scenario and not used during load or performance testing.

1. Go to Scenario and open the last played scenario.
2. Go to the Test Cases tab that will list all the defined Test Cases under each of the iterations.
3. Click **Activate only last execution failures** link

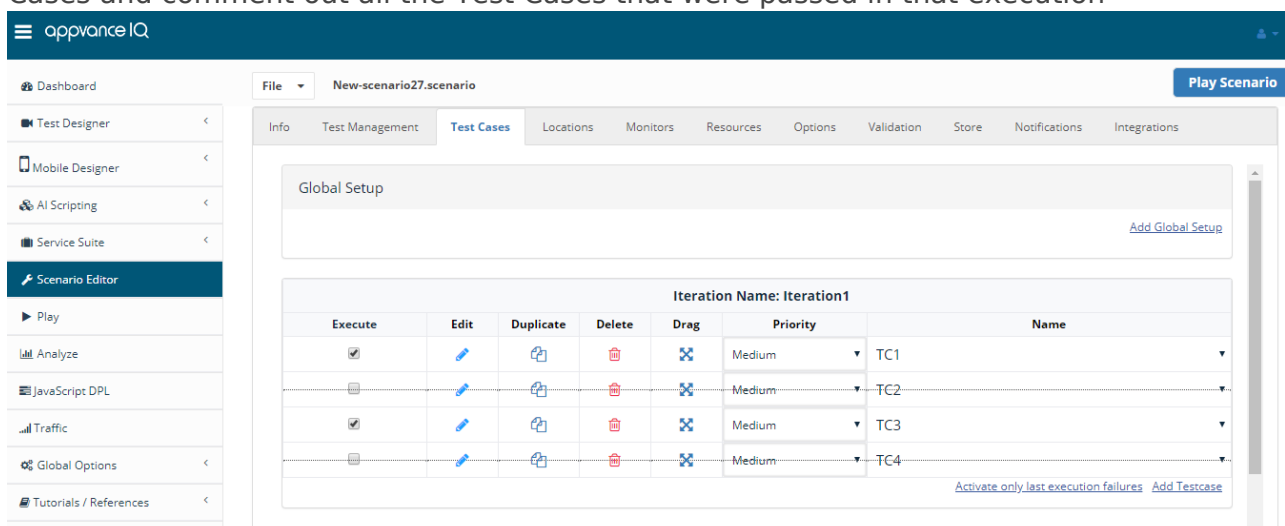


4. This will open a pop up showing all the prior execution with the number of successes and failures, if the running AIQ is not connected to a database, you will have a link that will

navigate to the results page under the global options > Admin options to configure the database



5. From the pop-up, you can choose the recent execution or choose from an old execution as well
6. Upon choosing the execution, the feature would automatically select all the failed Test Cases and comment out all the Test Cases that were passed in that execution



This way you can enable execution to re-run scenarios with failed only Test Cases

Revision #2

Created 2026-07-31 14:12:52 UTC by Admin

Updated 2026-07-31 14:15:36 UTC by Admin