

Menu

quiliup version 2.6.3.4

Try all this:

Arguments:

-a or --authorization	User authorization.
-u or --url	Hostname and port eg. https://<hostname>:<port>.
-ad	Active Directory mode.
-krb	kerberos authentication.

Single run options:

-t or --test	Test ID to run.
-f or --flow	Execution flow ID to run.

Multiple run options:

-serial	Run objects one by one.
-p or --project	Project name that store all object that you want to run.
-tag	Tag name.
-limit	Limit operation to single type, e.g -limit "-t" - will run only tests objects in the run.
-allTags	Run all objects that are related to all the tags specified.

Option Configuration:

-s or --sample	Run in test mode.
-v or --variable	External variable eg. '<name>:<value>'. -h or --help Show the help page (this message) and exit.
-d or --debug	Debug mode.

Linux

Pre Requirements:

- 1.curl installed
- 2.execution permissions
- 3.full permissions 777 to run_qp.sh file
4. in case of getting the following error message:
/bin/bash^M: bad interpreter: No such file or directory
run the following command:
sed -i -e 's/\r\$//' run_qp.sh

Examples:

Help:

```
sh ./run_qp.sh -h
```

Run test with variable:

```
./run_qp.sh -u "https://192.168.50.177:8443" -a "UUFfdXNlckBRQToxMjM=" -t "513" -v "x:1"
```

Run test in Active Directory mode

```
./run_qp.sh -u "https://192.168.50.156:8443" -a 'devdomain\moran:Qw!23456' -ad -d -t "502"
```

Run Execution Flow in Active Directory mode, Kerberos*

```
./run_qp.sh -u "https://192.168.50.177:8443" -a 'devdomain\moran:Qw!23456' -krb -f "66" -d
```

Run execution flow in debug mode with variable:

```
./run_qp.sh -u "https://192.168.50.177:8443" -a "UUFfdXNlckBRQToxMjM=" -f "69" -v "x:1" -d
```

Run multiple objects , of type test, in parallel

```
./run_qp.sh -u "https://192.168.50.177:8443" -a "UUFfdXNlckBRQToxMjM=" -p "project1,project2" -tag "t1,t2" -limit "t"
```

Run multiple objects, of type execution flow, in serial

```
./run_qp.sh -u "https://192.168.50.177:8443" -a "UUFfdXNlckBRQToxMjM=" -p "QVD" -tag "moran" -limit "f" -serial
```

Run multiple objects, of test and execution flow, in serial

```
./run_qp.sh -u "https://192.168.50.177:8443" -a "UUFfdXNlckBRQToxMjM=" -p "QVD" -tag "moran" -serial
```

Windows

Pre Requirements:

1. .Net 3.5 Framework
2. Copy the file run_qp.exe to a certain location the windows server, the exe file is located in the tomcat (Webapps->ROOT->Utils)

Examples:

Help:

```
run_qp.exe -h
```

Run test:

```
run_qp.exe -u "https://localhost:8443" -a "UUFfdXNlckBRQToxMjM=" -t "815" -v "x:1"
```

Run test in Active Directory mode

```
run_qp.exe -u "https://localhost:8443" -ad -t "815"
```

Run execution flow in debug mode:

```
run_qp.exe -u "https://localhost:8443" -a "UUFfdXNlckBRQToxMjM=" -f "92"
```

Run multiple objects in parallel

```
run_qp.exe -u "https://localhost:8443" -a "UUFfdXNlckBRQToxMjM=" -p "QVD" -tag "moran" -limit "t" -d
```

Run multiple objects in serial

```
run_qp.exe -u "https://localhost:8443" -a "UUFfdXNlckBRQToxMjM=" -p "QVD" -tag "moran" -limit "t" -serial -d
```

Run multiple objects in serial

```
run_qp.exe -u "https://localhost:8443" -a "UUFfdXNlckBRQToxMjM=" -p "QVD" -tag "t1" -tag "t2" -limit "t" -allTags -d
```

*If you use a Windows SSPI-enabled curl binary and perform Kerberos V5, Negotiate, NTLM or Digest authentication then you can tell curl to select the user name and password from your environment by specifying a single colon with this option: "-a :".

Command line execution statuses from quilliup

- (-3) Calculating*
- (-2) Pending*
- (-1) Running*
- (0) Passed*
- (1) Failed*
- (2) Warning*
- (3) Wrong Query*
- (4) Bad Configuration*
- (5) Cancelled by User*
- (6) System Failure*
- (7) Disabled Execution*
- (8) All Objects are Disabled*
- (9) Cancelled by Shutdown*
- (10) Cancelled by timeout*
- (11) Cancelled by internal execution*

Command line error statuses from run qp (not from quilliup)

Windows

- (13) Missing parameters
- (14) Request to the server failed

Linux

- (13) HTTP Status 401 - Bad credentials

(14) Missing argument: --test / --flow

(15) Missing argument!- URL or Authentication