

API Keywords for Jmeter

\api\fileManagement		
Keyword	Description	Condition
SetPathPermission	Given variables (path, user and permission) When setUpData contains permission Then update only permission of user	<pre> \$__groovy(vars.get("setUpData")!=null && vars.get("setUpData").toLowerCase().split("\\,").toString().trim().contains("path") && vars.get("setUpData").toLowerCase().split("\\,").toString().trim().contains("user") && vars.get("setUpData").toLowerCase().split("\\,").toString().trim().contains("permission")) </pre>
createFiles	Create file conform corresponding extension in variable "path"	<pre> \$__groovy(vars.get("path")!=null && vars.get("path").toLowerCase().trim().contains(".xml")) </pre>
createPath	Given variables (setUpData, path) When setUpData contains path Then delete Path if exists And create File if path contains "." dot char And create Directory if path dont contains "." dot char	<pre> \$__groovy(vars.get("setUpData") && vars.get("path") != null && vars.get("path").length() > 0 && vars.get("setUpData").toLowerCase().split("\\,")*.trim().contains("path") && !["public\\,home"].contains(vars.get("path")) // dont override default directories) </pre>
deletePathPermanently	Required variable: path	<pre> \$__groovy(vars.get("setUpData")!=null && vars.get("setUpData").toLowerCase().split("\\,")*.trim().contains("path") && vars.get("path")!= null && ![":\\,%3a\\,home\\,%3ahome\\,public\\,%3apublic\\,home:admin\\,%3ahome%3aadmin\\,home:suzy\\,%3ahome%3asuzy\\,home:pat\\,%3ahome%3apat\\,home:tifany\\,%3ahome%3atifany"].contains(vars.get("path").toLowerCase().trim())) </pre>
emptyTrash	Permanently deletes all items in the trash.	<pre> \$__groovy(vars.get("trashedObjectId_ALL") != null) </pre>
overwritePathPermissions	Given variables (path, user and permission) When setUpData contains permission Then overwrite all permissions path to user and Admin	<pre> \$__groovy(vars.get("setUpData")!=null && vars.get("setUpData").toLowerCase().split("\\,")*.trim().containsAll(["path\\,user\\,permission"])) </pre>

Keyword	Description	Condition
setACL	<p>Conditions setUpData contains "setACL"</p> <p>Parameters userPathPermission, format example ("user:read,write user2:read,write,delete user3: ") rolePathPermission, format example ("role:read,write role2:read,write,delete")</p> <p>Output Will remove, add or update the related recipient (user/role) with permissions (if permissions null delete recipient)</p>	<pre>\$_groovy(vars.get("setUpData") && vars.get("path") != null && vars.get("setUpData").toLowerCase().split("\",").trim().contains("setacl") && (vars.get("rolePathPermission") != null vars.get("userPathPermission") != null))}</pre>
setRolePathPermission	<p>Given variables (path, role, setUpData and pathPermission) And pathPermission with some of values (read,write,delete,manage,all or empty/null) When setUpData contains setUserPathPermission Then update only path permission of role</p>	<pre>\$_groovy(vars.get("setUpData") && vars.get("path") && vars.get("role") != null && vars.get("setUpData").toLowerCase().split("\",").trim().contains("setrolepathpermission") && ["read","write","delete","manage","all",""].containsAll(vars.get("pathPermission").toLowerCase().split("\",").trim()))}</pre>
setUserPathPermission	<p>Given variables (path, user, setUpData) And pathPermission with some of values (read,write,delete,manage,all) When setUpData contains setUserPathPermission Then update only path permission of user</p>	<pre>\$_groovy(vars.get("setUpData") && vars.get("path") && vars.get("user") != null && vars.get("setUpData").toLowerCase().split("\",").trim().contains("setuserpathpermission") && ["read","write","delete","manage","all",""].containsAll(vars.get("pathPermission").toLowerCase().split("\",").trim()))}</pre>

\api\userRoleManagement\userRolesPentahoSecurity

Keyword	Description	Condition
assignRoleToUser	<p>Given variables (setUpData, role, user) When setUpData contains "role" and "user" And user and role don't is null Then assign role to user</p>	<pre>\$_groovy(vars.get("setUpData") && vars.get("role") && vars.get("user") !=null && vars.get("setUpData").toLowerCase().split("\",").trim().containsAll(["role","user"]))}</pre>

Keyword	Description	Condition
createRole	Given variables (setUpData, role, rolePermission) When setUpData contains role And role is not null And role is not default role ["Administrator", "Power User", "Report Author", "Business Analyst", "Anonymous"] Then delete role if exists And create role And assign rolePermission to role	<pre> \$__groovy(vars.get("setUpData") && vars.get("role") != null && vars.get("setUpData").toLowerCase().split(",")*.trim().contains("role") && !["administrator", "power user", "report author", "business analyst", "anonymous"].contains(vars.get("role").trim().toLowerCase())) </pre>
createUser	Given variables (setUpData, user) When setUpData contains "user" And "user" is not null And "user" is not "default user" [admin,suzy,pat,tiffany] Then delete user if exists And create user	<pre> \$__groovy(vars.get("setUpData") && vars.get("user") != null && vars.get("setUpData").toLowerCase().split(",")*.trim().contains("user") && !["admin", "suzy", "pat", "tiffany"].contains(vars.get("user").trim().toLowerCase())) </pre>
deleteRole	Given variables (setUpData, role) When setUpData contains "role" And role is not null And role is not default ["Administrator", "Power User", "Report Author", "Business Analyst", "NoPermissions"] Then Delete role	<pre> \$__groovy(vars.get("setUpData") && vars.get("role") != null && vars.get("setUpData").toLowerCase().split(",")*.trim().contains("role") && !["administrator", "power user", "report author", "business analyst", "anonymous"].contains(vars.get("role").trim().toLowerCase())) </pre>
deleteUser	Given variables (setUpData, user) When setUpData contains "user" And user is not null And user is not default user [admin,suzy,pat,tiffany] Then delete user And delete user default directory if exists	<pre> \$__groovy(vars.get("setUpData") && vars.get("user") != null && vars.get("setUpData").toLowerCase().split(",")*.trim().contains("user") && !["admin", "suzy", "pat", "tiffany"].contains(vars.get("user").trim().toLowerCase())) </pre>
roleAssignments		<pre> \$__groovy(vars.get("role") && vars.get("rolePermission") != null && !vars.get("role").equalsIgnoreCase("Anonymous")) </pre>

setUp

Keyword	Description	Condition
---------	-------------	-----------

Keyword	Description	Condition
setUpProperties	<p>Scenario: Get settings properties to run test suite Given variables rampUpDivisor and delimiterCsv When execute script Then create <properties></p> <pre> properties description countLines rows number at csv file (less the header) rampUp division between nº rows and rampUpDivisor csvFile name of csv file csvColumns list of columns in csv file </pre>	

tearDown		
Keyword	Description	Condition
deleteRootPath	<p>Delete Root Path Given variables csvFile, path, setUpData, delimiterCsv When setUpData contains "path" string Then get list of atomic paths # *1 And Delete each path in List if path id exist</p> <p>*1 atomic paths are: The root file/directory to delete excluding default ones</p>	