Command to execute individual test case in robot frame work:

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

**robot -t testcase1 mytestsuite.robot**

-t testcase1 ---test case to execute

To put the result in specific folder

**pybot -d ../results TestingRestAPI.robot**

-d ../results --- to put in results folder (results log)

../ -- to move to parent dir from current dir

Python handling Json.

def restJSON():

r = requests.get("http://httpbin.org/get")

# print "Code" , r.status\_code

# print "Text " , r.text

return r.json()

We stored the json inside a file output. And we have written robot test case to evaluate json comparison as below

\*\*\* Settings \*\*\*

Library rest2.py

Library OperatingSystem

\*\*\* Test Cases \*\*\*

Example that calls a python keyword

${result}= restJSON

${json}= Get file output

Should be equal ${result} ${json}

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

Example of using only robot for testing REST call

https://stackoverflow.com/questions/36916304/comparing-json-result-from-a-function-in-robot-framework

\*\*\* Settings \*\*\*

Library RequestsLibrary

Library Collections

Library XML use\_lxml=True

Force Tags REST

\*\*\* Variables \*\*\*

${SERVICE\_ROOT} http://ip.jsontest.com/

${SERVICE\_NAME} testing

\*\*\* Keywords \*\*\*

json\_property\_should\_equal

[Arguments] ${json} ${property} ${value\_expected}

${value\_found} = Get From Dictionary ${json} ${property}

${error\_message} = Catenate SEPARATOR= Expected value for property " ${property} " was " ${value\_expected} " but found " ${value\_found} "

Should Be Equal As Strings ${value\_found} ${value\_expected} ${error\_message} values=false

\*\*\* Test Cases \*\*\*

Example REST JSON

Create session ${SERVICE\_NAME} ${SERVICE\_ROOT}

${headers}= Create Dictionary Content-Type=application/json Accept=application/json

${result}= Get Request ${SERVICE\_NAME} ${SERVICE\_ROOT} headers=${headers}

Should Be Equal ${result.status\_code} ${200}

Log ${result.content}

${json}= To Json ${result.content}

${pp}= To Json ${result.content} pretty\_print=True

Log ${pp}

json\_property\_should\_equal ${json} ip [Your\_IP]

Remember to install all needed libraries:

pip install robotframework-requests

pip install requests