pybot 参数详解   
1. -M –metadata name:value 给最顶层测试套件设置metadata (主要体现在测试报告中)   
- a) example: –metadata version:1.2   
2. -G –gettag tag \* 给所有执行的测试 case设置给出的tag    
3. -t –test name \* 根据测试case的名字或全面去选择运行。   
- a) 名字是大小写和空格不敏感的，同时也可以用\* 匹配所有case, ?可以匹配任意字符   
- b) 如果用 \* 和 ？运行， 在控制台是有问题的，请查看 –escape 和 –argumentfile 参数   
4. -s –suite name \* 根据选择的case 名字运行测试套件，只有测试case 在对应的测试套件中并且符合筛选标准的，才能和–test, –include 或 – exclude 同时使用时，name 参数和 –test 的name 参数标准类似。也可以使用”.” 这个符号作为父节点名的分隔符：   
- a) 例如：-s X.Y 选择父节点是 X 的 Y 测试套件。   
5. -i –include tag \* 根据给出的tag 名去选择运行测试案例。 name 参数和–test 的参数类似。 tag 是大小写和空格不敏感的，它可以使用 “\*” 和 “?” 作为通配符。 Tags 和 模式可以用 AND、OR 和 NOT 关键字连接：   
- a) 例如：–include foo –include bar\*   
- b) –include fooANDbar\*   
6. -e –exclude tag \* 不运行带这些给出标签的案例。其余规则和 –include 相同   
7. -R –rerunfailed output 从之前失败的输出文件中选择失败的案例重新执行。   
- a) 选择相同的案例需要用分别单独使用–test 参数   
8. –runfailed output (RF 2.8.4 版本以后已经被–rerunfailed 替代)   
9. -c –critical tag \* 案例给予Tag被认为是重要的。 如果没有设置重要标签，那么所有的案例都被认为是重要的。 Tag 可以赋予一个模式例如 –include   
10. -n –noncritical tag \* 被赋予Tag 的案例，被认为是不重要的，即使他被设置了critical 标签。 Tag 可以用模式来表达例如： –include test\*   
11. -v –variable name:value \* 在测试数据中设置变量。 只支持 scalar 变量，变量名不需要带’′.请参考–escape参数去使用特殊字符和–variablefile参数去设置更多功能的变量，同样可以使用list变量。−a)例如：–variablestr:Hello=>′.请参考–escape参数去使用特殊字符和–variablefile参数去设置更多功能的变量，同样可以使用list变量。−a)例如：–variablestr:Hello=>{str}=’Hello’   
- b) -v str:Hello\_World -E space:\_ =>str=′HelloWorld′−c)−vx:−vy:42=>str=′HelloWorld′−c)−vx:−vy:42=>{x}='',${y}=’42’   
12. -V –variablefile path \* 变量文件的路径：   
- a) 样例文件：   
i.   
- b) 等同于：   
i.   
13. -d –outputdir dir 存放输出文件的路径。默认目录是测试案例执行的目录，给出的目录也是相对于案例执行的目录，除非设置的目录是绝对路径。   
14. -o –output file XML 输出文件，给出路径，和 –log , –report , –xunit , and –debugfile, 相似，基于 –outputdir 的目录，除非给出的是绝对路径。 其他输出文件都是基于案例执行后的xml 输出文件生成的。 XML 输出文件也可以为以后的robot 工具提供数据。可以给出 NONE 字符串禁止这个功能，同时还会禁止 log (日志) 和 report (报告) 功能。   
15. -l –log file HTML 日志文件。 可以给出 NONE 禁止输出，默认输出 log.html   
- a) 例如：–log mylog.html   
- b) -l NONE   
16. -r –report file HTML 报告文件。可以赋予 NONE 禁止输出，默认输出：report.html。 类似于 –log 参数。   
17. -x –xunit file xUnite 兼容性 测试文件。 除非特别声明，否则不会被创建。   
- a) –unitfile file (已经废弃 用 –xunit 代替)   
- b) –xunitskipnoncritical 标记非重要案例在 xUnit 输出结果中跳过。   
18. -b –debugfile file 运行期间的调试文件，除非特别声明，否则不被创建。   
19. -T –timestampoutputs 当使用这个参数时，时间戳格式 ‘YYYYMMDD-hhmmss’ 将被应用于所有输出文件名的基本名和扩展名之间。   
- a) 例如： -T -o ouputxml -r report.html -l none   
- b) 创建文件：ouput-20161023-121011.xml 和 report-20161023-121012.xml   
- c) –splitlog 分割日志文件成小文件，用浏览器打开更清晰   
- d) –logtitle title 生成日志的标题。默认标题是测试套件的名字。 所有测试标题中的下划线都被转换成空格。   
- e) –reportbackgroud colors 测试报告的背景设置。   
- - i. ‘all\_passed:critical\_passed:failed’ 或者 ‘passed:failed’ 颜色和代码都可以。   
- - - 1. 例如：–reportbackgroud green:yellow:red   
- - - 2. –reportbackgroud #00E:#E00   
20. -L –loglevel level 设置日志级别。可用级别：TRACE,DEBUG,INFO (默认级别). WARN,NONE(no loging)。使用语法：LOGLEVEL:DEFAULT   
- a) 例子：–loglevel DEBUG   
- b) –loglevel DEBUG:INFO   
21. –suitestatleve level 在日志和测试报告中显示多少层。 默认显示所有层.   
- a) 例子：–suitestatlevel 3   
22. –tagstatinclude tag \* 只在测试套件统计和测试细节在日志和报告中显示匹配的TAG。Tag默认在所有测试案例中显示。给出的Tag 可以写成一个简单的模式（参考 –test 例子）   
23. –tagstatexclude tag \* 只在 按测试套件统计和测试细节的log 和测试报告中不显示匹配的TAG. 这个选项不能和 –tagstatinclude 一起使用，类似 –exclude 和 –include 同时使用。   
24. –tagstatcombine tags:name \* 根据tag 创建综合统计，这些统计会被添加到按TAG 统和匹配的测试案例细节中。 如果可选项 name 没有给出，综合统计的名字会从指定的tags 里面获取。 Tags 将结合–include 给出的规则一起使用。   
- a) 例子： –tagstatcombine requrement-\*   
- b) —tagstatcombine tag1ANDtag2:My\_name   
25. –tagdoc pattern:doc \* 为符合条件的tag 添加文档说明。文档会显示在测试细节中，同样也会以 工具提示的形式显示在’Statistics by Tag’ 中。 模式可以包含 \* 匹配所有，和’?’ 匹配任意单个字符。 文档可以包含格式，类似于 –doc 选项。   
- a) 例子： –tagdoc mytag:My\_documentation   
- b) –tagdoc regression:\*See\*\_http://info.html   
- c) –tagdoc owner-\*:Original\_author   
26. –tagstatlink pattern:link:title \* 在按tag统计中添加额外的连接地址。pattern 可以包含 \* 匹配所有 和 ? 匹配任意字符。字符串通配符表达式可以用 %N 这种语法应用在连接 和 标题中。N 是从第几位开始匹配。 在标题中下划线会被自动转换成空格。   
- a) 例子：–tagstatlink mytag:[http://my.domain:Link](http://my.domain:Link/)   
- b) –tagstatlink bug-\*:http:/tracker/id=%1:Bug\_Tracker   
27. –removekeywords all|passed|for|wuks|name: \* 从生成的日志文件中移除关键字数据。 关键中如果包含警告则不会被移除，除非选项参数是all .   
- a) all: 从所有关键字中移除数据   
- b) passed: 从成功的测试案例和套件中移除关键字数据。   
- c) for: 从for循环成功的迭代中移除关键字数据。   
- d) wuks: 移除所有处理最后失败的关键字， 包含在’BuiltIn.Wait Until Keyword Succeds’   
- e) name:: 从符合匹配模式的关键字中移除数据。模式会针对关键字全面做匹配。   
- i. 例如：’MyLib.Keyword’,’resource.Secdon Keyword’   
- f) 大小写，空格，下划线不敏感， 可以包含 \* 和 ? 作为通配符。   
- i. 例子：–removekeywords name:Lib.HugeKw   
- ii. –removekeywords name:myresource.\*   
28. –flattenkeywords for|foritem|name: \* Flattens matching keywords in the generated log file. Matching keywords get all log messages from their child keywords and children are discarded otherwise.   
- a) for: flatten for loops fully   
- b) foritem: flatten individual for loop iterations   
- c) name:: flatten matched keywords using same matching rules as with   
- - i. ‘–removekeywords name:’   
29. –listener class \* 一个用于监视测试执行的类。 获取通知的示例当一个测试案例开始和结束时。监听类的参数可以跟在类名后面，用 ‘:’ 作为分隔符。   
- a) 例如：–listener MyListenerClass:arg1:arg2   
30. –warnonskippedfiles 如果使用了该选项，跳过的测试数据文件会在控制台和日志文件中产生一个可见 的警告信息。默认的跳过的文件只会产生一个信息基本的系统日志信息。   
31. –nostatusrc 设置返回码为0 不管测试案例中的失败。错误码正常返回。   
32. –runemptysuite 如果顶层测试套件为空也同样执行。如果和–include/–exclude 一起使用时，如果没有匹配的测试案例，这种场景下很有用。   
33. –dryrun 验证测试数据并且执行测试，以便测试库关键字不执行。   
34. –exitonfailure 任何 critical 基本的案例失败，停止测试。   
35. –exitonerror 当发生任何错误就停止测试   
36. –skipteardownonexit 如果测试执行暂停，就跳过teardown 退出。   
37. –randomize all|suites|tests|none 随机执行。   
- a) all: 随机执行测试套件和测试案例。   
- b) suites: 随机执行测试套件   
- c) tests: 随机执行测试案例。   
- d) none: 不随机执行，默认。   
- e) 使用语法 ‘VALUE:SEED’ 赋予一个用户随机种子，必须是整数。   
- - i. –randomize all   
- - ii. –randomize tesets:1234   
38. –runmode mode \* 2.8版本已经废弃，请分别独立使用：–dryrun, –exitonfailure,–skipteardownonexit, 或者 –randomize 选项代替。   
39. -W –monitorwidth chars 监视器的输出宽度。 默认是 78   
40. -C –monitorcolors auto|on|ansi|off 在控制台使用颜色输出。   
- a) auto: 当输出没有重定向时使用颜色输出。   
- b) on: 总是使用颜色。   
- c) ansi: 想on 参数一样，但是在windows 使用ANSI 颜色。   
- d) off: 禁止颜色   
- e) 注意Jpython 在windows 不显示颜色。   
41. -K –monitormarkers auto|on|off 当顶层关键字在测试案例中结束时，在控制台显示 ‘.’ 成功 或 ‘F’ 失败。 和–monitorcolors 参数的用法一直。   
42. -P –pythonpath path \* 添加额外的路径（directories,ZIPs,JARs）去搜索测试库当他们被导入时。多个路径可以用: 做分隔符。   
- a) 例子：–pythonpath libs/   
- b) –pythonpath /opt/testlibs:mylibs.zip:yourlib   
- c) -E star:STAR -P lib/STAR.jra -P mylib.jar   
43. -E –escape what:with \* 在控制台有问题的转移字符。   
- a) what 代表转移字符的名字，with 是要转移成的字符   
- b) 需要谨慎选择和使用转移字符。