

README BwlBulkRegistration

Martin Westphal, westphal@de.ibm.com

v1.0, 12.4.2017

By using Blueworks Live's REST API this application will register a list of given users for a specific Blueworks Live account.

Content

BwlBulkRegistration.java, BwlBulkRegistration.class

Java source code and compiled version of the program

commons-io-2.4.jar, wink-json4j-1.3.0.jar

required library files, sources:

<https://commons.apache.org/proper/commons-io/>

<http://mvnrepository.com/artifact/org.apache.wink/wink-json4j>

compile.bat, run.bat

Windows batch file to compile and run the program. Should also run as shell commands under Unix/Linux if ';' is replaced by '.'.

Compile

This step is optional. With a Java SDK installed on your machine and the right PATH settings you can simply run either the batch file "compile.bat" or copy the command and run it in a command window:

```
javac -cp .;commons-io-2.4.jar;wink-json4j-1.3.0.jar BwlBulkRegistration.java
```

This will create the compiled version: BwlBulkRegistration.class

Run

With a Java runtime installed on your machine and the right PATH settings you can simply run the batch file "run.bat" or copy the command and run it in a command window. A simple call without any further command line arguments will complain about missing arguments but also gives you all the options, e.g.

```
java -cp .;commons-io-2.4.jar;wink-json4j-1.3.0.jar BwlBulkRegistration
```

```
ERROR: missing command line arguments, 4 arguments required
Usage: BwlBulkRegistration <user> <password> <account> <userListFile> [optional_arguments]
Optional arguments:
  -h          This help message
  -r <role>   The default role for a new user, default=viewer
  -a          Make new users to admin by default
  -c          Check only user list file but do not actually register users
```

So before the batch file "run.bat" can be used, the required arguments such as name, password, Blueworks live account name, and user list file must be adapted by using a text editor of your choice.

```
java -cp .;commons-io-2.4.jar;wink-json4j-1.3.0.jar BwlBulkRegistration
bwladmin@ibm.com XXX IBM70 user.csv
```

Note that the user given as a command argument is only an administrative user that is used to perform the API calls and must be already registered for this account. He needs admin rights to register the additional users given in the user list file.

In order to successfully register new users there must be licenses left and the users must match the user email pattern defined in the account settings. The user list must contain at least one *email* address per line. You can also add more information as comma separated fields. The second field would be the *full name*. You can also add the desired role as a third field which can be one of:

- editor
- contributor
- viewer

You only need to give some matching characters, so an “e” or “edit” would be mapped to “editor”. If this field is not given, the role will be picked according to the default role which is ‘viewer’ if you do not overwrite this setting with flag ‘-r’. Also you can use the flag ‘-a’ if users should have administrative rights (only works for editors and viewers). Here is an example user list:

```
tom@ibm.com
tim@ibm.com,Tim Jones
paul@ibm.com,Paul Miller,e
john@ibm.com,John Smith,editor,true
```

Before you actually run the program to register the users, you should run it with flag ‘-c’ to make sure the entries in your user list file are parsed correctly. If you run it without this flag the program tries to register the users for the BlueworksLive account and reports back the results, e.g.:

```
>REGISTRATION-REQUEST #1 for user bwltest54@ibm.com: {"license":"editor","admin":false,"fullname":"St...
<REGISTRATION-RESULT successfully registered user bwltest54@ibm.com: {"license":"EDITOR","admin":fals...
>REGISTRATION-REQUEST #2 for user bwltest55@ibm.com: {"license":"viewer","admin":false,"fullname":"","...
ERROR: <REGISTRATION-ERROR for user bwltest55@ibm.com (Code=400): Bad Request {"message":"There are n...
>REGISTRATION-REQUEST #3 for user bwltest56@ibm.com: {"license":"contributor","admin":true,"fullname"...
<REGISTRATION-RESULT successfully registered user bwltest56@ibm.com: {"license":"CONTRIBUTOR","admin"...
===== SUMMARY =====
  lines processed : 3
   user entries   : 3
  valid entries   : 3
 registered users : 2
=====
```

A simple approach is to have lists for each role such as editors, contributors and viewers respectively. This could be just an email list or “email, full name”. Then you run each list with the appropriate default role setting controlled by the flag ‘-r’, e.g. ‘-r editor’ for the list with editors.

Finally, here is a short list of messages that you might get back if something goes wrong with the registration for individual user entries:

- "The provided user name is already a member of this account."
- "The provided user name is not valid. Ensure the user name is provided and is a valid email address."
- "There are no remaining licenses for the requested license type."
- "The provided email address belongs to a domain which is blocked by the account."