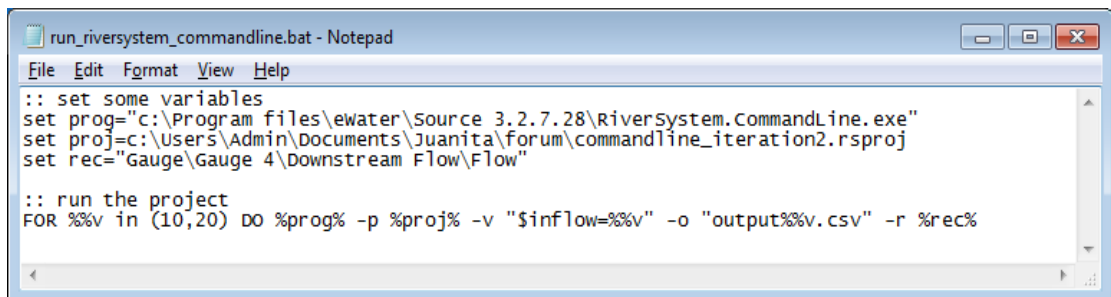


How to run Source from the command line: Hints and examples

Required: *RiverSystem.CommandLine.exe*

Found in Source install folder: *C:\Program Files\ewater\<Source version>*

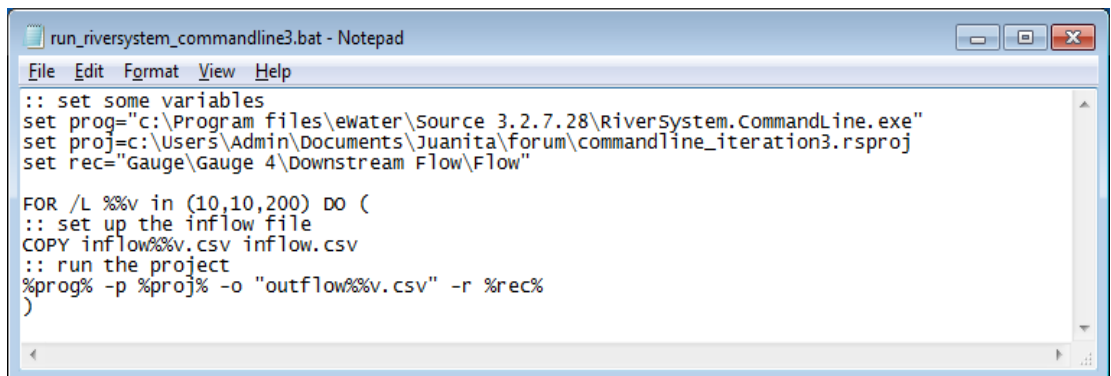
- 1 From a DOS batch file:
 - a) Here is an example of a simple .bat file which iteratively changes the demand on the storage in *commandline_iteration2.rsproj*. The inflow node is set up to expect a variable called *\$inflow*.



```
run_riversystem_commandline.bat - Notepad
File Edit Format View Help
:: set some variables
set prog="c:\Program files\ewater\Source 3.2.7.28\RiverSystem.CommandLine.exe"
set proj=c:\Users\Admin\Documents\Juanita\Forum\commandline_iteration2.rsproj
set rec="Gauge\Gauge 4\Downstream Flow\Flow"

:: run the project
FOR %%v in (10,20) DO %prog% -p %proj% -v "$inflow=%%v" -o "output%%v.csv" -r %rec%
```

- b) And another simple example which iteratively steps through multiple input files, using a new input file for each run of *commandline_iteration3.rsproj*. The project is set up to expect a file called *inflow.csv*, therefore the new input file is copied to a file called *inflow.csv* on each run. An output file is generated for each new input set. This example creates 20 output files numbered 10 to 200 at steps of 10 (e.g. *output10.csv*, *output20.csv*, *output30.csv*, etc). The inflow files have been set up to increase incrementally in the same way.

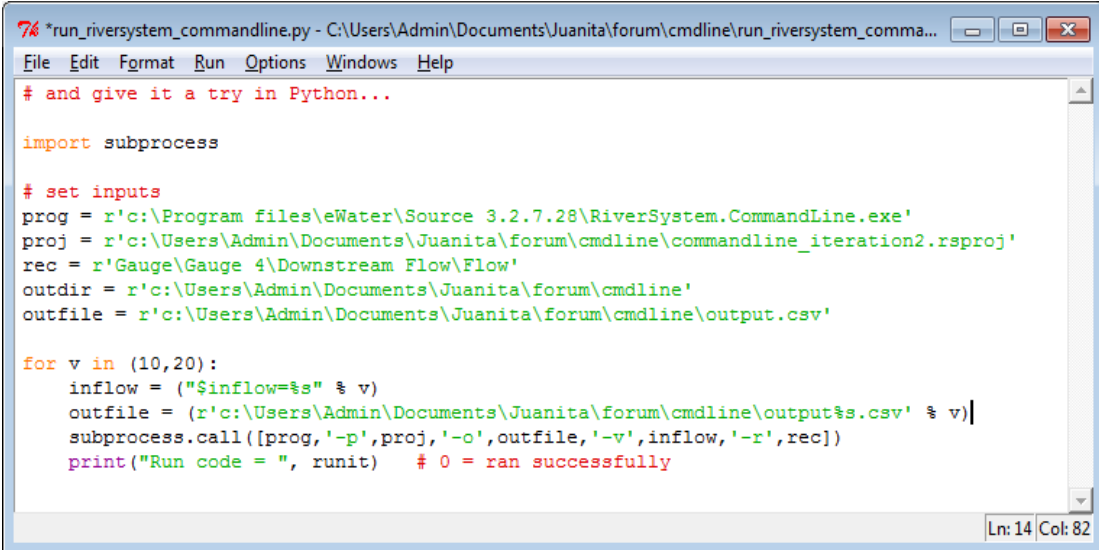


```
run_riversystem_commandline3.bat - Notepad
File Edit Format View Help
:: set some variables
set prog="c:\Program files\ewater\Source 3.2.7.28\RiverSystem.CommandLine.exe"
set proj=c:\Users\Admin\Documents\Juanita\Forum\commandline_iteration3.rsproj
set rec="Gauge\Gauge 4\Downstream Flow\Flow"

FOR /L %%v in (10,10,200) DO (
  :: set up the inflow file
  COPY inflow%%v.csv inflow.csv
  :: run the project
  %prog% -p %proj% -o "outflow%%v.csv" -r %rec%
)
```

2 From a Python script:

a) Here is the same simple example as in 1a) above:



```
*run_riversystem_commandline.py - C:\Users\Admin\Documents\Juanita\forum\cmdline\run_riversystem_comma...
File Edit Format Run Options Windows Help
# and give it a try in Python...

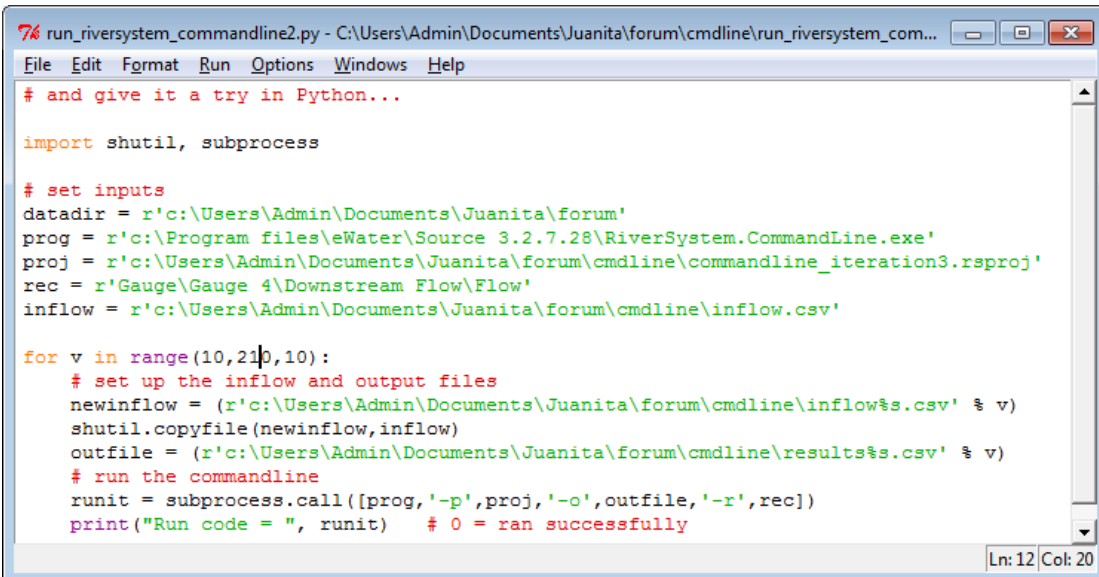
import subprocess

# set inputs
prog = r'c:\Program files\eWater\Source 3.2.7.28\RiverSystem.CommandLine.exe'
proj = r'c:\Users\Admin\Documents\Juanita\forum\cmdline\commandline_iteration2.rsproj'
rec = r'Gauge\Gauge 4\Downstream Flow\Flow'
outdir = r'c:\Users\Admin\Documents\Juanita\forum\cmdline'
outfile = r'c:\Users\Admin\Documents\Juanita\forum\cmdline\output.csv'

for v in (10,20):
    inflow = ("inflow=%s" % v)
    outfile = (r'c:\Users\Admin\Documents\Juanita\forum\cmdline\output%s.csv' % v)|
    subprocess.call([prog, '-p', proj, '-o', outfile, '-v', inflow, '-r', rec])
    print("Run code = ", runit) # 0 = ran successfully

Ln: 14 Col: 82
```

b) And the Python version of 1b) above:



```
*run_riversystem_commandline2.py - C:\Users\Admin\Documents\Juanita\forum\cmdline\run_riversystem_com...
File Edit Format Run Options Windows Help
# and give it a try in Python...

import shutil, subprocess

# set inputs
datadir = r'c:\Users\Admin\Documents\Juanita\forum'
prog = r'c:\Program files\eWater\Source 3.2.7.28\RiverSystem.CommandLine.exe'
proj = r'c:\Users\Admin\Documents\Juanita\forum\cmdline\commandline_iteration3.rsproj'
rec = r'Gauge\Gauge 4\Downstream Flow\Flow'
inflow = r'c:\Users\Admin\Documents\Juanita\forum\cmdline\inflow.csv'

for v in range(10,210,10):
    # set up the inflow and output files
    newinflow = (r'c:\Users\Admin\Documents\Juanita\forum\cmdline\inflow%s.csv' % v)
    shutil.copyfile(newinflow, inflow)
    outfile = (r'c:\Users\Admin\Documents\Juanita\forum\cmdline\results%s.csv' % v)
    # run the commandline
    runit = subprocess.call([prog, '-p', proj, '-o', outfile, '-r', rec])
    print("Run code = ", runit) # 0 = ran successfully

Ln: 12 Col: 20
```

c) Another Python example is described on Confluence, at:

<https://ewater.atlassian.net/wiki/display/SC/Calculating+System+Yield+using+Python+and+the+Source+Command+Line>

3 From an R script:

a) Again, the same simple example as in 1a) above:

```

1 # and now in R
2
3 # use shQuote to deal with the spaces in the path name
4 prog<-shQuote("c:\\Program files\\ewater\\Source 3.2.7.28\\RiverSystem.CommandLine.exe")
5 proj<-"c:\\Users\\Admin\\Documents\\Juanita\\forum\\cmdline\\commandline_iteration2.rsproj"
6 rec<-shQuote("Gauge\\Gauge 4\\Downstream Flow\\Flow")
7
8 for (v in c(10,20))
9 {
10   inflow<-sprintf("$inflow=%s",v)
11   output<-sprintf("c:/Users/Admin/Documents/Juanita/forum/cmdline/outflow_r%s.csv",v)
12   system(paste(prog,'-p',proj,'-o',output,'-v',inflow,'-r',rec),wait=FALSE)
13 }

```

b) And the same example as in 1b) above:

```

1 # and now in R
2
3 # use shQuote to deal with the spaces in the path name
4 prog<-shQuote("c:\\Program files\\ewater\\Source 3.2.7.28\\RiverSystem.CommandLine.exe")
5 proj<-"c:\\Users\\Admin\\Documents\\Juanita\\forum\\cmdline\\commandline_iteration3.rsproj"
6 rec<-shQuote("Gauge\\Gauge 4\\Downstream Flow\\Flow")
7 inflow<-"c:\\Users\\Admin\\Documents\\Juanita\\forum\\cmdline\\inflow.csv"
8
9 for (v in seq(10,200,10))
10 {
11   # inflow file
12   inflow_file<-sprintf("c:/Users/Admin/Documents/Juanita/forum/cmdline/inflow%s.csv",v)
13   file.copy(inflow_file,inflow,overwrite=TRUE)
14   # run the command
15   output<-sprintf("c:/Users/Admin/Documents/Juanita/forum/cmdline/outflow2_r%s.csv",v)
16   system(paste(prog,'-p',proj,'-o',output,'-r',rec),wait=FALSE)
17 }

```

c) More hints for using R with the commandline can be found on Confluence, at: <https://ewater.atlassian.net/wiki/display/R1/Using+RiverSystem.CommandLine+in+R>

4 Compare speed for multiple runs:

Running DOS batchfile for 1b) above, but increase the number of runs to 500.

a) In Standalone mode

Start:	End:	Total time:
15:57:38	16:22:16	~ 25min

a) In Client/Server mode

Start a Server in a DOS command line window e.g.:

"C:\Program Files\Water\<Source version>\RiverSystem.CommandLine.exe" -m Server -p <project>

Open another DOS command line window; run the batch file

Start:	End:	Total time:
15:47:28	15:56:01	~ 9min