



Techila with Python

Contents

- **Short introduction of the main helper functions:**
 - Peach
- **Preparation**
 - Installing required Python packages
- **Creating a test Project**

Main Helper Functions - Peach

- **Simple interface for creating Projects from Python**
 - Included in the Python 'techila' package in the Techila SDK
 - Internal functionality in shared libraries (.dll for Windows, .so for Linux)
- **Peach function syntax typically defines following parameters:**
 - Name of the executable function that will be executed on Workers
 - Python-script containing executable function definitions
 - Input arguments for the executable function
 - List of data files that should be transferred to Workers
 - Number of Jobs in the Project

Main Helper Functions - Peach

- **Example Python peach syntax**
 - Importing the 'techila' package gives access to the Techila helper functions

```
import techila # Import the techila package

result = techila.peach(funcname = 'exec_func',      # Executable function
                       files = ['ex_file.py'],     # Function definitions
                       jobs = 10)                 # Number of Jobs
```

Main Helper Functions - Peach

- Defining input arguments

```
import techila

var = 5

result = techila.peach(funcname = 'exec_func',          # Executable function
                      params = [var, '<vecidx>', '<param>'], # Input arguments
                      files = ['ex_file.py'],         # Function definitions
                      peachvector = range(1, 11))    # Job count & arguments
```

- '<vecidx>' replaced with peachvector index
- '<param>' replaced with peachvector element

Main Helper Functions - Peach

- Transferring additional data files

```
import techila

var = 5

result = techila.peach(funcname = 'exec_func',      # Executable function
                       params = [var, '<vecidx>'], # Function input arguments
                       files = ['ex_file.py'],     # Function definitions
                       datafiles = ['datafile.txt'], # Additional data files
                       peachvector = range(0, 10)) # Job count & arguments
```

Main Helper Functions - Peach

- **Packaging additional Python packages**

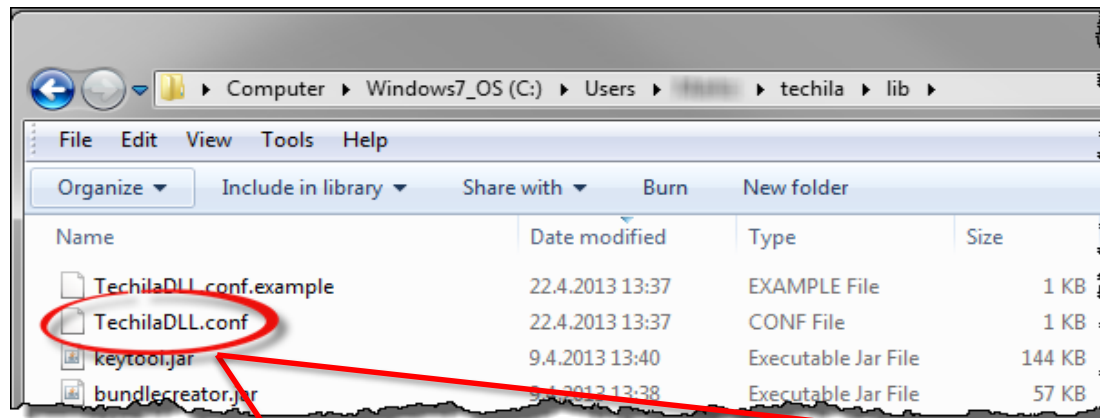
```
import techila

results = techila.peach(funcname = 'ex_func', # Executable function
                        files = ['ex_file.py'], # Function definitions
                        packages = ['numpy', 'bottleneck'], # Package list
                        jobs = 10) # Number of Jobs
```

- **By default, packages can be used by Workers with the same platform as the local computer**
- **Possible to also make the package available to all platforms using the 'bundleit' helper function**
 - **Requires that package only contains pure Python code**
- **Packages containing C extensions require that platforms are same**

Preparation

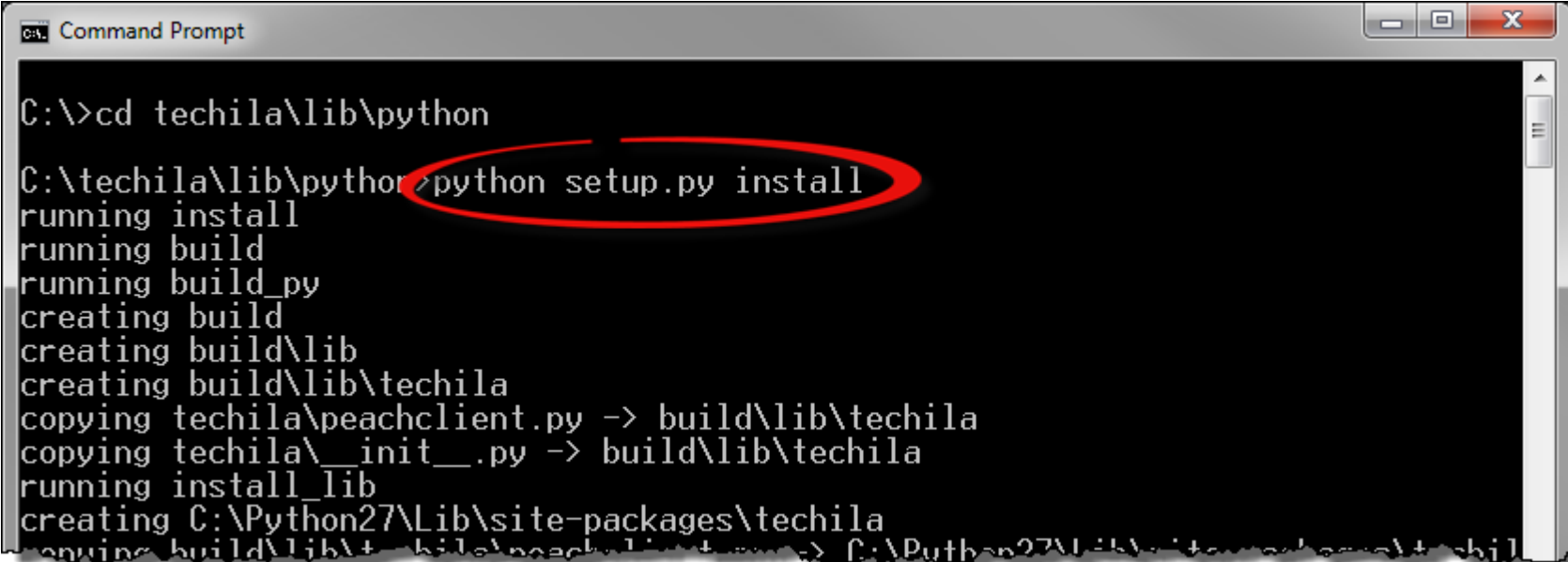
- **Configuring the location of the Java Virtual Machine ('jvm.dll' or 'jvm.so') in the 'TechilaDLL.conf' file.**
 - Navigate to 'techila/lib' in the Techila SDK
 - Remove the '.example' suffix from 'TechilaDLL.conf.example'
 - Edit the file so the JVMPATH parameter points to the 'jvm.dll' or 'jvm.so'



```
1 JVMPATH=C:\Program Files\Java\jdk1.7.0_17\jre\bin\server\jvm.dll
2
```


Preparation

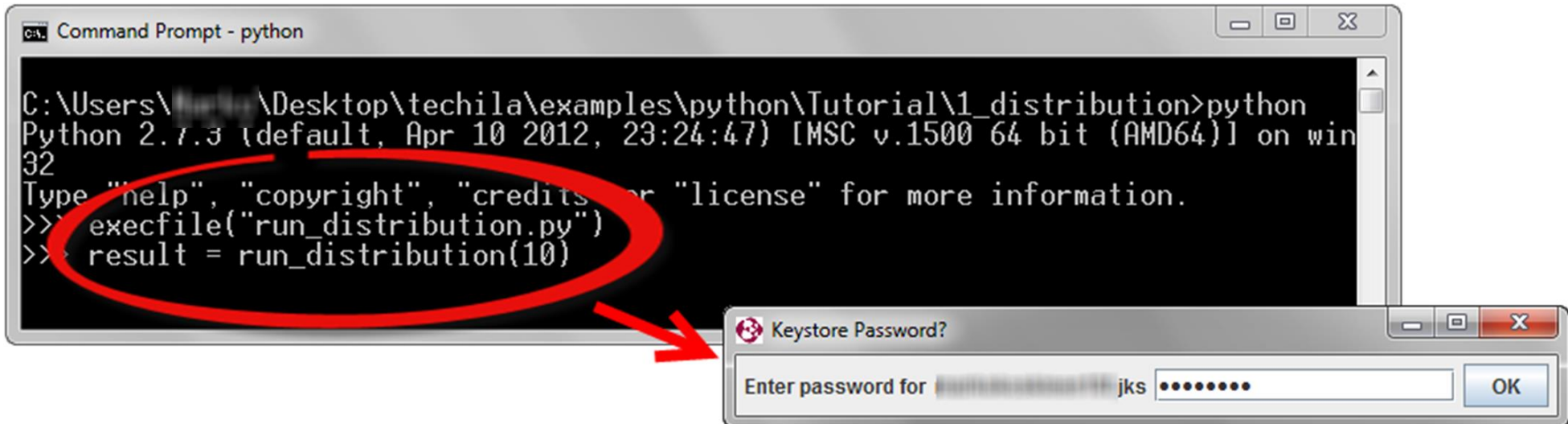
- **Installing the 'techila' package:**
 - Launch a command prompt / terminal and navigate to 'techila\lib\python' in the Techila SDK
 - Install using command:
 - python setup.py install OR
 - python setup.py install --user



```
Command Prompt
C:\>cd techila\lib\python
C:\techila\lib\python>python setup.py install
running install
running build
running build_py
creating build
creating build\lib
creating build\lib\techila
copying techila\peachclient.py -> build\lib\techila
copying techila\__init__.py -> build\lib\techila
running install_lib
creating C:\Python27\Lib\site-packages\techila
copying build\lib\techila\peachclient.py -> C:\Python27\Lib\site-packages\techila
```

Creating a Test Project

- Navigate to '<full path>\techila\examples\python\Tutorial\1_distribution
- In Python, execute:
 - `execfile("run_distribution.py")`
 - `result = run_distribution(10)`



```
Command Prompt - python
C:\Users\...\Desktop\techila\examples\python\Tutorial\1_distribution>python
Python 2.7.3 (default, Apr 10 2012, 23:24:47) [MSC v.1500 64 bit (AMD64)] on win
32
Type "help", "copyright", "credits" or "license" for more information.
>>> execfile("run_distribution.py")
>>> result = run_distribution(10)
```

Keystore Password?

Enter password for jks

OK

Creating a Test Project

- After the Project has been completed, results and statistics will be displayed.

```
Command Prompt - python
Creating Parameter Bundle...
Project ID 5605 created (10 jobs)
Waiting project completion...
Downloading result...
Unzipping...

##### Project Statistics #####
Project ID:          5605
Workers participated: 10
Total CPU time used: 0 d 0 h 0 m 0 s
Wall clock time used: 0 d 0 h 0 m 9 s
Acceleration factor: 0.00x

##### Job Statistics #####
Avg CPU core usage: 75.34% (CPU time / wall clock time)
CPU Time:          0.046 s (min) 0.065 s (avg) 0.077 s (max)
Memory used:       5.757 MB (avg) 5.770 MB (max)
I/O read:          0.395 MB (avg) 0.395 MB (max)
I/O write:         0.000 MB (avg) 0.000 MB (max)
Average total I/O: 0.395 MB (6.111 MB/s)

[2, 2, 2, 2, 2, 2, 2, 2, 2]
>>>
```

WWW.TECHILATECHNOLOGIES.COM