

## Table of Contents

Table of Contents	1
FUTU-PY-CONFFI DEVOPS GUIDE	2
1. INSTALLATION AND DEPLOYMENT	2
1.1. Prerequisites	2
1.1.1. Install python	2
1.1.2. Install pip	2
1.1.3. Install the Google oauth2client	3
1.1.4. Install the PyOpenSSL	3
1.1.5. Install the gspread	3
1.2. Fetch the source	3
1.3. Create you local conf file	4
1.4. run the tests	4
2. CONFIGURATION	4
2.1. oAuthClient configuration	4
3. INTERNALS	4
3.1. Configure logging	4
3.2. Python objects listing	4
3.3. Configure the credentials for the oAuth2Client	4
3.4. sfw/python/futu_py_conffi.py	4
3.5. sfw/python/ctrl/GoogleSheetToYAML.py	5
3.6. sfw/python/utlis/Initializer.py	5
3.7. sfw/python/utlis/Configurator.py	5
4. TESTING	5
4.1. How-to add / remove tests	5
5. PACKAGING AND DEPLOYMENT	5
5.1. to create a zip file for deploy	5
5.2. to create a password protected file for deploy	5

# FUTU-PY-CONFFI DEVOPS GUIDE

## 1. INSTALLATION AND DEPLOYMENT

### 1.1. Prerequisites

The must have binaries are:  
bash, perl, zip, python3

The nice to have are:  
tmux, vim ,ctags

The examples are for Ubuntu - use you OS package manager ...

```
apt-get autoclean
apt-get install --only-upgrade bash
sudo apt-get install -y perl

# optionally
sudo apt-get install -y excuberant-ctags
sudo apt-get install -y 7z

apt-get upgrade
```

#### 1.1.1. Install python

Install the python3 binary as follows:

```
sudo apt-get install python3

# and verify
python3 -V
Python 3.5.2
```

#### 1.1.2. Install pip

Install the python package manager:

The example is for Ubuntu - use the package manager of your OS by choice.

```
sudo apt-get install python3-pip

# and upgrade to the latest version
sudo pip3 install --upgrade pip
```

```
# verify the correct version
pip3 -V
# output:
# pip 9.0.1 from /usr/local/lib/python3.5/dist-packages (python 3.5)
```

### 1.1.3. Install the Google oauth2client

Install the Google oauth2client ( <https://github.com/google/oauth2client> ) as follows:

```
sudo pip3 install --upgrade oauth2client

# and verify ( those are 2 lines )
python3 -c 'import oauth2client
print ( oauth2client.__version__ )'

# output:
# 4.0.0
```

### 1.1.4. Install the PyOpenSSL

Install the Google oauth2client ( <https://github.com/google/oauth2client> ) as follows:

```
sudo pip3 install PyOpenSSL
```

### 1.1.5. Install the gspread

The Google Spreadsheets Python API is utilized and it must be imported via the pip manager.

Use the gspread module

<https://github.com/burnash/gspread>

```
sudo pip3 install gspread

# output:...
Successfully installed gspread-0.6.2
```

## 1.2. Fetch the source

Fetch the source from git hub as follows:

```
# create your product dir:
mkdir -p /opt/futu/futu-pyconfi
cd /opt/futu/futu-pyconfi/
git clone https://github.com/futurice/futu-py-confi

mv -v /opt/futu/futu-py-confi /opt/futu/futu-py-confi.0.1.4.dev.$USER
```

### 1.3. Create you local conf file

The default conf file provides only limited functionality ( this is by design ) , thus copy and configure the configuration file for your host

```
# go to the product version dir
cd /opt/futu/futu-py-confii.0.1.4.dev.$USER

mv -v sfw/bash/futu-pyconfii/futu-pyconfii.set-your-host.conf \ sfw/bash/futu-pyconfii/futu-pyconfii.`hostname -s`.conf
```

### 1.4. run the tests

To run all the tests issue the following call

```
# go to the product version dir
cd /opt/futu/futu-py-confii.0.1.4.dev.$USER

bash sfw/bash/futu-py-confii/test-futu-py-confii.sh
```

## 2. CONFIGURATION

### 2.1. oAuthClient configuration

oAuthClient configuration todo ...

## 3. INTERNALS

### 3.1. Configure logging

To configure the logging edit the logging.conf file by consulting the following documentation:  
<https://docs.python.org/3.5/howto/logging.html#logging-basic-tutorial>

### 3.2. Python objects listing

The following section provides short explanation on the python objects

### 3.3. Configure the credentials for the oAuth2Client

Using the :  
<http://gspread.readthedocs.io/en/latest/oauth2.html>  
<https://github.com/burnash/gspread>

### 3.4. sfw/python/futu\_py\_confii.py

This is the main entry of the python code ...  
Kicks up the configurator and the GoogleSheetToYAML controller obj

### 3.5. [sfw/python/ctrl/GoogleSheetToYAML.py](#)

Implements the control flow for reading the Google SpreadSheet data , data transformations and writing to yaml configuration files

### 3.6. [sfw/python/utils/Initializer.py](#)

A "self-reflecting" module to fetch generically from the project dirs , configuration files etc.

### 3.7. [sfw/python/utils/Configurator.py](#)

Responsible for the gloal configuration of the tool.

## 4. TESTING

### 4.1. [How-to add / remove tests](#)

To add remove tests edit the test run action files

```
# go to the product version dir
cd /opt/futu/futu-py-confii.0.1.4.dev.$USER

sfw/bash/futu-py-confii/tests/run-futu-py-confii-tests.lst
```

## 5. PACKAGING AND DEPLOYMENT

### 5.1. [to create a zip file for deploy](#)

to create a zip file for deploy issue the following call

```
# go to the product version dir
cd /opt/futu/futu-py-confii.0.1.4.dev.$USER

bash sfw/bash/futu-py-confii/futu-py-confii.sh -a create-full-package
```

### 5.2. [to create a password protected file for deploy](#)

to create a password protected file for deploy issue the following call

```
# go to the product version dir
cd /opt/futu/futu-py-confii.0.1.4.dev.$USER
```

```
bash sfw/bash/futu-py-confii/futu-py-confii.sh -a create-full-7z-package
```

You will be prompted to add a secret pass for the package if you have not configured the pw in you shell as folows:

```
export pcking_pw=***
```