7. Entry Orders

This section covers **entry orders**.

In [1]:

```python
import fxcmpy
import pandas as pd
import datetime as dt
con = fxcmpy.fxcmpy(config_file='fxcm.cfg')
```

7.1. Existing Orders

After connecting the FXCM Server, `fxcmpy` collects all existing orders. `list` or `DataFrame` object of the existing orders is returned by the `con.get_orders()` method.

In [2]:

```python
con.get_orders().T  # returns DataFrame by default
```

Out[2]:

	0	1
accountId	2815291	2815291
accountName	02815291	02815291
amountK	300	30
buy	110	0
currency	USD/JPY	EUR/USD
currencyPoint	26.7011	3
expireDate	12012018110000	10112018000000
isBuy	True	False
isELSOrder	False	False
isEntryOrder	True	True
isLimitOrder	True	False
isNetQuantity	False	False
isStopOrder	False	True
limit	112	1.12
limitPegBaseType	-1	-1
limitRate	112	1.12
ocoBulkId	0	423235902
orderId	422719537	423235903
range	0	0
ratePrecision	3	5
sell	0	1.13
status	1	1
stop	0	1.15
stopMove	0	0
stopPegBaseType	-1	-1
stopRate	0	1.15
t	3	3
time	07122018091708	07122018143856
timeInForce	GTD	GTD
tradeId	186558353	186829211
type	LE	SE

In [3]:

```python
con.get_orders(kind='list')  #  returns a list
```

Out[3]:

```python
[{'accountId': '2815291',
  'accountName': '02815291',
  'amountK': 300,
  'buy': 110,
  'currency': 'USD/JPY',
```

```
  'currencyPoint': 26.70156,
  'expireDate': '12012018110000',
  'isBuy': True,
  'isELSOrder': False,
  'isEntryOrder': True,
  'isLimitOrder': True,
  'isNetQuantity': False,
  'isStopOrder': False,
  'limit': 112,
  'limitPegBaseType': -1,
  'limitRate': 112,
  'ocoBulkId': 0,
  'orderId': '422719537',
  'range': 0,
  'ratePrecision': 3,
  'sell': 0,
  'status': 1,
  'stop': 0,
  'stopMove': 0,
  'stopPegBaseType': -1,
  'stopRate': 0,
  't': 3,
  'time': '07122018091708',
  'timeInForce': 'GTD',
  'tradeId': '186558353',
  'type': 'LE'},
 {'accountId': '2815291',
  'accountName': '02815291',
  'amountK': 30,
  'buy': 0,
  'currency': 'EUR/USD',
  'currencyPoint': 3,
  'expireDate': '10112018000000',
  'isBuy': False,
  'isELSOrder': False,
  'isEntryOrder': True,
  'isLimitOrder': False,
  'isNetQuantity': False,
  'isStopOrder': True,
  'limit': 1.12,
  'limitPegBaseType': -1,
  'limitRate': 1.12,
  'ocoBulkId': 423235902,
  'orderId': '423235903',
  'range': 0,
  'ratePrecision': 5,
  'sell': 1.13,
  'status': 1,
  'stop': 1.15,
  'stopMove': 0,
  'stopPegBaseType': -1,
  'stopRate': 1.15,
  't': 3,
  'time': '07122018143856',
  'timeInForce': 'GTD',
  'tradeId': '186829211',
  'type': 'SE'}]
```

Each order has a unique `order_id`, these `ids` are returned by `con.get_order_ids()`.

In [4]:

```
con.get_order_ids()
```

Out[4]:

```
[422719537, 423235903]
```

7.2. Placing Entry Orders

Entry orders are placed via the `con.create_entry_order()` method. For a detailed description of the parameters refer to the FXCM API documentation.

In [5]:

```
order = con.create_entry_order(symbol='USD/JPY', is_buy=True,
                               amount=300, limit=112,
                               is_in_pips = False,
                               time_in_force='GTD', expiration='2108-10-01',
                               rate=110, stop=None, trailing_step=None,
                               trailing_stop_step=None)
```

In [6]:

```
con.get_orders().T
```

Out[6]:

	0	1	2
accountId	2815291	2815291	2815291
accountName	02815291	02815291	02815291
amountK	300	30	300
buy	110	0	110
currency	USD/JPY	EUR/USD	USD/JPY
currencyPoint	26.7016	3	26.7016
expireDate	12012018110000	10112018000000	10012108000000
isBuy	True	False	True
isELSOrder	False	False	False
isEntryOrder	True	True	True
isLimitOrder	True	False	True
isNetQuantity	False	False	False
isStopOrder	False	True	False
limit	112	1.12	112
limitPegBaseType	-1	-1	-1
limitRate	112	1.12	112
ocoBulkId	0	423235902	0
orderId	422719537	423235903	424767349
range	0	0	0
ratePrecision	3	5	3
sell	0	1.13	0
status	1	1	1
stop	0	1.15	0
stopMove	0	0	0
stopPegBaseType	-1	-1	-1
stopRate	0	1.15	0
t	3	3	3
time	07122018091708	07122018143856	07162018145702255
timeInForce	GTD	GTD	GTD
tradeId	186558353	186829211	187584382
type	LE	SE	LE

7.3. Changing Orders

There are two methods to change entry orders:

- `change_order_stop_limit()` to change an order's stop or limit rate.
- `change_order()` to change an order's amount, rate, order range and/or trailing step.

In [7]:

```
order_id = con.get_order_ids()[-1]
```

In [8]:

```
order_id
```

Out[8]:

```
424767349
```

In [9]:

```
con.change_order_stop_limit(order_id=order_id, is_stop_in_pips=True,
                            is_limit_in_pips=True, limit=110, stop=-1)
```

In [10]:

```
con.change_order(order_id=order_id, amount=500, rate=112)
```

The order is changed accordingly.

In [11]:

```
con.get_orders().T
```

Out[11]:

	0	1	2
accountId	2815291	2815291	2815291
accountName	02815291	02815291	02815291
amountK	300	30	500
buy	110	0	112
currency	USD/JPY	EUR/USD	USD/JPY
currencyPoint	26.6999	3	44.4998
expireDate	12012018110000	10112018000000	10012108000000
isBuy	True	False	True
isELSOrder	False	False	False
isEntryOrder	True	True	True
isLimitOrder	True	False	True
isNetQuantity	False	False	False
isStopOrder	False	True	False
limit	112	1.12	110
limitPegBaseType	-1	-1	0
limitRate	112	1.12	113.1
ocoBulkId	0	423235902	0
orderId	422719537	423235903	424767349
range	0	0	0
ratePrecision	3	5	3
sell	0	1.13	0
status	1	1	1
stop	0	1.15	-1
stopMove	0	0	0
stopPegBaseType	-1	-1	1
stopRate	0	1.15	111.968
t	3	3	3
time	07122018091708	07122018143856	07162018145707783
timeInForce	GTD	GTD	GTD
tradeId	186558353	186829211	187584382
type	LE	SE	LE

7.4. Deleting Orders

Orders can be **deleted** through the method `con.delete_orders()`.

In [12]:

```
order_id = con.get_order_ids()[0]
order_id
```

Out[12]:

```
422719537
```

In [13]:

```
con.delete_order(order_id)
```

In [14]:

```
con.get_orders().T
```

Out[14]:

	0	1
accountId	2815291	2815291
accountName	02815291	02815291
amountK	30	500
buy	0	112
currency	EUR/USD	USD/JPY
currencyPoint	3	44.5
expireDate	10112018000000	10012108000000
isBuy	False	True
isELSOrder	False	False
isEntryOrder	True	True
isLimitOrder	False	True
isNetQuantity	False	False
isStopOrder	True	False
limit	1.12	110
limitPegBaseType	-1	0
limitRate	1.12	113.1
ocoBulkId	423235902	0
orderId	423235903	424767349
range	0	0
ratePrecision	5	3
sell	1.13	0
status	1	1
stop	1.15	-1
stopMove	0	0
stopPegBaseType	-1	1
stopRate	1.15	111.969
t	3	3
time	07122018143856	07162018145707783
timeInForce	GTD	GTD
tradeId	186829211	187584382
type	SE	LE

Deleted or executed orders of the current session are stored in the attribute **old_orders** as an `fxcm_order` object.

In [15]:

```
con.old_orders
```

Out[15]:

```
{422719537: <fxcmpy.fxcmpy_order.fxcmpy_order at 0x7eff1a3a1ba8>}
```

In [16]:

```
con.close()
```