

Python 2.7.13 (v2.7.13:a06454b1afa1, Dec 17 2016, 20:53:40) [MSC v.1500 64 bit (AMD64)] on win32

Type "copyright", "credits" or "license()" for more information.

>>>

===== RESTART: C:\Users\GEORGE.BENNETT\Desktop\pyeds2-example1.py =====

C:\Program Files\EDS92\Api\PyEds2.pyd

['__doc__', '__file__', '__name__', '__package__', 'createLive',
'createThermodynamics', 'setupLogger']

createLive_object = PyEds2.createLive()

createLive_object.connect(asClient=0,asSource=1,Opts=dict())

import PyEds2 as eds

eds.createLive().connect(asClient=0,asSource=1,Opts=dict())

['__doc__', '__file__', '__name__', '__package__', 'createLive',
'createThermodynamics', 'setupLogger']

C:\Program Files\EDS92\Api\PyEds2.pyd

createLive_object.connect(0,0,None)

connInfo

(PyEds2_Live<Disconnected:(none):0> , 'connInfo')

waitTillSynced(None) --> Dict

return dictionary containing infos about connection

{'state': 'Disconnected', 'serverTime': None, 'pointCount': 0, 'rport': 0, 'rhost':
'(none)'}

connect

(PyEds2_Live<Disconnected:(none):0> , 'connect')

connect(asClient, asSource, Opts) --> None

Initialize connection to the EDS2 Live server.

asClient(Int) if 1 connect as client

asSource(Int) if 1 connect as source

Opts(Dict) should contain connection parameters.

originate

(PyEds2_Live<Disconnected:(none):0> , 'originate')

originate(Point) --> None

Point(String) is IEES or Point(Int) is LID

Originate point to start sending point dynamic data

pointDynamicInfo

(PyEds2_Live<Disconnected:(none):0> , 'pointDynamicInfo')

pointDynamicInfo(Point) --> Dict

Point(String) is IESS or Point(Int) is LID
return dictionary containing point dynamic infos

pointStaticInfo
(PyEds2_Live<Disconnected:(none):0> , 'pointStaticInfo')
pointStaticInfo(Point) --> Dict
Point if String is IESS or LID if Int
return dictionary containing point static infos

shut
(PyEds2_Live<Disconnected:(none):0> , 'shut')
shut(None) --> None
Close all the connections to EDS2 Live server.
None

subscribe
(PyEds2_Live<Disconnected:(none):0> , 'subscribe')
subscribe(Point) --> None
Point(String) is IESS or Point(Int) is LID
Subscribe point to receive dynamic data

unoriginate
(PyEds2_Live<Disconnected:(none):0> , 'unoriginate')
unoriginate(Point) --> None
Point(String) is IESS or Point(Int) is LID
UnOriginate point to stop sending dynamic data

unsubscribe
(PyEds2_Live<Disconnected:(none):0> , 'unsubscribe')
unsubscribe(Point) --> None
Point(String) is IESS or Point(Int) is LID
UnSubscribe point to stop receiving dynamic data

waitTillSynced
(PyEds2_Live<Disconnected:(none):0> , 'waitTillSynced')
waitTillSynced(None) --> None
Wait till point db gets synchronized

source.pointDynamicInfo(pointname)=
{ 'RT': 'U', 'TS': (0L, 0L), 'ST': 49920, 'QUAL': 'B', 'AT': (0L, 0L), 'TSS': 0L}
source.pointStaticInfo(pointname)=
{ 'LID': -1, 'RT': 'U', 'UN': '(none)', 'DF': 4, 'IESS': '~dummy point~', 'IDCS':
'~dummy point~', 'ZD': '~dummy point~', 'AR': 'N', 'SID': 0, 'DTS': (0L, 0L), 'AUX':

```

'', 'DESC': '~dummy point~'}
source.connInfo()=
{'state': 'Connecting', 'serverTime': (0L, 0L), 'pointCount': 0, 'rport': 0,
'rhost': '127.0.0.1'}
source=
PyEds2_Live<Connecting:127.0.0.1:0>

```

```

both.pointDynamicInfo(pointname)=
{'RT': 'U', 'TS': (0L, 0L), 'ST': 49920, 'QUAL': 'B', 'AT': (0L, 0L), 'TSS': 0L}
both.pointStaticInfo(pointname)=
{'LID': -1, 'RT': 'U', 'UN': '(none)', 'DF': 4, 'IESS': '~dummy point~', 'IDCS':
 '~dummy point~', 'ZD': '~dummy point~', 'AR': 'N', 'SID': 0, 'DTS': (0L, 0L), 'AUX':
'', 'DESC': '~dummy point~'}
both.connInfo()=
{'state': 'Connecting', 'serverTime': (0L, 0L), 'pointCount': 0, 'rport': 0,
'rhost': '127.0.0.1'}
both=
PyEds2_Live<Connecting:127.0.0.1:0>

```

```

subscriber.pointDynamicInfo(pointname)=
{'RT': 'U', 'TS': (0L, 0L), 'ST': 49920, 'QUAL': 'B', 'AT': (0L, 0L), 'TSS': 0L}
subscriber.pointStaticInfo(pointname)=
{'LID': -1, 'RT': 'U', 'UN': '(none)', 'DF': 4, 'IESS': '~dummy point~', 'IDCS':
 '~dummy point~', 'ZD': '~dummy point~', 'AR': 'N', 'SID': 0, 'DTS': (0L, 0L), 'AUX':
'', 'DESC': '~dummy point~'}
subscriber.connInfo()=
{'state': 'Connecting', 'serverTime': (0L, 0L), 'pointCount': 0, 'rport': 0,
'rhost': '127.0.0.1'}
subscriber=
PyEds2_Live<Connecting:127.0.0.1:0>

```

```

>>>
===== RESTART: C:\Users\GEORGE.BENNETT\Desktop\pyeds2-example1.py =====
C:\Program Files\EDS92\Api\PyEds2.pyd
['__doc__', '__file__', '__name__', '__package__', 'createLive',
'createThermodynamics', 'setupLogger']
createLive_object = PyEds2.createLive()
createLive_object.connect(asClient=0,asSource=1,Opts=dict())
import PyEds2 as eds
eds.createLive().connect(asClient=0,asSource=1,Opts=dict())

```

```

['__doc__', '__file__', '__name__', '__package__', 'createLive',
'createThermodynamics', 'setupLogger']
C:\Program Files\EDS92\Api\PyEds2.pyd

```

`createLive_object.connect(0,0,None)`

`connInfo`

`(PyEds2_Live<Disconnected:(none):0> , 'connInfo')`

`waitTillSynced(None) --> Dict`

return dictionary containing infos about connection

`{'state': 'Disconnected', 'serverTime': None, 'pointCount': 0, 'rport': 0, 'rhost': '(none)'}`

`connect`

`(PyEds2_Live<Disconnected:(none):0> , 'connect')`

`connect(asClient, asSource, Opts) --> None`

Initialize connection to the EDS2 Live server.

`asClient(Int)` if 1 connect as client

`asSource(Int)` if 1 connect as source

`Opts(Dict)` should contain connection parameters.

`originate`

`(PyEds2_Live<Disconnected:(none):0> , 'originate')`

`originate(Point) --> None`

`Point(String)` is IEES or `Point(Int)` is LID

Originate point to start sending point dynamic data

`pointDynamicInfo`

`(PyEds2_Live<Disconnected:(none):0> , 'pointDynamicInfo')`

`pointDynamicInfo(Point) --> Dict`

`Point(String)` is IEES or `Point(Int)` is LID

return dictionary containing point dynamic infos

`pointStaticInfo`

`(PyEds2_Live<Disconnected:(none):0> , 'pointStaticInfo')`

`pointStaticInfo(Point) --> Dict`

`Point` if `String` is IEES or `LID` if `Int`

return dictionary containing point static infos

`shut`

`(PyEds2_Live<Disconnected:(none):0> , 'shut')`

`shut(None) --> None`

Close all the connections to EDS2 Live server.

`None`

`subscribe`

```
(PyEds2_Live<Disconnected:(none):0> , 'subscribe')
subscribe(Point) --> None
Point(String) is IESS or Point(Int) is LID
Subscribe point to receive dynamic data
```

```
unoriginate
(PyEds2_Live<Disconnected:(none):0> , 'unoriginate')
unoriginate(Point) --> None
Point(String) is IESS or Point(Int) is LID
UnOriginate point to stop sending dynamic data
```

```
unsubscribe
(PyEds2_Live<Disconnected:(none):0> , 'unsubscribe')
unsubscribe(Point) --> None
Point(String) is IESS or Point(Int) is LID
UnSubscribe point to stop receiving dynamic data
```

```
waitTillSynced
(PyEds2_Live<Disconnected:(none):0> , 'waitTillSynced')
waitTillSynced(None) --> None
Wait till point db gets synchronized
```

```
source = eds.createLive()
source.connect(0,1,dict())
source.originate(pointname)
source.pointDynamicInfo(pointname)=
{'RT': 'U', 'TS': (0L, 0L), 'ST': 49920, 'QUAL': 'B', 'AT': (0L, 0L), 'TSS': 0L}
source.pointStaticInfo(pointname)=
{'LID': -1, 'RT': 'U', 'UN': '(none)', 'DF': 4, 'IESS': '~dummy point~', 'IDCS':
 '~dummy point~', 'ZD': '~dummy point~', 'AR': 'N', 'SID': 0, 'DTS': (0L, 0L), 'AUX':
 '', 'DESC': '~dummy point~'}
source.connInfo()=
{'state': 'Connecting', 'serverTime': (0L, 0L), 'pointCount': 0, 'rport': 0,
 'rhost': '127.0.0.1'}
source=
PyEds2_Live<Connecting:127.0.0.1:0>
```

```
both = eds.createLive()
both.connect(1,1,dict())
both.originate(pointname)
both.subscribe(pointname)
both.pointDynamicInfo(pointname)=
{'RT': 'U', 'TS': (0L, 0L), 'ST': 49920, 'QUAL': 'B', 'AT': (0L, 0L), 'TSS': 0L}
both.pointStaticInfo(pointname)=
{'LID': -1, 'RT': 'U', 'UN': '(none)', 'DF': 4, 'IESS': '~dummy point~', 'IDCS':
```

```
'~dummy point~', 'ZD': '~dummy point~', 'AR': 'N', 'SID': 0, 'DTS': (0L, 0L), 'AUX':
'', 'DESC': '~dummy point~'}
both.connInfo()=
{'state': 'Connecting', 'serverTime': (0L, 0L), 'pointCount': 0, 'rport': 0,
'rhost': '127.0.0.1'}
both=
PyEds2_Live<Connecting:127.0.0.1:0>
```

```
subscriber = eds.createLive()
subscriber.connect(1,0,dict())
both.subscribe(pointname)
subscriber.pointDynamicInfo(pointname)=
{'RT': 'U', 'TS': (0L, 0L), 'ST': 49920, 'QUAL': 'B', 'AT': (0L, 0L), 'TSS': 0L}
subscriber.pointStaticInfo(pointname)=
{'LID': -1, 'RT': 'U', 'UN': '(none)', 'DF': 4, 'IESS': '~dummy point~', 'IDCS':
 '~dummy point~', 'ZD': '~dummy point~', 'AR': 'N', 'SID': 0, 'DTS': (0L, 0L), 'AUX':
'', 'DESC': '~dummy point~'}
subscriber.connInfo()=
{'state': 'Connecting', 'serverTime': (0L, 0L), 'pointCount': 0, 'rport': 0,
'rhost': '127.0.0.1'}
subscriber=
PyEds2_Live<Connecting:127.0.0.1:0>
```

```
>>>
===== RESTART: C:\Users\GEORGE.BENNETT\Desktop\pyeds2-example1.py =====
C:\Program Files\EDS92\Api\PyEds2.pyd
['__doc__', '__file__', '__name__', '__package__', 'createLive',
'createThermodynamics', 'setupLogger']
createLive_object = PyEds2.createLive()
createLive_object.connect(asClient=0,asSource=1,Opts=dict())
import PyEds2 as eds
eds.createLive().connect(asClient=0,asSource=1,Opts=dict())
```

```
['__doc__', '__file__', '__name__', '__package__', 'createLive',
'createThermodynamics', 'setupLogger']
C:\Program Files\EDS92\Api\PyEds2.pyd
```

```
createLive_object.connect(0,0,None)
```

```
connInfo
(PyEds2_Live<Disconnected:(none):0> , 'connInfo')
waitTillSynced(None) --> Dict
return dictionary containing infos about connection
{'state': 'Disconnected', 'serverTime': None, 'pointCount': 0, 'rport': 0, 'rhost':
'(none)'}

```

connect
(PyEds2_Live<Disconnected:(none):0> , 'connect')
connect(asClient, asSource, Opts) --> None
Initialize connection to the EDS2 Live server.
asClient(Int) if 1 connect as client
asSource(Int) if 1 connect as source
Opts(Dict) should contain connection parameters.

originate
(PyEds2_Live<Disconnected:(none):0> , 'originate')
originate(Point) --> None
Point(String) is IESS or Point(Int) is LID
Originate point to start sending point dynamic data

pointDynamicInfo
(PyEds2_Live<Disconnected:(none):0> , 'pointDynamicInfo')
pointDynamicInfo(Point) --> Dict
Point(String) is IESS or Point(Int) is LID
return dictionary containing point dynamic infos

pointStaticInfo
(PyEds2_Live<Disconnected:(none):0> , 'pointStaticInfo')
pointStaticInfo(Point) --> Dict
Point if String is IESS or LID if Int
return dictionary containing point static infos

shut
(PyEds2_Live<Disconnected:(none):0> , 'shut')
shut(None) --> None
Close all the connections to EDS2 Live server.
None

subscribe
(PyEds2_Live<Disconnected:(none):0> , 'subscribe')
subscribe(Point) --> None
Point(String) is IESS or Point(Int) is LID
Subscribe point to receive dynamic data

unoriginate
(PyEds2_Live<Disconnected:(none):0> , 'unoriginate')
unoriginate(Point) --> None

Point(String) is IESS or Point(Int) is LID
UnOriginate point to stop sending dynamic data

unsubscribe
(PyEds2_Live<Disconnected:(none):0> , 'unsubscribe')
unsubscribe(Point) --> None
Point(String) is IESS or Point(Int) is LID
UnSubscribe point to stop receiving dynamic data

waitTillSynced
(PyEds2_Live<Disconnected:(none):0> , 'waitTillSynced')
waitTillSynced(None) --> None
Wait till point db gets synchronized

```
source = eds.createLive()
source.connect(0,1,dict())
source.originate(pointname)
source.pointDynamicInfo(pointname)=
{'RT': 'U', 'TS': (0L, 0L), 'ST': 49920, 'QUAL': 'B', 'AT': (0L, 0L), 'TSS': 0L}
source.pointStaticInfo(pointname)=
{'LID': -1, 'RT': 'U', 'UN': '(none)', 'DF': 4, 'IESS': '~dummy point~', 'IDCS':
 '~dummy point~', 'ZD': '~dummy point~', 'AR': 'N', 'SID': 0, 'DTS': (0L, 0L), 'AUX':
 '', 'DESC': '~dummy point~'}
source.connInfo()=
{'state': 'Connecting', 'serverTime': (0L, 0L), 'pointCount': 0, 'rport': 0,
 'rhost': '127.0.0.1'}
source=
PyEds2_Live<Connecting:127.0.0.1:0>
```

```
both = eds.createLive()
both.connect(1,1,dict())
both.originate(pointname)
both.subscribe(pointname)
both.pointDynamicInfo(pointname)=
{'RT': 'U', 'TS': (0L, 0L), 'ST': 49920, 'QUAL': 'B', 'AT': (0L, 0L), 'TSS': 0L}
both.pointStaticInfo(pointname)=
{'LID': -1, 'RT': 'U', 'UN': '(none)', 'DF': 4, 'IESS': '~dummy point~', 'IDCS':
 '~dummy point~', 'ZD': '~dummy point~', 'AR': 'N', 'SID': 0, 'DTS': (0L, 0L), 'AUX':
 '', 'DESC': '~dummy point~'}
both.connInfo()=
{'state': 'Connecting', 'serverTime': (0L, 0L), 'pointCount': 0, 'rport': 0,
 'rhost': '127.0.0.1'}
both=
PyEds2_Live<Connecting:127.0.0.1:0>
```



```
subscriber = eds.createLive()
subscriber.connect(1,0,dict())
subscriber.subscribe(pointname)
subscriber.pointDynamicInfo(pointname)=
{'RT': 'U', 'TS': (0L, 0L), 'ST': 49920, 'QUAL': 'B', 'AT': (0L, 0L), 'TSS': 0L}
subscriber.pointStaticInfo(pointname)=
{'LID': -1, 'RT': 'U', 'UN': '(none)', 'DF': 4, 'IESS': '~dummy point~', 'IDCS':
 '~dummy point~', 'ZD': '~dummy point~', 'AR': 'N', 'SID': 0, 'DTS': (0L, 0L), 'AUX':
 '', 'DESC': '~dummy point~'}
subscriber.connInfo()=
{'state': 'Connecting', 'serverTime': (0L, 0L), 'pointCount': 0, 'rport': 0,
 'rhost': '127.0.0.1'}
subscriber=
PyEds2_Live<Connecting:127.0.0.1:0>
```

```
>>>
```