



"Python plugin"

1		1
2		1
3	Python plugin	2
4		3
5		3
6	API	4
7		5
8		6
9		8
10		9

1

Python 3. Python plugin Python
API.

Python.

2

Python plugin :

: Windows 2000 SP4 ,

32-x 64-x

5 MB

(), Advanced Serial Data Logger.

Microsoft Vista :

Program Files,

- 1.
- 2.
3. Windows Vista

Google.

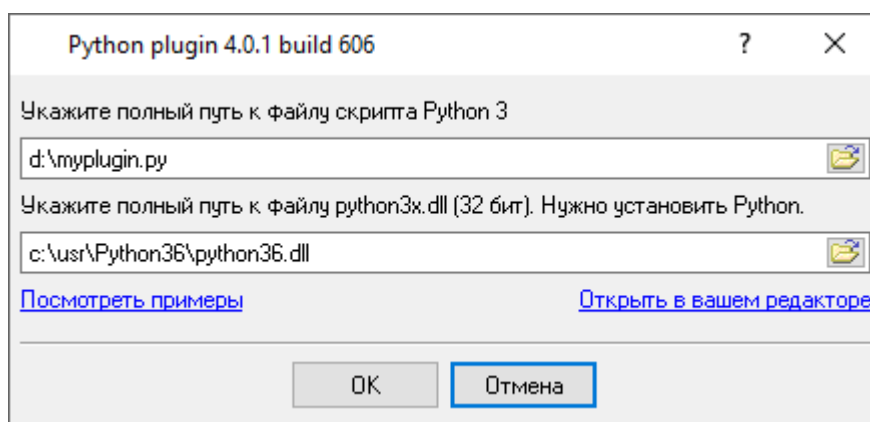
4

Plug-in -

Advanced Serial Data Logger

5

1. [Python 3](#) 32 (x86).
2. ,
3. (. 1).
4. python3x.dll ().



.1.

6 API

```
def FilterData(callparams, datasource, datain, dataout):
```

```
    "Python plugin"
```

```
    callparams - _____
```

```
    datasource - _____ ( _____ None).
```

```
    datain - _____
```

```
    dataout - _____
```

```
        dataout
```

```
        datain
```

```
def ParseData(callparams, datasource, buffer, dataout):
```

```
    "Python plugin"
```

```
( _____ )
```

```
    callparams - _____
```

```
    datasource - _____ ( _____ None).
```

```
    buffer - _____
```

```
            (65 _____)
```

```
    dataout - _____
```

```
FULL_DATA_PACKET - _____
```

```
DATA_PACKET - _____
```

```
( _____
```

```
    FULL_DATA_PACKET).
```

DATA_TIME_STAMP - (datetime.datetime.now()).

```
def ExportData(callparams, datasource, datain):
```

```
    , "Python plugin"
```

callparams - _____

datasource - (None).

datain - _____ **FilterData.**

```
def EventData(event_params, datasource, event):
```

```
    , "Python plugin" - ,
```

event_params - _____

datasource - (None).

event -

```
def main():
```

```
    :
```

```
def main():
```

```
    print('Loaded')
```

```
main()
```

7

Clear() -

Shift(number) - "number" ().

GetBytes() - "bytes".
 , " Shift **GetBytes,** "bytes"

InsertItem(index, name, value) -
index.

name

value

```
>>> item = list.InsertItem(0, 'VALUE', 5)
```

SetItem(name, value[, canadd = False]) -
value.

canadd

True,

name

```
>>> item = list.SetItem('VALUE', 5)
```

AddItem(name, value) -

name

value

```
>>> item = list.AddItem('VALUE', 5)
```

AddItemCopy(item) -

item

```
>>> item_new = list.AddItemCopy(item_old)
```

FindRow(item, end_idx, is_row_end_sign) -

start_idx. True

start_idx). end_idx -

. is_row_end_sign -

True, end_idx

```
from plugin import *
```

```
start_idx = 0
```

```
end_idx = CreateVarParam(-1)
```

```
is_row_end_sign = CreateVarParam(False)
```

```
row_cnt = 0
```

```
while datain.FindRow(start_idx, end_idx, is_row_end_sign):
```

```
    end_idx2 = end_idx.value
```

```
    print('Row: ' + str(row_cnt))
```

```
    print('Start index: ' + str(start_idx))
```

```
    print('End index: ' + str(end_idx2))
```

```
    if is_row_end_sign.value:
```

```
        end_idx2 -= 1
```

```
    row_cnt += 1
```

```
    start_idx = end_idx.value + 1
```

NewRow() -

```
>>> list.NewRow()
```

Count -

```
>>> count = list.Count
```

Items[index] -

```
>>> item = list.Items[0]
```

Name -

```
( )
```

Value -

None.

9

CreateVarParam(value) -

value

```
>>> from plugin import *
>>> obj = CreateVarParam(-1)
```

SendData(datasource, buffer) -

buffer

datasource.

buffer bytes bytearray. datasource

[API.](#)

API.

```
>>> from plugin import *
>>> SendData(datasource, b'ABCD')
```

GetModuleParameter(name) -

name.

```
>>> from plugin import *
>>> value = GetModuleParameter('ModulePath')
```

SetModuleParameter(name, value) -

value

name.

```
>>> from plugin import *
>>> value = GetModuleParameter('ModulePath')
>>> SetModuleParameter('LogTitle', 'MyPlugin')
```

10

API.

"[w]",

ModulePath - (string)

ModuleName - (string)

ApplicationFullName - (string)

ModuleRegistryRoot1 - (dword)
HKEY_LOCAL_MACHINE HKEY_CURRENT_USER

ModuleRegistryRoot2 - (dword)

ModuleRegistryPath - (string)

INIFile - (string) INI

INISectionPrefix - (string)
INI

DisplayFullVersion - (string)

IsTemporaryLoad - (bool)

LogTitle - (string) [w]

Description - (string) [w]

EventsSupported - (string) [w]

LOG-MESSAGE -
NEW-LOG-FILE -
LOG-FILE-DELETE -
NEW-DATA-PACKET -
ERROR-WRITE-FILE -
PORT-OPEN - ()
PORT-CLOSE -
CONFIG-CHANGE -
USER-LOGOFF, USER-LOGON -

STOP-SERVICE - , .