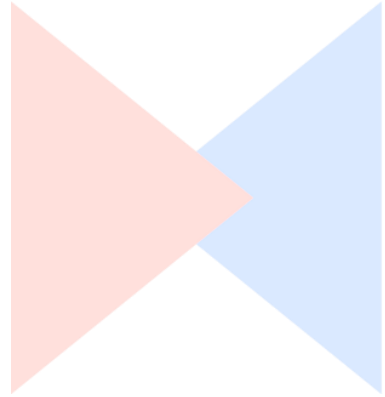


# ***CAF Python***



**CLASSIC AUSIRIS**  
***Plugins***  
INVESTMENT ADVISORY SECURITIES CO.,LTD.



## วัตถุประสงค์

CLASSIC AUSIRIS  
INVESTMENT ADVISORY SECURITIES CO.,LTD.

**การจัดทำ CAF Python Plugin นี้เพื่อให้ลูกค้า CAF ที่ใช้งาน Settrade Streaming สามารถกำหนดให้ Code Python ของลูกค้าส่งคำสั่งซื้อขายโดยการกำหนด สถานะพอร์ต (Target Portfolios) รวมถึงการส่งคำสั่งรูปแบบต่าง ผ่าน CAF Gateway powered by Settrade ได้**

# ฟังก์ชันใน Plugin และการใช้งาน

**Function :** CAF\_Settrade\_SetPortPosition

**Syntax :** CAF\_Settrade\_SetPortPosition(brokerId, accountNo, pin, symbol, targetPortfolios, apiPort)

**Return :** Integer

**Parameter :**

**brokerId** = รหัส Broker (CAF คือ "063")

**accountNo** = เลขที่บัญชี (เช่น "9900123")

**pin** = pin ของบัญชีดังกล่าว (เช่น "123456")

**symbol** = ชื่อสินค้าที่ต้องการเทรด

**targetPortfolios** = จำนวน Position ที่ต้องการให้ถือในพอร์ต โดย

*n* หมายถึง ถือ Long จำนวน *n* Positions

0 หมายถึง ไม่มีสถานะ

*-n* หมายถึง ถือ Short จำนวน *n* Positions

**apiPort** = เลข Port ในไฟล์ application.properties (ค่ามาตรฐาน 8090)

# ฟังก์ชันใน Plugin และการใช้งาน

**Description :** เป็นฟังก์ชันที่ใช้เชื่อมต่อกับ CAF Gateway เพื่อส่งคำสั่งไปยัง Settrade

โดย User จะต้องกรอกข้อมูล brokerId, accountNo, pin, symbol, targetPortfolios, apiPort โดยฟังก์ชันนี้จะ

ทำงานโดยการเช็คพอร์ตตาม symbol ที่กำหนด จากนั้นจะทำการปรับปรุง Port ให้มีสถานะพอร์ตตรงกับ

targetPortfolios ที่เรากำหนด ด้วยการเปิด/ปิด Order ให้เหลือสถานะคงค้างน้อยที่สุด และใช้ราคาตลาด (MP)

## ตัวอย่างการใช้งาน

**CAF\_Settrade\_SetPortPosition("063", "9900227", "111111", "S50M19",20,8090)**

เมื่อเราเรียกใช้งานคำสั่งดังกล่าว Function ใน Library ของ CAF VBA Plugins จะทำการเช็คพอร์ต ที่มี Symbol =

S50U18 จากนั้นจะทำการส่งคำสั่งเพื่อปรับให้พอร์ต ถือสถานะ Long S50M19 จำนวน 20 สัญญา

# ฟังก์ชันใน Plugin และการใช้งาน

**Function :** CAF\_Settrade\_GetAccountEquity

**Syntax :** CAF\_Settrade\_GetAccountEquity(brokerId, accountNo, apiPort)

**Return :** Double

**Parameter :**

|                  |  |
|------------------|--|
| <b>brokerId</b>  | = รหัส Broker (CAF คือ “063”)                              |
| <b>accountNo</b> | = เลขที่บัญชี (เช่น “9900227”)                             |
| <b>apiPort</b>   | = เลข Port ในไฟล์ application.properties (ค่ามาตรฐาน 8090) |

**Description :** เป็นฟังก์ชันที่ใช้เชื่อมต่อกับ CAF Gateway เพื่อตรวจสอบ Equity ในพอร์ต

โดย User จะต้องกรอกข้อมูล brokerId, accountNo, apiPort โดยฟังก์ชันนี้จะทำงานโดยการตรวจสอบ Equity ในพอร์ตของลูกค้าและ Return ค่าดังกล่าวออกมา

## ตัวอย่างการใช้งาน

CAF\_Settrade\_GetAccountEquity(“063”, “9900227”, 8090)

# ฟังก์ชันใน Plugin และการใช้งาน

**Function :** CAF\_Settrade\_GetAccountInfo

**Syntax :** CAF\_Settrade\_GetAccountInfo(brokerId, accountNo, apiPort)

**Return :** String

**Parameter :**

|                  |  |
|------------------|--|
| <b>brokerId</b>  | = รหัส Broker (CAF คือ "063")                              |
| <b>accountNo</b> | = เลขที่บัญชี (เช่น "9900123")                             |
| <b>apiPort</b>   | = เลข Port ในไฟล์ application.properties (ค่ามาตรฐาน 8090) |

**Description :** เป็นฟังก์ชันที่ใช้เชื่อมต่อกับ CAF Gateway เพื่อตรวจสอบข้อมูลของบัญชี

โดย User จะต้องกรอกข้อมูล brokerId, accountNo, apiPort โดยฟังก์ชันนี้จะทำงานโดยสรุปข้อมูลของบัญชีของ

ลูกค้าและ Return ค่าดังกล่าวออกมาเป็นข้อความในรูปแบบของ JSON ข้อมูลได้แก่ creditLine, excessEquity,

cashBalance, equity, totalIMR, totalIMM, callForceFlag และ callForceMargin

## ตัวอย่างการใช้งาน

```
a = caf.CAF_Settrade_GetAccountInfo("063","9900123",8090)
```

# ฟังก์ชันใน Plugin และการใช้งาน

**Function :** CAF\_Settrade\_GetPortfolios

**Syntax :** CAF\_Settrade\_GetPortfolios(brokerId, accountNo, apiPort)

**Return :** String

**Parameter :**

|                  |  |
|------------------|--|
| <b>brokerId</b>  | = รหัส Broker (CAF คือ "063")                              |
| <b>accountNo</b> | = เลขที่บัญชี (เช่น "9900123")                             |
| <b>apiPort</b>   | = เลข Port ในไฟล์ application.properties (ค่ามาตรฐาน 8090) |

**Description :** เป็นฟังก์ชันที่ใช้เชื่อมต่อกับ CAF Gateway เพื่อตรวจสอบข้อมูลของพอร์ต

โดย User จะต้องกรอกข้อมูล brokerId, accountNo, apiPort โดยฟังก์ชันนี้จะทำงานโดยสรุปข้อมูลของพอร์ตของ

ลูกค้าและ Return ค่าดังกล่าวออกมาเป็นข้อความในรูปแบบของ JSON โดยแยกตามสถานะที่ถือครองอยู่ในขณะนั้น

## ตัวอย่างการใช้งาน

`p = caf.CAF_Settrade_GetPortfolios("063","9900123",8090)`

# ฟังก์ชันใน Plugin และการใช้งาน

**Function :** CAF\_Settrade\_GetAllOrderList

**Syntax :** CAF\_Settrade\_GetAllOrderList(brokerId, accountNo, apiPort)

**Return :** String

**Parameter :**

|                  |  |
|------------------|--|
| <b>brokerId</b>  | = รหัส Broker (CAF คือ "063")                              |
| <b>accountNo</b> | = เลขที่บัญชี (เช่น "9900123")                             |
| <b>apiPort</b>   | = เลข Port ในไฟล์ application.properties (ค่ามาตรฐาน 8090) |

**Description :** เป็นฟังก์ชันที่ใช้เชื่อมต่อกับ CAF Gateway เพื่อตรวจสอบคำสั่งซื้อขายทั้งหมด

โดย User จะต้องกรอกข้อมูล brokerId, accountNo, apiPort โดยฟังก์ชันนี้จะทำงานโดยสรุปข้อมูลคำสั่งซื้อขายทั้งหมด

ของลูกค้าและ Return ค่าดังกล่าวออกมาเป็นข้อความในรูปแบบของ JSON โดยแยกตามคำสั่งที่ส่งไปแต่ละคำสั่ง

## ตัวอย่างการใช้งาน

`o = caf.CAF_Settrade_GetAllOrderList("063","9900123",8090)`



# ฟังก์ชันใน Plugin และการใช้งาน

**Function :** CAF\_Settrade\_GetAllTradeList

**Syntax :** CAF\_Settrade\_GetAllTradeList(brokerId, accountNo, apiPort)

**Return :** String

**Parameter :**

|                  |  |
|------------------|--|
| <b>brokerId</b>  | = รหัส Broker (CAF คือ "063")                              |
| <b>accountNo</b> | = เลขที่บัญชี (เช่น "9900123")                             |
| <b>apiPort</b>   | = เลข Port ในไฟล์ application.properties (ค่ามาตรฐาน 8090) |

**Description :** เป็นฟังก์ชันที่ใช้เชื่อมต่อกับ CAF Gateway เพื่อตรวจสอบ Trade List ทั้งหมด

โดย User จะต้องกรอกข้อมูล brokerId, accountNo, apiPort โดยฟังก์ชันนี้จะทำงานโดยสรุปข้อมูล Trade List ทั้งหมด

ของลูกค้าและ Return ค่าดังกล่าวออกมาเป็นข้อความในรูปแบบของ JSON

## ตัวอย่างการใช้งาน

o = caf.CAF\_Settrade\_GetAllTradeList("063","9900123",8090)

# ฟังก์ชันใน Plugin และการใช้งาน

**Function :** *CAF\_Settrade\_CancelOrder*

**Syntax :** *CAF\_Settrade\_CancelOrder(brokerId, accountNo, pin, symbol, orderNo, apiPort)*

**Return :** *String*

**Parameter :**

|                  |  |
|------------------|--|
| <i>brokerId</i>  | = รหัส Broker (CAF คือ "063")                              |
| <i>accountNo</i> | = เลขที่บัญชี (เช่น "9900123")                             |
| <i>pin</i>       | = pin ของบัญชีดังกล่าว (เช่น "123456")                     |
| <i>symbol</i>    | = ชื่อสินค้าที่ต้องการเทรด                                 |
| <i>orderNo</i>   | = เลขคำสั่งซื้อขายที่ต้องการยกเลิก                         |
| <i>apiPort</i>   | = เลข Port ในไฟล์ application.properties (ค่ามาตรฐาน 8090) |

# ฟังก์ชันใน Plugin และการใช้งาน

**Description :** เป็นฟังก์ชันที่ใช้เชื่อมต่อกับ CAF Gateway เพื่อยกเลิกคำสั่งซื้อขาย

โดย User จะต้องกรอกข้อมูล *brokerId, accountNo, pin, symbol, orderNo, apiPort* โดยฟังก์ชันนี้จะทำงานโดยส่งยกเลิกคำสั่งซื้อขายผ่านทาง CAF Gateway และ Return ผลลัพธ์ออกมาเป็นข้อความในรูปแบบของ JSON

## ตัวอย่างการใช้งาน

```
c = caf.CAF_Settrade_CancelOrder("063","9900123","111111","S50M20","1001",8090)
```

# ฟังก์ชันใน Plugin และการใช้งาน

**Function :** CAF\_Settrade\_CancelAllOrder

**Syntax :** CAF\_Settrade\_CancelAllOrder(brokerId, accountNo, pin, symbol, apiPort)

**Return :** String

|                    |                  |  |
|--------------------|------------------|--|
| <b>Parameter :</b> | <b>brokerId</b>  | = รหัส Broker (CAF คือ "063")                              |
|                    | <b>accountNo</b> | = เลขที่บัญชี (เช่น "9900123")                             |
|                    | <b>pin</b>       | = pin ของบัญชีดังกล่าว (เช่น "123456")                     |
|                    | <b>symbol</b>    | = ชื่อสินค้าที่ต้องการเทรด                                 |
|                    | <b>apiPort</b>   | = เลข Port ในไฟล์ application.properties (ค่ามาตรฐาน 8090) |

**Description :** เป็นฟังก์ชันที่ใช้เชื่อมต่อกับ CAF Gateway เพื่อยกเลิกคำสั่งซื้อขายทั้งหมดที่มีสถานะ: Queuing

โดย User จะต้องกรอกข้อมูล brokerId, accountNo, pin, symbol, apiPort โดยฟังก์ชันนี้จะทำงานโดยส่งยกเลิกคำสั่งซื้อขายผ่านทาง CAF Gateway และ Return ผลลัพธ์ออกมาเป็นข้อความแสดง เลขที่คำสั่งทั้งหมดที่ยกเลิกไปคันด้วยเครื่องหมาย “,”

## ตัวอย่างการใช้งาน

```
c = caf.CAF_Settrade_CancelAllOrder("063", "9900123", "111111", "S50M20", 8090)
```

# ฟังก์ชันใน Plugin และการใช้งาน

**Function :** CAF\_Settrade\_SendOrder

**Syntax :** CAF\_Settrade\_SendOrder(brokerId, accountNo, pin, symbol, side, priceType, validity, price, vol, apiPort)

**Return :** String

**Parameter :**

**brokerId** = รหัส Broker (CAF คือ "063")

**accountNo** = เลขที่บัญชี (เช่น "9900123")

**pin** = pin ของบัญชีดังกล่าว (เช่น "123456")

**symbol** = ชื่อสินค้าที่ต้องการเทรด

**side** = "LONG" หรือ "SHORT"

**priceType** = "LIMIT", "ATO", "SPECIAL\_MARKET", "MP", "MP-MTL" หรือ "MP-MKT"

**validity** = "GOOD\_TILL\_DAY", "GOOD\_TILL\_NEXT\_AUTOMATCH", "IOC" หรือ "FOK"

**price** = ราคาที่ต้องการ (มีผลเฉพาะ priceType แบบ LIMIT เท่านั้น)

**vol** = จำนวนสัญญา

**apiPort** = เลข Port ในไฟล์ application.properties (ค่ามาตรฐาน 8090)

# ฟังก์ชันใน Plugin และการใช้งาน

**Description :** เป็นฟังก์ชันที่ใช้เชื่อมต่อกับ CAF Gateway เพื่อส่งคำสั่งซื้อขาย

โดย User จะต้องกรอกข้อมูล *brokerId, accountNo, pin, symbol, side, priceType, validity, price, vol, apiPort*

โดยฟังก์ชันนี้จะทำงานโดยส่งคำสั่งซื้อขายผ่านทาง CAF Gateway และ Return ผลลัพธ์ออกมาเป็นข้อความในรูปแบบของ JSON

## ตัวอย่างการใช้งาน

```
s = caf.CAF_Settrade_SendOrder("063","9900123","111111","S50M20","LONG","LIMIT",  
"GOOD_TILL_DAY",720.2,2,8090)
```

คำสั่งดังกล่าวจะเป็นการส่งคำสั่ง Long S50M20 แบบ Limit Order ที่ราคา 720.2 จำนวน 2 สัญญา

# ฟังก์ชันใน Plugin และการใช้งาน

**Function :** CAF\_Settrade\_SendStopOrder

**Syntax :** CAF\_Settrade\_SendStopOrder(brokerId, accountNo, pin, symbol, priceType, validity, stopSymbol, stopCond, price, vol, stopPrice, apiPort)

**Return :** String

**Parameter :**

|                   |  |
|-------------------|--|
| <b>brokerId</b>   | = รหัส Broker (CAF คือ "063")  |
| <b>accountNo</b>  | = เลขที่บัญชี (เช่น "9900123")   |
| <b>pin</b>        | = pin ของบัญชีดังกล่าว (เช่น "123456")                                 |
| <b>symbol</b>     | = ชื่อสินค้าที่ต้องการเทรด   |
| <b>side</b>       | = "LONG" หรือ "SHORT"  |
| <b>priceType</b>  | = "LIMIT", "ATO", "SPECIAL_MARKET", "MP", "MP-MTL" หรือ "MP-MKT"       |
| <b>validity</b>   | = "GOOD_TILL_DAY", "GOOD_TILL_NEXT_AUTOMATCH", "IOC" หรือ "FOK"        |
| <b>stopSymbol</b> | = ชื่อสินค้าที่ต้องการใช้เป็นเงื่อนไข                                  |
| <b>stopCond</b>   | = "ASK_OR_HIGHER", "ASK_OR_LOWER", "BID_OR_HIGHER" หรือ "BID_OR_LOWER" |

# ฟังก์ชันใน Plugin และการใช้งาน

|                  |   |
|------------------|---|
| <i>price</i>     | = ราคาที่ต้องการ (มีผลเฉพาะ <i>priceType</i> แบบ LIMIT เท่านั้น)        |
| <i>vol</i>       | = จำนวนสัญญา  |
| <i>stopPrice</i> | = ราคาที่ต้องการกำหนดในเงื่อนไข   |
| <i>apiPort</i>   | = <i>apiPort</i> ในไฟล์ <i>application.properties</i> (ค่ามาตรฐาน 8090) |

**Description :** เป็นฟังก์ชันที่ใช้เชื่อมต่อกับ CAF Gateway เพื่อส่งคำสั่งซื้อขายแบบ Stop Order

โดย User จะต้องกรอกข้อมูล *brokerId*, *accountNo*, *pin*, *symbol*, *priceType*, *stopSymbol*, *stopCond*, *price*, *vol*, *stopPrice*, *apiPort* โดยฟังก์ชันนี้จะทำงานโดยส่งคำสั่งซื้อขายผ่านทาง CAF Gateway และ Return ผลลัพธ์ออกมาเป็นข้อความในรูปแบบของ JSON

## ตัวอย่างการใช้งาน

```
s = caf.CAF_Settrade_SendStopOrder("063", "9900123", "111111", "S50M20", "LONG", "LIMIT",  
"GOOD_TILL_DAY", "S50U20", "BID_OR_HIGHER", 720.1, 2, 790, 8090)
```

คำสั่งดังกล่าวจะเป็นการตั้ง Stop Order เงื่อนไขคือ เมื่อ S50U20 มี Bid  $\geq$  790 จะส่งคำสั่ง Long S50M20 แบบ Limit Order ที่ราคา 720.1 จำนวน 2 สัญญา