redis\_mdld.py

# 初始化

class Config

def \_\_init\_\_

config\_flist: dict

config\_optlist:dict

config\_slist: list

def init\_flist: # 期货列表

遍历115\_6380:qdb:securityex:derivatives:\*:CODE #> qdb:securityex:derivatives:IC01:CODE

return { "IH01": "IH1908", ...} 共计12个合约

def init\_optlist: # 期权列表

遍历170:OPLST:01\*

取InstrumentCode[7:]建字典 # 510050P1909M03000

return {"1909M03000":("10001746", "10001750"), ...}

def init\_slist: # 现货列表

从redis\_mdld.yaml读入

return ["S510050", "I000001"]

# run(key)

9:30-11:30 13:00-15:00 每隔一秒调用一次

ts = key # 3\*3600 + h\*3600+m\*60+s #type: int 头天21:00:00以来的秒数,

cur\_ts为当前key

rd.set("MDLD:cur\_ts", key)

# MDLD:key共计4\*3600+2个键值,循环写入,ts>cur\_ts则为昨天的数据

遍历flist:

取期货行情, KZ:F%s:LATEST, BP1,SP1 #> IH1908 mget

写入MDLD:ts:F:F%s #> IH1908 d={"BP1":

遍历slist.values():

取现货行情, KZ:S%s:LATEST, BP1, SP1

写入MDLD:ts:S:S510050

取净值, KZ:JZ0000KZE%s:NEW, B1, S1

写入MDLD:ts:JZ:S510050

if 510050:

pe = latest

pe\_510050\_SP1

pe\_510050\_BP1

JZ\_510050\_SP1,

JZ\_510050\_BP1

遍历optlist.values(): {"1909M03000": [10001677, 10001678], ...} .items()

取CP两个合约行情170:MD:01+ InstrumentID, 写入MDLD:ts:OP:10001750

MDLD:ts:OP: C1909M03000 CALL

170:MD:0110001677

MDLD:ts:OP: P1909M03000 PUT

170:MD:0110001678

根据pe计算A5和OP价

MDLD:ts:A5:1909P2950M

MDLD:ts:PO:1909P2950M

根据pe计算A5和OP价

PO = Px + Pc - Pp

A5 = PO – Pe

A5S = Px + PcBP1 - PpSP1 – Pe

A5B = Px + PcSP1 - PpBP1 - Pe

MDLD:ts:A5:1909M02950 ｛ “LATEST”:5.4 , "BP1", "SP1", “Pe”= ｝

MDLD:ts:PO:1909M02950

根据pe计算A13

MDLD:ts:A13:IH01 { "B" : IH\_BP1 – pe\_510050\_SP1 ,

"S" : IH\_SP1 – pe\_510050\_BP1,

"L" : IH\_LATEST – pe\_510050\_LATEST,

"C" : IH\_BP1 – JZ\_510050\_SP1,

"R" : IH\_SP1 – JZ\_510050\_BP1 }

…IH02/03/04 / IF01/02/03/04 / IC01/02/03/04

波动率

MDLD:ts:V:C1909MV0000 2922

MDLD:ts:V:C1909MVN050 2872

MDLD:ts:V:C1909MV0050 2972

RT:C/ F？

均线数据: [LIST] （OP,FUTURE,A5,V,JZ）

5s/10s/15s/30s/1m/3m/5m/10m/15m/30m/1h/2h/4h/1d/

......

# 输入

## Config的输入：

redis\_mdld.yaml：存储现货的编号（S510050|S510300|S510500）

## NewThread的输入：

Work\_period:[“09:30:00”,”11:30:00”],[“13:00:00”,”15:00:00”]]

|  |  |  |  |
| --- | --- | --- | --- |
|  | 数值 | string | 单位 |
| KZ:F[IC1908-IH2003] | 43300000 | LATEST、BP1、SP1 | 毫 |
| KZ:S510050 | 125000 | LATEST BP1 SP1 | 角 |
| KZ:JZ0000KZE[510050|510300|510500] | 2.84204 | NEW S1 B1 | 元 |
| MD:01[10001845-10001846] | 6178 6310 6200 | Latest BP1 SP1 | 毫 |

# 输出

## Config的输出：

|  |
| --- |
| config\_flist: {'IC01': 'IC1908', 'IC02': 'IC1909', 'IC03': 'IC1912', 'IC04': 'IC2003', 'IF01': 'IF1908', 'IF02': 'IF1909', 'IF03': 'IF1912', 'IF04': 'IF2003', 'IH01': 'IH1908', 'IH02': 'IH1909', 'IH03': 'IH1912', 'IH04': 'IH2003'} |
| config\_oplist: {'1912M02700': ['10001845', '10001846'], ……} |
| config\_slist: ['S510050', 'S510300', 'S510500'] |

## newThread的输出：

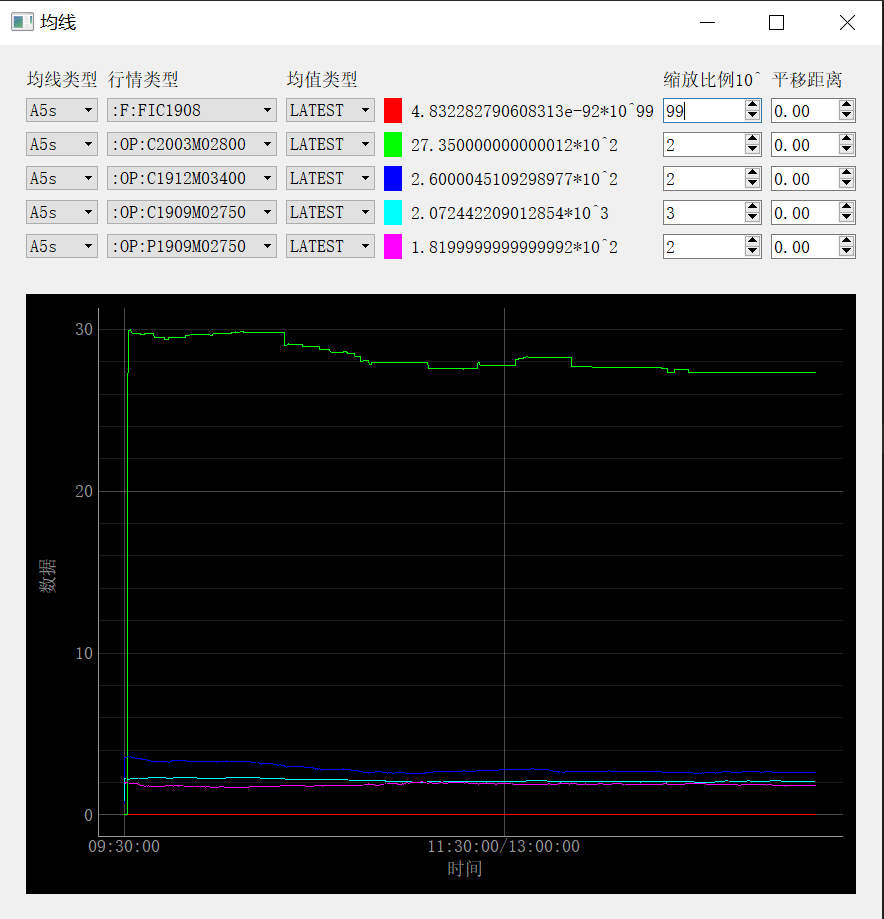
|  |  |  |  |
| --- | --- | --- | --- |
| 键值 | 数值 | Hash键 | 单位 |
| MDLD:cur\_ts | 37800-57600 | 无 | 秒S |
| MDLD:cur\_ts:A13:[IC01-IH04] | -3840、-3838、-3850、46244.9、46246.9 | B、S、L、C、R | 十\*点 |
| MDLD:cur\_ts:A5:[1908M02600-2003M03400] | 18、34、17、28130 | LATEST、BP1、SP1、Pe | 十\*点 |
| MDLD:cur\_ts:F:[FIC1908-FIH2003] | 46250000、46250000、46250000 | LATEST、BP1、SP1 | 毫 |
| MDLD:cur\_ts:JZ:[S510050-S510500] | 2.8130 2.8130 2.8140 | LATEST、BP1、SP1 | 元 |
| MDLD:cur\_ts:OP:[C1908M02600-P2003M03400] | 0、5780、6000 | LATEST、BP1、SP1 (当LATEST未0时，LATEST取PreSettle) | 毫 |
| MDLD:cur\_ts:PO:[1908M02600-2003M03400] | 28287 | 无 | 毫 |
| MDLD:cur\_ts:S:[S510050-S510500] | 28450 | LATEST、BP1、SP1 | 毫 |
| MDLD:cur\_ts:V:[C1908M02600-P2003M03400] | 415.1 418.3 415.1 28450 | LATEST BP1 SP1 PE | 毫 |

# 数据流

|  |  |  |  |
| --- | --- | --- | --- |
| 输入 | 计算 | 输出 | 函数 |
| KZ:F[IC1908-IH2003]:[LATEST|BP1|SP1] | = | MDLD:cur\_ts:F:[FIC1908-FIH2003]: :[LATEST|BP1|SP1] | run |
| KZ:S510050:[LATEST|BP1|SP1] | = | MDLD:cur\_ts:S:[S510050-S510500] [LATEST|BP1|SP1] | run |
| KZ:JZ0000KZE[510050|510300|510500]:[ NEW |S1 | B1] | = | MDLD:cur\_ts:JZ:[S510050-S510500] [LATEST|BP1|SP1] | run |
| MD:01[10001845]:[Latest|BP1|SP1|PreSettle] | If Latest == 0:  MDLD[LATEST]=MD[PreSettle]  Else:  =  **pc = LATEST**  **px = 2600\*10 or 2650 \*10**  **pcbp1 = BP1**  **pcsp1 = SP1** | MDLD:cur\_ts:OP:[C1908M02600]:[LATEST|BP1|SP1] | run |
| MD:01[10001845]:[Latest|BP1|SP1|PreSettle] | If Latest == 0:  MDLD[LATEST]=MD[PreSettle]  Else:  =  **pp = LATEST**  **px = 2600\*10 or 2650 \*10**  **ppbp1 = BP1**  **ppsp1 = SP1** | MDLD:cur\_ts:OP:[P1908M02600]:[LATEST|BP1|SP1] | run |
| KZ:[S510050|S510300|S510500]:[LATEST |SP1|BP1] | pe=KZ[LATEST]  pe\_\*\_SP1 = [SP1]  pe\_\*\_BP1 = [BP1] | pe  pe\_\*\_SP1  pe\_\*\_BP1 | run |
| KZ:JZ0000KZE[510050|510300|510500]:[NEW|S1|B1] | jz\_\*\_SP1 = KZ[S1]  jz\_\*\_BP1 = KZ[B1] | jz\_\*\_SP1  jz\_\*\_BP1 | run |
| **px pc pp** | po = px + pc – pp | MDLD:cur\_ts:PO:[1908M02600-2003M03400] | run |
| **po pe ppbp1 ppcp1** | LATEST =po – pe  SP1 = px + pcbp1 - ppsp1 - pe  BP1 = px + pcsp1 - ppbp1 - pe | MDLD:cur\_ts:A5:[1908M02600-2003M03400]: [LATEST |SP1|BP1] | run |
| KZ:F[IC1908-IH2003]:[LATEST|BP1|SP1]  pe\_\*\_SP1  pe\_\*\_BP1  jz\_\*\_SP1  jz\_\*\_BP1 | MDLD[B] = KZ[LATEST]//1000 – pe\_\*\_SP1  MDLD[S] = KZ[SP1]//1000 – pe\_\*\_BP1  MDLD[L] = KZ[LATEST]//1000 – pe  MDLD[C] = KZ[BP1]//1000 – jz\_\*\_sp1 \* 10000  MDLD[P] = KZ[LATEST]//1000 – jz\_\*\_BP1 \* 10000 | MDLD:cur\_ts:A13:[IC01-IH04]:  [B,S,L,C,R] | run |

# 168.36.1.115：6380Redis 读写要使用PIPELINE否则速度很慢，用了pipeline之后，不能用多线成了，因为pipeline是不能pickle的

# 均线界面



# 程序代码

RealTimeData/mdld.py