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

170:MD:0110001677

MDLD:ts:OP: P1909M03000

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 ｛ “A5”:5.4 , "A5B", "A5S", “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:A5:1909MV0000 2922

MDLD:ts:A5:1909MVN050 2872

MDLD:ts:A5:1909MV0050 2972

Ceiling / Floor？

均线数据: [LIST] 设计为FIFO队列,最多存4000条(屏幕分辨率)

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

......