

trc-market-data-client-sdk:1.0.1 Sample Application

This is a sample application of using trc-market-data-client-sdk:1.0.2

Dependencies

1. java 11 sdk
2. trc-market-data-client-sdk:1.0.2
3. com.opencsv:opencsv:5.1

Subscribing to fx market data

1. Create market data listener

```
public class MarketDataListener implements MarketDataCallback {
    @Override
    public void onSubscriptionFailure(MarketDataSubscriber
marketDataSubscriber, SubscriptionError subscriptionError,
                                   String message) {

        System.out.printf("onSubscriptionFailure. ErrorCode:%s.
Message:%s", subscriptionError, message);
    }

    @Override
    public void onInstrumentSnapshot(MarketDataSubscriber
marketDataSubscriber,
                                   InstrumentSnapshot
instrumentSnapshot) {

        System.out.println("On Data :" + instrumentSnapshot);
    }
}
```

2. Implement connection listener

```
public class ConnectionListener implements ConnectionCallback {  
    @Override  
    public void onConnect(MarketDataSubscriber marketDataSubscriber) {  
  
    }  
  
    @Override  
    public void onDisconnect(MarketDataSubscriber  
marketDataSubscriber, ConnectionError connectionError,  
                            String message) {  
  
    }  
  
    @Override  
    public void onConnectionFailure(MarketDataSubscriber  
marketDataSubscriber, ConnectionError connectionError,  
                                    String message) {  
  
    }  
}
```

3. Subscribe for currency pairs in "onConnect" callback

```
public class ConnectionListener implements ConnectionCallback {  
    @Override  
    public void onConnect(MarketDataSubscriber marketDataSubscriber) {  
        System.out.println("OnConnect");  
        MarketDataListener marketDataListener = new MarketDataListener();  
        marketDataSubscriber.subscribeFx("CHFJPY", marketDataListener);  
    }  
}
```

4. Create subscriber

```
MarketDataSubscriber subscriber =  
MarketDataSubscriberFactory.createSubscriber();
```

5. Connect

```
SslConfig sslConfig = new SslConfig("test-ks-path", "test-ks-pw",  
    "test-ts-path", "test-ts-pw");  
subscriber.connect("192.168.5.5", 23, "test-user", "test-pw",  
    sslConfig, connectionListener);
```

Since this is a test client all connection parameters will be ignored. Details about these parameters will be shared in the next release

6. Listen for data

```
On Data :InstrumentSnapshot{timestamp=2023-06-15T07:44:04.095054Z,  
    instrumentId='CHFJPY', bidPricePoint=PricePoint{price=153.601,  
    quantity=1000000}, askPricePoint=PricePoint{price=153.611,  
    quantity=1000000}}  
On Data :InstrumentSnapshot{timestamp=2023-06-15T07:44:05.022826Z,  
    instrumentId='CHFJPY', bidPricePoint=PricePoint{price=153.689,  
    quantity=1000000}, askPricePoint=PricePoint{price=153.709,  
    quantity=1000000}}  
On Data :InstrumentSnapshot{timestamp=2023-06-15T07:44:06.022660Z,  
    instrumentId='CHFJPY', bidPricePoint=PricePoint{price=153.613,  
    quantity=1000000}, askPricePoint=PricePoint{price=153.621,  
    quantity=1000000}}  
On Data :InstrumentSnapshot{timestamp=2023-06-15T07:44:07.022487Z,  
    instrumentId='CHFJPY', bidPricePoint=PricePoint{price=153.572,  
    quantity=1000000}, askPricePoint=PricePoint{price=153.591,  
    quantity=1000000}}
```

Supported currency pairs

- GBPJPY
- CHFJPY
- AUDJPY
- CADJPY
- NZDJPY
- SGDJPY
- EURJPY
- CNHJPY
- HKDJPY
- USDJPY