

## 유안타 증권 티 레이더(모바일) 표준 API Reference Guide

## 주요 변경 사항

| 버전   | 변경일      | 변경 내역  |
|------|----------|--|
| V1.0 | 2017.11. |  |
| V1.1 | 2018.03  | <ul style="list-style-type: none"> <li>•모의투자 로그인 추가               <ul style="list-style-type: none"> <li>-추가 : void startMotuLogin(String id, String pw)</li> <li>-내용 : 모의투자 로그인 기능 추가</li> </ul> </li> <li>•해외주식 종목 리스트 불러오기 함수추가               <ul style="list-style-type: none"> <li>-추가 : ArrayList&lt;ItemCode&gt; getNyseCodeList()<br/>ArrayList&lt;ItemCode&gt; getNasdaqCodeList()<br/>ArrayList&lt;ItemCode&gt; getAmexCodeList()<br/>ArrayList&lt;ItemCode&gt; getSortedUSAStockCodeList()<br/>ArrayList&lt;ItemCode&gt; getSanghaeCodeList()<br/>ArrayList&lt;ItemCode&gt; getShenzhenCodeList()<br/>ArrayList&lt;ItemCode&gt; getHugangtungCodeList()</li> <li>-내용 : (NYSE , NASDAQ , AMEX , 상해 , 심천 , 홍콩 종목 거래를 위한 리스트 불러오기 함수 추가</li> </ul> </li> <li>•운영/개발 서버 접속함수 삭제               <ul style="list-style-type: none"> <li>-삭제 : void SetDevSetting(String s)</li> <li>-내용 : 해당함수 삭제로 운영서버만 접속가능</li> </ul> </li> </ul> |

## SmartNetMng 함수 목록

|                           |                                      |
|---------------------------|--------------------------------------|
| initSmartNet              | SmartNet Manager 초기화                 |
| releaseSmartNet           | SmartNet Manager 해제                  |
| SetInitListener           | ISmartNetInitListener 셋팅             |
| SetLoginListener          | ISmartNetLoginListener 셋팅            |
| getInstance               | SmartNet Manager instance 값을 가져오는 함수 |
| startLogin                | 로그인 시작                               |
| startMotuLogin            | 모의투자 로그인 시작                          |
| checkAccPwd               | 계좌 비밀번호 체크                           |
| getLoginUserID            | 로그인 사용자 ID 를 구하는 함수                  |
| getAccountSize()          | 계좌리스트 갯수를 구하는 함수                     |
| getAccountNo              | 계좌번호를 구하는 함수                         |
| getAccountName            | 계좌명을 구하는 함수                          |
| getAccountInfo            | 계좌 정보를 구하는 함수                        |
| getUserCertList           | 인증서 리스트를 얻는 함수                       |
| getKospiCodeList          | 코스피 종목 리스트를 구하는 함수                   |
| getKosdaqCodeList         | 코스닥 종목 리스트를 구하는 함수                   |
| getSortedStockCodeList    | 정렬된 형태의 종목 리스트를 구하는 함수               |
| getNyseCodeList           | NYSE 종목 리스트를 구하는 함수                  |
| getNasdaqCodeList         | NASDAQ 종목 리스트를 구하는 함수                |
| getAmexCodeList           | AMEX 종목 리스트를 구하는 함수                  |
| getSortedUSASTockCodeList | 미국 종목 리스트를 구하는 함수                    |
| getSanghaeCodeList        | 상해 종목 리스트를 구하는 함수                    |
| getShenzhenCodeList       | 심천 종목 리스트를 구하는 함수                    |
| getHugangtungCodeList     | 홍콩 종목 리스트를 구하는 함수                    |

## SmartNetTranProc 함수 목록

|               |                  |
|---------------|------------------|
| initInstance  | Tran 서비스 처리기 초기화 |
| clearInstance | Tran 서비스 처리기 해제  |
| setQueryFile  | qry 파일 지정        |
| setTranId     | Tran 서비스 아이디 지정  |

|                   |   |
|-------------------|---|
| clearInputData    | 입력 데이터 초기화                                  |
| clearOutputData   | 수신 데이터 초기화                                  |
| setSingleData     | 서비스의 Input 이 단건(Single 형) 데이터 값 셋팅          |
| setMultiData      | 서비스의 Input 이 다건(Multi 형) 데이터 값 셋팅           |
| getSingleData     | 서비스의 Output 이 단건(Single 형) 데이터 값을 구하는 함수    |
| getMultiData      | 서비스의 Output 이 다건(Multi 형)값을 구하는 함수          |
| getSingleDataAttr | 서비스의 Output 이 단건(Single 형) 속성값을 구하는 함수      |
| getMultiDataAttr  | 서비스의 Output 이 다건(Multi 형) 속성값을 구하는 함수       |
| requestData       | 서버에 TR 데이터 요청                               |
| requestNextData   | 서버에 TR Next 데이터 요청                          |
| getDataCount      | 서비스의 Output 이 다건(Multi 형)일때 데이터의 갯수를 구하는 함수 |
| setCertType       | 공인인증타입 설정 (0 서명안함, 1 축약서명)                  |

#### SmartNetRealProc 함수 목록

|                      |                                  |
|----------------------|----------------------------------|
| initInstance         | Real 서비스 처리기 초기화                 |
| clearInstance        | Real 서비스 처리기 해제                  |
| setQueryFile         | Qry 파일 지정                        |
| setRealId            | Real 서비스 아이디 지정                  |
| requestReal          | 실시간 데이터 요청                       |
| releaseReal          | 실시간 데이터 해지                       |
| setRealData          | 실시간 데이터 input 데이터 지정             |
| getRealData          | 실시간 데이터 Out put 데이터 값을 구하는 함수    |
| getRealDataAttr      | 실시간 데이터 output 에서 데이터 속성을 구하는 함수 |
| getRealMultiData     | 실시간 데이터 다건 형태에서 데이터 값을 구하는 함수    |
| getRealMultiDataAttr | 실시간 데이터 다건 형태의 데이터 속성값을 구하는 함수   |

#### ISmartNetInitListener 함수 목록

|                     |                         |
|---------------------|-------------------------|
| onSessionConnecting | 서버 접속 시작시 호출되는 함수       |
| onSessionConnected  | 서버 접속 완료시 호출되는 함수       |
| onAppVersionState   | 라이브러리 버전체크 상태를 알려주는 함수  |
| onMasterDownState   | 마스터 파일 다운로드 상태를 알려주는 함수 |

|                   |                       |
|-------------------|-----------------------|
| onMasterLoadState | 마스터 파일 로딩 상태를 알려주는 함수 |
| onInitFinished    | 초기화 완료 시 호출 되는 함수     |
| onRequiredRefresh | 재접속 완료 시 호출 되는 함수     |

### ISmartNetLoginListener 함수 목록

|               |                    |
|---------------|--------------------|
| onLoginStart  | 로그인 시작을 알려주는 함수    |
| onLoginResult | 로그인 TR 결과를 알려주는 함수 |

### ITranDataListener 함수 목록

|                    |                        |
|--------------------|------------------------|
| onTranDataReceived | 조회 TR 데이터를 수신하는 함수     |
| onTranTimeout      | 조회 TR 타임아웃 발생시 호출되는 함수 |

### IRealDataListener 함수 목록

|                    |                  |
|--------------------|------------------|
| onRealDataReceived | 실시간 데이터를 수신하는 함수 |
|--------------------|------------------|

### ISmartNetAccountPwdCheckListener 함수 목록

|                         |                     |
|-------------------------|---------------------|
| onAccountPwdCheckResult | 비밀번호 검증 결과를 수신하는 함수 |
|-------------------------|---------------------|

## - SmartNetMng 함수 목록

|   |                                   |
|---|-----------------------------------|
| <b>◆ initSmartNet</b>   |                                   |
|   |                                   |
|   | . App Activity 를 SmartNet 매니저에 셋팅 |
|   |                                   |
| <b>구문</b>   |                                   |
| <pre>static void initSmartNet (     Activity activity )</pre> |                                   |

| 매개변수 |   |
|------|---|
|      | <p>activity</p> <p>App 의 activity</p>   |
| 반환값  |   |
|      | 없음  |
| 설명   |   |
|      | <p>만들고자 하는 안드로이드 App 의 . Activity 를 SmartNet 매니저에 셋팅하는 함수.</p> <p>Activity 에서 호출하는 함수</p> |
| 예제   |   |
|      | <pre>//Activity 셋팅 SmartNetMng.initSmartNet (this);</pre>                                 |

## ◆ releaseSmartNet

. Smart Net Manager 해제

### 구문

```
static void releaseSmartNet ()
```

### 매개변수

없음

### 반환값

없음

### 설명

Smart Net Manager 의 인스턴스를 해제한다.  
Activity 에서 호출하는 함수

### 예제

```
// 인스턴스 해제
SmartNetMng.releaseSmartNet();
```

## ◆ setInitListener

ISmartNetInitListener 셋팅

### 구문

```
void setInitListener (
    ISmartNetInitListener listener,
)
```

### 매개변수

listener  
ISmartNetInitListener 초기화를 알려주는 listener

### 반환값

없음

### 설명

SmartNet 매니저 초기화 및 통신 상태를 알려주는 listener 를 등록하여 사용.  
Implements ISmartNetInitListener 하여 등록하면 초기화 상황에 따라 함수를 호출한다.

### 예제

```
//Listener 셋팅
SmartNetMng.getInstance().setInitListener(sampleActivity.this);
```



|                           |  |
|---------------------------|--|
| <b>◆ setLoginListener</b> |  |
|                           |  |
|                           | ISmartNetLoginListener 셋팅  |
|                           |  |
| <b>구문</b>                 |  |
|                           | <pre>void setLoginListener (     ISmartNetLoginListener listener )</pre>   |
|                           |  |
| <b>매개변수</b>               |  |
|                           | <p>Listener</p> <p>ISmartNetLoginListener 초기화를 알려주는 listener</p>   |
|                           |  |
| <b>반환값</b>                |  |
|                           |  |
|                           |  |
| <b>설명</b>                 |  |
|                           | <p>SmartNet Manager 로그인 상태를 알려주는 listener 를 등록하여 사용.</p> <p>Implements ISmartNetLoginListener 하여 등록하면 초기화 상황에 따라 함수를 호출한다.</p> |
|                           |  |
| <b>예제</b>                 |  |
|                           | <pre>//Listener 셋팅 SmartNetMng.getInstance().setLoginListener(sampleActivity.this);</pre>                                      |

## ◆ getInstance

SmartNet Manager instance 값을 가져오는 함수

### 구문

```
SmartNetMng getInstance()
```

### 매개변수

없음

### 반환값

SmartNetMng 의 instance

### 설명

SmartNet Manager instance 값을 가져오는 함수  
SmartNetMng 의 함수를 호출 할때 사용

### 예제

```
SmartNetMng.getInstance().setLoginListener(sampleActivity.this);
```

## ◆ startLogin

로그인 시작

아이디/비번/인증서비번을 이용한 로그인과  
공인인증서/인증서 비번을 통한 로그인 2 가지 호출 방법이 있음

### 구문

```
void startLogin (
    String      strUserId
    String      strUserPW
    String      strCertPW
)

void startLogin (
    String      strCertDN
    String      strCertPW
)
```

### 매개변수

strUserId  
    사용자 로그인 ID  
strUserPW  
    사용자 로그인 비밀번호  
strCertPW  
    공인인증 비밀번호  
strCertDN  
    공인인증서 DN

### 반환값

**설명**

로그인 프로세스를 시작한다..

**예제**

```
//로그인 시작
SmartNetMng.getInstance().startLogin(m_viewLogin.getUserId(),
                                     m_viewLogin.getUserPwd(), m_viewLogin.getCertPwd());
```

**◆ startMotuLogin**

모의투자 로그인 시작

아이디/비밀번호 만으로 모의투자 로그인을 시작한다

**구문**

```
void startMotuLogin (
    String      strUserId
    String      strUserPW
)
```

**매개변수**

strUserId  
사용자 로그인 ID  
strUserPW  
사용자 로그인 비밀번호

**반환값**

|  |  |
|--|--|
|  |  |
|--|--|

**설명**

|  |                     |
|--|---------------------|
|  | 모의투자 로그인 서비스를 시작한다. |
|--|---------------------|

**예제**

|  |   |  |
|--|---|--|
|  | <pre>//모의투자 로그인 시작 SmartNetMng.getInstance().startMotuLogin(getUserId(),  getUserPwd() );</pre> |  |
|--|---|--|

## ◆ getLoginUserID

로그인 사용자 ID 를 구하는 함수

### 구문

```
String getLoginUserID ()
```

### 매개변수

### 반환값

로그인 사용자 ID

### 설명

로그인 사용자 ID 를 반환합니다.

### 예제

```
m_strUserID = SmartNetMng.getInstance().getLoginUserID();
```

## ◆ checkAccPwd

계좌비번 체크

### 구문

```
void checkAccPwd(
    String strAccNo,
    String strAccPwd,
    ISmartNetAccountPwdCheckListener listener
)
```

### 매개변수

|           |                  |
|-----------|------------------|
| strAccNo  | 계좌번호             |
| strAccPwd | 계좌비번             |
| listener  | 비번 체크 결과를 받을 리스너 |

### 반환값

없음

### 설명

사용자 계좌의 비번을 체크하여 결과를 리턴받는 함수

### 예제

```
// 계좌 비번 체크
SmartNetMng.getInstance().checkAccPwd(strAccNo, strAccPwd, Activity.this);
```

## ◆ getAccountSize

계좌리스트 갯수를 구하는 함수

### 구문

```
int getAccountSize( )
```

### 매개변수

### 반환값

계좌리스트 갯수.

### 설명

사용자 계좌리스트의 갯수를 구하는 함수.

### 예제

```
//계좌리스트  
int nCount = SmartNetMng.getInstance().getAccountSize();
```



|                       |  |
|-----------------------|--|
| <b>◆ getAccountNo</b> |  |
|                       |  |
|                       | 계좌번호를 구하는 함수   |
|                       |  |
| <b>구문</b>             |  |
|                       | <pre>String getAccountNo (                 int nIndex             )</pre>  |
|                       |  |
| <b>매개변수</b>           |  |
|                       | nIndex,<br>계좌번호 리스트 인덱스  |
|                       |  |
| <b>반환값</b>            |  |
|                       | 선택한 인덱스의 계좌번호를 반환  |
|                       |  |
| <b>설명</b>             |  |
|                       | 계좌리스트에서 인덱스의 계좌번호를 구하는 함수  |
|                       |  |
| <b>예제</b>             |  |
|                       | <pre>//계좌번호 strAcc = SmartNetMng.getInstance().getAccountNo(nIndex);</pre> |

## ◆ getAccountName

계좌명을 구하는 함수

### 구문

```
String getAccountName (
    int nIndex
)
```

### 매개변수

nIndex  
계좌 리스트 인덱스

### 반환값

선택한 인덱스의 계좌의 계좌명을 반환

### 설명

계좌리스트에서 인덱스의 계좌의 계좌명을 구하는 함수

### 예제

```
// 계좌명
strAccName = SmartNetMng.getInstance().getAccountName(nIndex);
```

|                         |   |
|-------------------------|---|
| <b>◆ getAccountInfo</b> |   |
|                         |   |
|                         | 계좌 정보를 구하는 함수   |
|                         |   |
| <b>구문</b>               |   |
|                         | <pre>String getAccountInfo (                                 String strAccNo,                                 String strInfoId</pre>        |
|                         |   |
| <b>매개변수</b>             |   |
|                         | <p>strAccNo<br/>계좌번호<br/>계좌정보 id<br/>(productname : 상품명, productcode : 상품코드, accountname : 계좌명,<br/>accounted : 계좌아이디, nickname : 계좌별명)</p> |
|                         |   |
| <b>반환값</b>              |   |
|                         | 지정된 계좌의 지정된 정보를 리턴한다  |
|                         |   |
| <b>설명</b>               |   |
|                         | 계좌 리스트에서 지정된 계좌의 정보를 얻을 때 사용함   |
|                         |   |
| <b>예제</b>               |   |
|                         | <pre>// 계좌상품코드 strAccProdCode = SmartNetMng.getInstance().getAccountInfo(sAccNo, "productcode");</pre>                                      |

## ◆ getKospiCodeList

코스피 종목 리스트를 구하는 함수

### 구문

```
ArrayList<ItemCode> getKospiCodeList()
```

### 매개변수

### 반환값

코스피 종목 ItemCode 클래스 리스트를 반환

### 설명

코스피 종목 ItemCode 클래스 리스트를 반환합니다

### 예제

```
ArrayList<ItemCode> arrItemKospiCode =
    SmartNetMng.getInstance().getKospiCodeList();
for(int i = 0; i< arrItemKospiCode.size(); i++)
{
    ItemCode itme = arrItemKospiCode.get(i);
    //종목코드
    String sCode = itme.getCode();
    //종목명
    String sName = itme.getName() ;
}
```

## ◆ getKosdaqCodeList

코스닥 종목 리스트를 구하는 함수

### 구문

```
ArrayList<ItemCode> getKosdaqCodeList()
```

### 매개변수

### 반환값

코스닥 종목 ItemCode 클래스 리스트를 반환

### 설명

코스닥 종목 ItemCode 클래스 리스트를 반환합니다

### 예제

```
ArrayList<ItemCode> arrItemKosdaqCode =
    SmartNetMng.getInstance().getKosdaqCodeList();

for(int i = 0; i < arrItemKosdaqCode.size(); i++)
{
    ItemCode itme = arrItemKosdaqCode.get(i);
    //종목코드
    String sCode = itme.getCode();
    //종목명
    String sName = itme.getName() ;
}
```

## ◆ getSortedStockCodeList

정렬된 종목 리스트를 구하는 함수

### 구문

```
ArrayList<ItemCode> getSortedStockCodeList()
```

### 매개변수

### 반환값

정렬된 주식 종목 ItemCode 클래스 리스트를 반환

### 설명

정렬된 주식 종목 ItemCode 클래스 리스트를 반환합니다

### 예제

```
ArrayList<ItemCode> arrItemSortedCode =
    SmartNetMng.getInstance().getSortedStockCodeList();

for(int i = 0; i < arrItemSortedCode.size(); i++)
{
    ItemCode itme = arrItemSortedCode.get(i);
    //종목코드
    String sCode = itme.getCode();
    //종목명
    String sName = itme.getName() ;
}
```

## ◆ getNyseCodeList

NYSE 종목 리스트를 구하는 함수

### 구문

```
ArrayList<ItemCode> getNyseCodeList()
```

### 매개변수

### 반환값

Nyse 종목 ItemCode 클래스 리스트를 반환

### 설명

Nyse 종목 ItemCode 클래스 리스트를 반환합니다

### 예제

```
ArrayList<ItemCode> arrItemNyseCode =
    SmartNetMng.getInstance().getNyseCodeList();
for(int i = 0; i< arrItemNyseCode.size(); i++)
{
    ItemCode itme = arrItemNyseCode.get(i);
    //종목코드
    String sCode = itme.getCode();
    //종목명
    String sName = itme.getName() ;
}
```

## ◆ getNasdaqCodeList

NASDAQ 종목 리스트를 구하는 함수

### 구문

```
ArrayList<ItemCode> getNasdaqCodeList()
```

### 매개변수

### 반환값

NASDAQ 종목 ItemCode 클래스 리스트를 반환

### 설명

NASDAQ 종목 ItemCode 클래스 리스트를 반환합니다

### 예제

```
ArrayList<ItemCode> arrItemNasdaqCode =
    SmartNetMng.getInstance().getNasdaqCodeList();
for(int i = 0; i< arrItemNyseCode.size(); i++)
{
    ItemCode itme = arrItemNyseCode.get(i);
    //종목코드
    String sCode = itme.getCode();
    //종목명
    String sName = itme.getName() ;
}
```



## ◆ getAmexCodeList

AMEX 종목 리스트를 구하는 함수

### 구문

```
ArrayList<ItemCode> getAmexCodeList()
```

### 매개변수

### 반환값

Amex 종목 ItemCode 클래스 리스트를 반환

### 설명

NASDAQ 종목 ItemCode 클래스 리스트를 반환합니다

### 예제

```
ArrayList<ItemCode> arrItemNasdaqCode =
    SmartNetMng.getInstance().getNasdaqCodeList();
for(int i = 0; i< arrItemNyseCode.size(); i++)
{
    ItemCode itme = arrItemNyseCode.get(i);
    //종목코드
    String sCode = itme.getCode();
    //종목명
    String sName = itme.getName() ;
}
```

## - SmartNetTranProc 함수 목록

| ◆ <b>initInstance</b> |  |
|-----------------------|--|
|                       | Tran 서비스처리를 초기화한다, ITranDataListener 셋팅  |
| <b>구문</b>             |  |
|                       | <pre> void InitInstance (                 String strTrCode,                 ITranDataListener listener )  void InitInstance (                 ITranDataListener listener </pre>  |
| <b>매개변수</b>           |  |
|                       | strTrCode<br>query 파일명<br>listener<br>ITranDataListener TR 조회데이터를 받는 listener  |
| <b>반환값</b>            |  |
|                       | 없음   |
| <b>설명</b>             |  |
|                       | <p>TR 조회 데이터를 받는 listener 를 등록하여 사용.<br/> strTrCode 에 맞는 쿼리 파일을 읽어서 초기화한다.<br/> implements ITranDataListener 하여 등록하면 TR 데이터 조회시 서버에서 내려오는 데이터를 받는다.</p> <p>strTrCode 가 지정되는 경우, strTrCode 에 지정된 TRCode + ".qry" 파일을 assets/qry/ 폴더에서 로드한다.</p> |
| <b>예제</b>             |  |

```
SmartNetTranProc m_procTran = new SmartNetTranProc();
```

```
// Tran 서비스 처리기 초기화
```

```
m_procTran.initInstance("DSO300001", this);
```

|                        |                                    |
|------------------------|------------------------------------|
| <b>◆ clearInstance</b> |                                    |
|                        |                                    |
|                        | 인스턴스 해제                            |
|                        |                                    |
| <b>구문</b>              |                                    |
|                        | <b>void clearInstance()</b>        |
|                        |                                    |
| <b>매개변수</b>            |                                    |
|                        |                                    |
|                        |                                    |
| <b>반환값</b>             |                                    |
|                        | 없음                                 |
|                        |                                    |
| <b>설명</b>              |                                    |
|                        | Tran 서비스 처리기를 해제한다.                |
|                        |                                    |
| <b>예제</b>              |                                    |
|                        | <b>m_tranProc.clearInstance();</b> |

## ◆ setQueryFile

Tran 서비스 처리기의 qry 파일을 지정

### 구문

```
void setQueryFile(
    String strTrCode
)
```

### 매개변수

strTrCode  
Tran 서비스 쿼리 코드

### 반환값

없음

### 설명

strTrCode 에 지정된 TRCode + “.qry” 파일을 assets/qry/ 폴더에서 로드한다.

### 예제

```
// 현재가 조회
SmartNetTranProc m_procTran = new SmartNetTranProc();

m_procTran.initInstance(this);
m_procTran.setQueryFile("DSO300001");
```

## ◆ setTranId

Tran 서비스 처리기의 id 를 지정

### 구문

```
void setTranId(
    String strTranId
)
```

### 매개변수

strTranId  
Tran 서비스 아이디

### 반환값

없음

### 설명

지정하지 않으면, 기본값은 initInstance 나 setQueryFile 에서 지정한 strTrCode 가 된다.  
  
같은 qry 파일을 사용하나, 종목이 다른 경우 SmartNetTranProc 인스턴스 마다 별도의 아이디를 지정하여 결과를 수신할 때 어느 인스턴스인지 구분할 수 있게 한다.

### 예제

```
// 현재가 조회
SmartNetTranProc m_procTran1 = new SmartNetTranProc();
SmartNetTranProc m_procTran2 = new SmartNetTranProc();

m_procTran1.initInstance("DSO300001", this);
m_procTran2.initInstance("DSO300001", this);

m_procTran1.setTranId("현재가 1");
m_procTran2.setTranId("현재가 2");

// 결과 수신하는 곳에서
void onTranDataReceived(String sTranID, String sMsgCode, String sMsgText)
{
    if(sTranID.equals("현재가 1"))
    {
        // m_procTran1 에 대해 처리
    } else if(sTranID.equals("현재가 2"))
    {
        // m_procTran2 에 대해 처리
    }
}
```

|                         |   |
|-------------------------|---|
| <b>◆ clearInputData</b> |   |
|                         |   |
|                         | 서비스의 Input 에 지정된 값을 초기화                               |
|                         |   |
| <b>구문</b>               |   |
|                         | <b>void clearInputData()</b>                          |
|                         |   |
| <b>매개변수</b>             |   |
|                         | 없음  |
|                         |   |
| <b>반환값</b>              |   |
|                         | 없음  |
|                         |   |
| <b>설명</b>               |   |
|                         | 서비스의 입력값에 지정된 내용을 초기화한다.                              |
|                         |   |
| <b>예제</b>               |   |
|                         | <pre>//현재가 조회 m_PriceTranProc.clearInputData();</pre> |



## ◆ clearOutputData

서비스의 Output 에 지정된 값을 초기화

### 구문

```
void clearOutputData()
```

### 매개변수

없음

### 반환값

없음

### 설명

서비스의 출력값에 지정된 내용을 초기화한다.

### 예제

```
//현재가 조회
m_PriceTranProc.clearOutputData();
```

## ◆ setSingleData

서비스의 Input 이 단건(Single 형) 데이터 값 셋팅

### 구문

```
void SetSingleData(
    int      nBlockIndex,
    int      nItemIndex,
    String    strData
)
```

### 매개변수

|             |            |
|-------------|------------|
| nBlockIndex | 블록인덱스      |
| nItemIndex  | 아이템 항목 인덱스 |
| strData     | Input 데이터  |

### 반환값

없음

### 설명

싱글블록 서비스인 경우 Input 항목에 데이터를 설정할 때 사용한다

### 예제

```
//현재가 조회
m_PriceTranProc.setSingleData(0,0, "J");
m_PriceTranProc.setSingleData(0,1, strCode);
```

## ◆ setMultiData

서비스의 Input 이 다건(Multi 형) 데이터 값 셋팅

### 구문

```
void setMultiData (
    int      nBlockIndex,
    int      nItemIndex,
    String   strData
    int      nDataIndex
)
```

### 매개변수

|             |               |
|-------------|---------------|
| nBlockIndex | 블록인덱스,        |
| nItemIndex  | 아이템 항목 인덱스    |
| strData     | Input 데이터     |
| nDataIndex  | Input 데이터 인덱스 |

### 반환값

없음

### 설명

멀티블록 서비스인 경우 Input 항목에 데이터를 설정할 때 사용한다

### 예제

```
//
m_PriceTranProc. setMultiData (0,0, strData1,0);
m_PriceTranProc. setMultiData (0,1, strData2,1);
```

## ◆ getSingleData

서비스의 Output 이 단건(Single 형) 데이터 값을 구하는 함수

### 구문

```
String getSingleData (
                int      nBlockIndex,
                int      nItemIndex,
)
```

### 매개변수

|             |            |
|-------------|------------|
| nBlockIndex | 블록인덱스      |
| nItemIndex  | 아이템 항목 인덱스 |

### 반환값

해당 블록/아이템항목에 해당하는 데이터

### 설명

싱글블록 서비스의 Output 항목의 데이터를 구할때 사용한다

### 예제

```
//현재가
String strPrice= m_TrانProc.getSingleData(0,1);
```

## ◆ getMultiData

서비스의 Output 이 다건(Multi 형)값을 구하는 함수

### 구문

```
String getMultiData (
                int      nBlockIndex,
                int      nItemIndex,
                int      nDataIndex
            )
```

### 매개변수

|             |            |
|-------------|------------|
| nBlockIndex | 블록인덱스      |
| nItemIndex  | 아이템 항목 인덱스 |
| nDataIndex  | 데이터 인덱스    |

### 반환값

해당 블록/아이템항목/데이터 인덱스에 해당하는 데이터

### 설명

멀티블록 서비스의 Output 항목의 데이터를 구할때 사용한다

### 예제

```
int  nCount = m_TranProc.GetValidCount(0);
for(int i = 0; i < nCount; i++ )
{
    strTime = m_TranProc.getMultiData(0, 0, i);    //주식 체결 시간
    strPrice = m_TranProc.getMultiData(0, 1 ,i);   //현재가
}
```

## ◆ getSingleDataAttr

서비스의 Output 이 단건(Single 형) 속성값을 구하는 함수

### 구문

```
int getSingleDataAttr (
    int nBlockIndex,
    int nItemIndex,
)
```

### 매개변수

|             |            |
|-------------|------------|
| nBlockIndex | 블록인덱스      |
| nItemIndex  | 아이템 항목 인덱스 |

### 반환값

해당 블록/아이템항목 해당하는 속성값

### 설명

싱글블록 서비스의 Output 항목의 속성값을 구할때 사용한다

### 예제

```
int nPriceAttr = m_TestTranProc.getSingleDataAttr (0,1);
```

## ◆ getMultiDataAttr

서비스의 Output 이 다건(Multi 형) 속성값을 구하는 함수

### 구문

```
String getMultiDataAttr (
    int      nBlockIndex,
    int      nItemIndex,
    int      nDataIndex
)
```

### 매개변수

|             |            |
|-------------|------------|
| nBlockIndex | 블록인덱스      |
| nItemIndex  | 아이템 항목 인덱스 |
| nDataIndex  | 데이터 인덱스    |

### 반환값

해당 블록/아이템항목/데이터 인덱스에 해당하는 속성값

### 설명

멀티블록 서비스의 Output 항목의 속성값을 구할때 사용한다

### 예제

```
int nCount = m_TranProc.GetValidCount(0);
for(int i = 0; i < nCount; i++ )
{
    Int nPriceAttr = m_TranProc.getMultiDataAttr (0, 1 ,i);
}
```

## ◆ requestData

서버에 TR 데이터 요청

### 구문

```
int requestData()
```

### 매개변수

없음

### 반환값

유니크한 조회 ID  
리턴값 < 0 : 오류

### 설명

조회 데이터를 서버에 요청하는 함수

### 예제

```
//현재가조회  
m_nPrcieRqId = m_PriceTranProc.requestData();
```



## ◆ requestNextData

서버에 TR Next 데이터 요청

### 구문

```
int requestNextData ()
```

### 매개변수

없음

### 반환값

유니크한 조회 ID

### 설명

조회 다음 데이터를 서버에 요청하는 함수

### 예제

```
if(m_TranProc.m_isMoreNextData)
{
    nRqId = m_TestTranProc.requestNextData();
}
```

## ◆ getDataCount

서비스의 Output 이 다건(Multi 형)일때 데이터의 갯수를 구하는 함수

### 구문

```
int  getDataCount (
                int      nBlockIndex)
```

### 매개변수

nBlockIndex  
블록 인덱스

### 반환값

해당 블록의 데이터의 개수

### 설명

서비스의 Output 이 다건(Multi 형)일때 데이터의 갯수를 구하는 함수

### 예제

```
int nCount =m_JangoTranProc.getDataCount(0);
for(int i = 0; i< nCount; i++)
{
}
```

## ◆ setCertType

서비스 요청 시, 로그인한 인증서로 전자서명을 하도록 지정한다

### 구문

```
void setCertType (
                int      nType)
```

### 매개변수

nType  
전자서명 방식 : 0 하지않음, 1 축약서명

### 반환값

### 설명

주문관련 TR 서비스 사용 시 공인 인증 타입을 축약으로 설정을 해야 한다.  
설정을 하지 않으면 주문이 나가지 않는다.

### 예제

```
//축약서명
m_OrderTranProc.setCertType(1);
```

## - SmartNetRealProc 함수 목록

|  |   |
|--|---|
| <b>◆ initInstance</b>                  |   |
| Real 데이터 처리기 초기화, IRealDataListener 셋팅 |   |
| <b>구문</b>                              |   |
|  | <pre>void initInstance (                     IRealDataListener listener )</pre>       |
| <b>매개변수</b>                            |   |
| listener                               | IRealDataListener 실시간 데이터를 받는 listener  |
| <b>반환값</b>                             |   |
|  | 없음  |
| <b>설명</b>                              |   |
|  | 실시간 데이터를 받는 listener 를 등록하여 사용.<br>Implements IRealDataListener 하여 등록하면 실시간 데이터를 받는다. |
| <b>예제</b>                              |   |
|  | <pre>// listener 등록 m_PriceRealProc.initInstance(this);</pre>                         |

## ◆ clearInstance

실시간 데이터 처리기 인스턴스를 해제한다.

### 구문

```
void clearInstance()
```

### 매개변수

### 반환값

없음

### 설명

실시간 데이터 처리기 인스턴스를 해제한다.

### 예제

```
m_ PriceRealProc.clearInstance();
```

## ◆ setQueryFile

실시간 서비스 처리기의 qry 파일을 지정

### 구문

```
void setQueryFile(
    String    strRealCode
)
```

### 매개변수

strRealCode  
실시간 서비스 쿼리 코드

### 반환값

없음

### 설명

strRealCode 에 지정된 RealCode + ".qry" 파일을 assets/qry/ 폴더에서 로드한다.

### 예제

```
// 현재가 실시간
SmartNetRealProc m_procReal = new SmartNetRealProc();

m_procReal.initInstance(this);
m_procReal.setQueryFile("AUTO11");
```

## ◆ setRealId

실시간 서비스 처리기의 id 를 지정

### 구문

```
void setRealId(
    String strRealId
)
```

### 매개변수

strRealId  
Real 서비스 아이디

### 반환값

없음

### 설명

지정하지 않으면, 기본값은 setQueryFile 에서 지정한 strRealCode 가 된다.

같은 qry 파일을 사용하나, 인스턴스 마다 별도의 아이디를 지정하여 결과를 수신할 때 어느 인스턴스인지 구분할 수 있게 한다.

### 예제

## ◆ requestReal

실시간 데이터를 요청하는 함수

### 구문

```
void requestReal ()
```

### 매개변수

없음

### 반환값

없음

### 설명

setRealData 로 지정된 종목 등에 대한 실시간 데이터를 요청한다.

### 예제

```
//호가 실시간 등록
m_PriceRealHogaProc.requestReal();
```



## ◆ releaseReal

실시간 데이터를 해지하는 함수

### 구문

```
void releaseReal ()
```

### 매개변수

없음

### 반환값

없음

### 설명

요청한 실시간 데이터를 해지하는 함수이다.

### 예제

```
//호가 실시간 해지
m_PriceRealHogaProc.releaseReal();
```

## ◆ setRealData

서비스의 input 에 데이터를 설정한다.

### 구문

```
void setRealData(
    int      nBlockIndex,
    int      nItemIndex,
    String    strData
)
```

### 매개변수

|             |            |
|-------------|------------|
| nBlockIndex | 블록인덱스      |
| nItemIndex  | 아이템 항목 인덱스 |
| strData     | Input 데이터  |

### 반환값

없음

### 설명

실시간 서비스에서 정보를 수신하고 하는 데이터를 설정한다.  
시세 서비스의 경우, 종목코드를 설정한다.

### 예제

```
//현재가 조회
m_PriceRealProc.setRealData(0,0, strItemCode);
```

## ◆ getRealData

실시간 데이터 Output 데이터 값을 구하는 함수

### 구문

```
String getRealData (
                int      nBlockIndex,
                int      nItemIndex
            )
```

### 매개변수

nBlockIndex  
블록인덱스  
nItemIndex  
아이템 항목 인덱스

### 반환값

해당 블록/아이템항목 데이터

### 설명

해당 인덱스의 Output 항목의 실시간 데이터를 구할때 사용한다.  
단건 데이터인 경우, 이 함수를 사용한다.

### 예제

```
// 현재간 실시간
String strPrice = m_PriceRealProc.getRealData(0,2);
```

## ◆ getRealMultiData

실시간 데이터 Out put 이 다건(multi) 형태인 경우 데이터 값을 구하는 함수

### 구문

```
String getRealMultiData (
    int      nBlockIndex,
    int      nItemIndex,
    int      nDataIndex
)
```

### 매개변수

nBlockIndex  
블록인덱스  
nItemIndex  
아이템 항목 인덱스  
nDataIndex  
여러 건 중 인덱스

### 반환값

해당 블록/아이템항목 데이터

### 설명

해당 인덱스의 Output 항목의 실시간 데이터를 구할때 사용한다.  
다건 데이터인 경우, 이 함수를 사용한다.

### 예제

// 현재간 실시간

int nDataCount = m\_PriceRealProc.getDataCount(0);

for(int i = 0; i < nDataCount; i++) {

String strPrice = m\_PriceRealProc.**getRealMultiData**(0,2, i);

}

## ◆ getRealDataAttr

실시간 데이터 Out put 속성값을 구하는 함수

### 구문

```
int getRealDataAttr (
                int      nBlockIndex,
                int      nItemIndex
)
```

### 매개변수

nBlockIndex  
블록인덱스  
nItemIndex  
아이템 항목 인덱스

### 반환값

해당 블록/아이템항목 속성값

### 설명

해당 인덱스의 Output 항목의 실시간 데이터의 속성값을 구할때 사용한다.

### 예제

```
int nAttr = m_PriceRealProc.getRealDataAttr (0,2);
```

## ◆ getRealMultiDataAttr

실시간 데이터 Output 의 다중 데이터에서 속성값을 구하는 함수

### 구문

```
int getRealMultiDataAttr (
    int nBlockIndex,
    int nItemIndex,
    int nDataIndex
```

### 매개변수

nBlockIndex  
블록인덱스  
nItemIndex  
아이템 항목 인덱스  
nDataIndex  
데이터 인덱스

### 반환값

해당 블록/아이템항목/데이터 인덱스의 속성값

### 설명

해당 인덱스의 Output 항목의 실시간 데이터의 속성값을 구할때 사용한다.

### 예제

```
int nAttr = m_PriceRealProc.getRealMutiDataAttr (0,2, 0);
```

## ◆ getDataCount

서비스의 Output 이 다건(Multi 형)일때 데이터의 갯수를 구하는 함수

### 구문

```
int  getDataCount (
                int      nBlockIndex)
```

### 매개변수

nBlockIndex  
블록 인덱스

### 반환값

해당 블록의 데이터의 개수

### 설명

서비스의 Output 이 다건(Multi 형)일때 데이터의 갯수를 구하는 함수

### 예제

```
int nCount =m_chegyulRealProc.getDataCount(1);
for(int i = 0; i< nCount; i++)
{
}
```



## - ISmartNetInitListener 함수 목록

|                              |  |
|------------------------------|--|
| <b>◆ onSessionConnecting</b> |  |
|                              |  |
|                              | 서버 접속 시작시 호출되는 함수  |
|                              |  |
| <b>구문</b>                    |  |
|                              | <pre>void onSessionConnecting ( )</pre>  |
|                              |  |
| <b>매개변수</b>                  |  |
|                              |  |
|                              |  |
| <b>반환값</b>                   |  |
|                              |  |
|                              |  |
| <b>설명</b>                    |  |
|                              | implements ISmartNetInitListener 선언하고 @Override 함수로 등록하여 사용<br>서버 접속을 시작하면 함수를 호출되어 상태를 알려준다.                                |
|                              |  |
| <b>예제</b>                    |  |
|                              | <pre>@Override public void onSessionConnecting() {     Toast.makeText(this, "서버 접속 시작.", Toast.LENGTH_SHORT).show(); }</pre> |

## ◆ onSessionConnected

서버 접속 완료시 호출되는 함수

### 구문

```
void onSessionConnected (
    boolean isSuccess,
    String strErrorMsg
)
```

### 매개변수

|             |                               |
|-------------|-------------------------------|
| isSuccess   | 서버 접속 성공여부 true:성공 / false:실패 |
| strErrorMsg | 상태 메시지                        |

### 반환값

### 설명

implements ISmartNetInitListener 선언하고 @Override 함수로 등록하여 사용  
서버 접속을 완료되면 성공여부의 상태를 알려준다.

### 예제

```
@Override
public void onSessionConnected(boolean isSuccess, String strErrorMsg)
{
    if(isSuccess == true) //서버접속 성공
        Toast.makeText(this, strErrorMsg, Toast.LENGTH_SHORT).show();
    else //서버접속 실패
        Toast.makeText(this, strErrorMsg, Toast.LENGTH_SHORT).show();
}
```

## ◆ onAppVersionState

라이브러리 버전 체크 상태를 알려주는 함수

### 구문

```
void onAppVersionState (
    boolean isDone,
)
```

### 매개변수

isDone  
버전비교여부 true:동일 / false:다름(다운로드 필요)

### 반환값

### 설명

implements ISmartNetInitListener 선언하고 @Override 함수로 등록하여 사용 라이브러리 버전을 체크하여 상태를 알려주는 함수.  
isDone 이 false 이면 프로그램이 종료되고, 새버전의 라이브러리를 받아야 실행 가능하다.

### 예제

```
@Override
public void onAppVersionState(boolean isDone)
{ Toast.makeText(this, "라이브러리 버전체크 완료.",
    Toast.LENGTH_SHORT ).show();
}
```

## ◆ onMasterDownState

마스터 파일 다운로드 상태를 알려주는 함수

### 구문

```
void onMasterDownState (
    boolean isDone,
)
```

### 매개변수

isDone  
다운로드 종료여부 true:종료

### 반환값

### 설명

implements ISmartNetInitListener 선언하고 @Override 함수로 등록하여 사용  
마스터 파일 다운로드 상태를 알려주는 함수.

### 예제

```
@Override
public void onMasterDownState(boolean isDone)
{
    Toast.makeText(this, "Master 파일 DownLoad...",
        Toast.LENGTH_SHORT ).show();
}
```

## ◆ onMasterLoadState

마스터 파일 로딩 상태를 알려주는 함수

### 구문

```
void onMasterLoadState (
    boolean isDone,
)
```

### 매개변수

isDone  
로딩 종료여부 true:종료

### 반환값

### 설명

implements ISmartNetInitListener 선언하고 @Override 함수로 등록하여 사용  
마스터 파일의 로딩 상태를 알려주는 함수

### 예제

```
@Override
public void onMasterLoadState(boolean isDone)
{ Toast.makeText(this, "Master 파일 Loading...",
    Toast.LENGTH_SHORT ).show();
}
```

## ◆ onInitFinished

초기화 완료 시 호출 되는 함수

### 구문

```
void onInitFinished (
)
```

### 매개변수

### 반환값

### 설명

implements ISmartNetInitListener 선언하고 @Override 함수로 등록하여 사용  
서버접속,버전체크,마스터파일 다운 및 로딩 작업 등의 초기화 작업이 완료되면 호출  
되는 함수.

### 예제

```
@Override
public void onInitFinished ()
{
    Toast.makeText(this, "초기화 완료.",
                    Toast.LENGTH_SHORT ).show();
}
```

## ◆ onRequiredRefresh

재접속 완료 시 호출 되는 함수

### 구문

```
void onRequiredRefresh (
)
```

### 매개변수

### 반환값

### 설명

implements ISmartNetInitListener 선언하고 @Override 함수로 등록하여 사용  
재접속 완료 시 호출 되는 함수.

### 예제

```
@Override
public void onRequiredRefresh ()
{ Toast.makeText(this, "재접속 완료.",
                Toast.LENGTH_SHORT ).show();
}
```

## - ISmartNetLoginListener 함수 목록

|                        |  |
|------------------------|--|
| <b>◆ onLoginResult</b> |  |
|                        |  |
|                        | 로그인 TR 결과를 알려주는 함수   |
| <b>구문</b>              |  |
|                        | <pre>void onLoginResult (     boolean    isSuccess,     String     strErrorMsg )</pre>   |
| <b>매개변수</b>            |  |
| isSuccess              | 성공여부 : true:성공 false:실패  |
| strErrorMsg            | 상태 메시지   |
| <b>반환값</b>             |  |
|                        |  |
| <b>설명</b>              |  |
|                        | implements ISmartNetLoginListener 선언하고 @Override 함수로 등록하여 사용<br>로그인 처리의 성공여부의 결과를 알려주는 함수.   |
| <b>예제</b>              |  |
|                        | <pre>@Override public void onLoginResult (boolean isSuccess, String strErrorMsg) { if(isSuccess == true )     Toast.makeText( getBaseContext(), "로그인처리 성공",                     Toast.LENGTH_SHORT ).show(); else     Toast.makeText( getBaseContext(), strErrorMsg,                     Toast.LENGTH_SHORT ).show();      }</pre> |



## ◆ onLoginStarted

로그인 절차가 시작되는 것을 알림

### 구문

```
void onLoginStarted ()
```

### 매개변수

없음

### 반환값

### 설명

implements ISmartNetLoginListener 선언하고 @Override 함수로 등록하여 사용

### 예제

```
@Override
public void onLoginStarted ()
{
    Toast.makeText( getBaseContext(), "로그인 시작됨",
                    Toast.LENGTH_SHORT ).show();
}
```

## - ITranDataListener 함수 목록

|                             |  |
|-----------------------------|--|
| <b>◆ onTranDataReceived</b> |  |
|                             |  |
|                             | 조회 TR 데이터를 수신하는 함수   |
|                             |  |
| <b>구문</b>                   |  |
|                             | <pre>void onTranDataReceived (                                 String      sTranID,                                 String      sMsgCode,                                 String      sMsgText                                 )</pre> |
|                             |  |
| <b>매개변수</b>                 |  |
|                             |  |
| sTranID                     | 서비스 TR 명   |
| sMsgCode                    | 메시지 코드   |
| sMsgText                    | 메시지 텍스트  |
|                             |  |
| <b>반환값</b>                  |  |
|                             | 없음   |
|                             |  |
| <b>설명</b>                   |  |
|                             | implements ITranDataListener 선언하고 @Override 함수로 등록하여 사용<br>조회 TR 데이터를 수신하는 함수<br>sTranID 로 어떤 서비스인지 알 수 있다.  |
|                             |  |
| <b>예제</b>                   |  |
|                             |  |

```
@Override
public void onTranDataReceived (String sTranID, String sMsgCode, String
    sMsgText)
{
    if(sTranID.equals("현재가")) //현재가 조회
    {
    }
}
```

## ◆ onTranTimeout

조회 TR 타임아웃 발생시 호출되는 함수

### 구문

```
void onTranTimeout (
    String sTranId
)
```

### 매개변수

sTranId  
타임아웃이 발생한 서비스의 아이디

### 반환값

### 설명

implements ITranDataListener 선언하고 @Override 함수로 등록하여 사용  
조회 TR 타임아웃 발생시 호출되는 함수

### 예제

```
@Override
public void onTranTimeout (String sTranId)
{
    Log.e("onTranTimeout", String.format("TranId:%s ", sTranId));
}
```

## - IRealDataListener 함수 목록

|                             |  |
|-----------------------------|--|
| <b>◆ onRealDataReceived</b> |  |
|                             |  |
|                             | 실시간 데이터를 수신하는 함수   |
|                             |  |
| <b>구문</b>                   |  |
|                             | <pre>void onRealDataReceived (                 String          strRealId             )</pre>                                     |
|                             |  |
| <b>매개변수</b>                 |  |
|                             |  |
| strRealId                   | 실시간 서비스 이름   |
| <b>반환값</b>                  |  |
|                             |  |
| <b>설명</b>                   |  |
|                             | implements IRealDataListener 선언하고 @Override 함수로 등록하여 사용<br>실시간 데이터를 수신하는 함수  |
|                             |  |
| <b>예제</b>                   |  |
|                             | <pre>@Override public void onRealDataReceived (String sRealId) {     //주식 종목체결 실시간 if(strRealId == "AUTO11")     {     } }</pre> |

## ISmartNetAccountPwdCheckListener 함수 목록

| ◆ onAccountPwdCheckResult |  |
|---------------------------|--|
|                           | 사용자가 입력한 계좌비번을 서버에 전송하여 암호화된 값을 생성하고, 계좌정보에 있는 비번정보와 일치하는지 체크하여 결과를 리턴한다.  |
| 구문                        |  |
|                           | <pre>void onAccountPwdCheckResult (     boolean          isSuccess,     String            strAccNo,     String            strMessage )</pre> |
| 매개변수                      |  |
| isSuccess                 | 비번 일치 여부   |
| strAccNo                  | 계좌번호   |
| strMessage                | 비번 체크 결과 메시지   |
| 반환값                       |  |
|                           |  |
| 설명                        |  |
|                           | implements ISmartNetAccountPwdCheckListener 선언하고 @Override 함수로 등록하여 사용<br>계좌 비번 체크에 대한 결과를 수신하는 함수   |
| 예제                        |  |
|                           |  |

```
@Override
public void onAccountPwdCheckResult(boolean isSuccess, String strAccNo, String
strMessage)
{
    if(isSuccess)
    {
        Toast.makeText(m_actMain, "계좌 비밀번호 체크 완료",
Toast.LENGTH_LONG).show();
    } else {
        Toast.makeText(m_actMain, strMessage, Toast.LENGTH_LONG).show();
        m_editAccountPwd.setText("");
    }
}
```