
(주)유진투자증권

SmartChamp OpenAPI 개발가이드

내용

1. 소개	3
1.1 SmartChamp OpenAPI란	3
1.2 개발환경	3
1.3 제공서비스 범위	4
2. OpenAPI 적용	4
2.1 파일 구성	4
2.2 샘플 프로젝트 구성	5
3. SmartChamp OpenAPI 동작	6
3.1 초기화 및 시작	6
3.2 종료	7
3.3 로그인	7
3.4 서비스 조회	7
3.5 실시간 조회	8
4. SmartChamp OpenAPI 명세서	9
4.1 클래스 명세서	9
4.1.1 SmartChampAPI	9
4.1.2 SmartChampTran	26
4.1.3 SmartChampReal	31
4.1.4 SmartChampDefs	34
4.2 인터페이스	35
4.2.1 INetInitDelegate	35
4.2.2 INetLoginDelegate	38
4.2.3 INetLogoutDelegate	38
4.2.4 IAccountPwdCheckDelegate	39

4.2.5	ITranDataDelegate	39
4.2.6	IRealDataDelegate	40
4.2.7	인증서 관련 함수	41
4.2.8	인증서 삭제	42

1. 소개

1.1 SmartChamp OpenAPI란

스마트 챔피언 모바일 앱의 시세 조회와 주문 등을 고객이 직접 작성할 수 있도록 제공하는 라이브러리 제품입니다. Xcode에서 a파일을 링크하여 원하는 형태의 어플리케이션을 작성할 수 있습니다.

1.2 개발환경

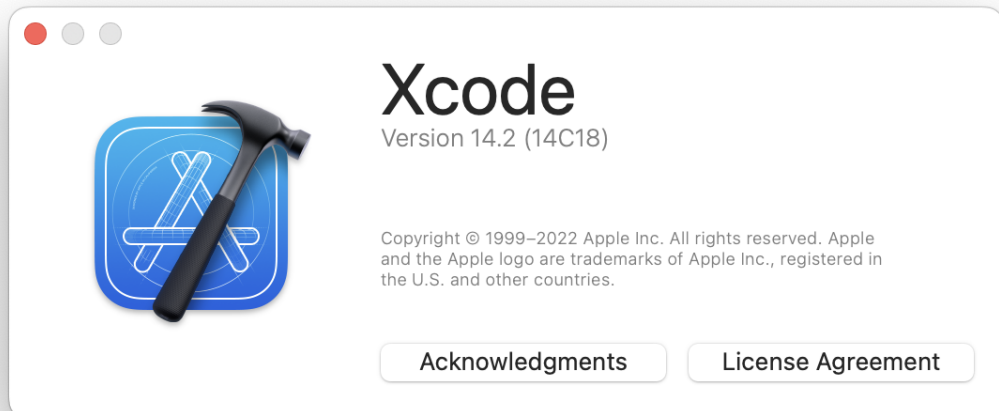
SmartChamp OpenAPI는 Xcode 스튜디오에서 Objective-C 버전으로 개발된 모듈입니다.

라이브러리 컴파일 환경의 android sdk 정보는 다음과 같습니다.

항목	내용
Xcode Version	14.2
Development target	iOS 11

SmartChamp OpenAPI 빌드환경은 다음과 같습니다. 이와 동일하게 맞춰 프로젝트를 생성하는 것은 아닙니다.

- Xcode



1.3 제공서비스 범위

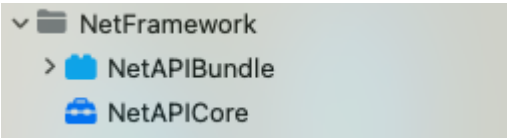
- 주식 (주식/ELW/ETF/ETN)
- 지수 선물/옵션, 주식 선물/옵션, 변동성/섹터
- 해외주식 (미국, 중국, 홍콩, (일본제외))
- 조건검색
- 공동인증서 관리(내보내기/가져오기/삭제)

2. OpenAPI 적용

2.1 파일 구성

라이브러리는 framework 형태로 제공되며, 모듈을 추가하여 다음과 같이 사용하면 됩니다.

NetAPIBundle 은 리소스 파일을 포함하고 있습니다.

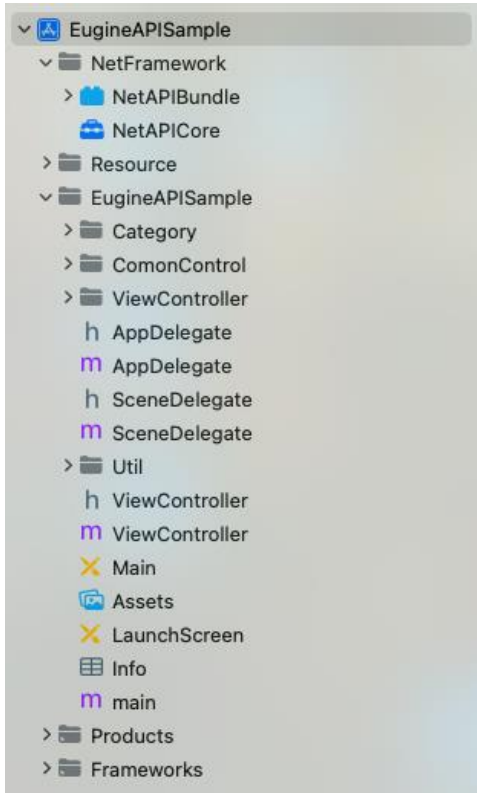


SmartChamp OpenAPI는 서버와의 통신을 위한 암호화 라이브러리와 공동인증서 관리/전자서명 작업을 위해 별도의 보안 라이브러리를 사용합니다. 그래서 OpenAPI를 사용하는 경우 보안 라이브러리까지 링크를 해줘야 합니다. 반드시 사용해야 하는 보안 라이브러리 목록은 다음과 같습니다.

라이브러리	형태	설명
Libsqlite3.tbd	tbd	Sqlite 라이브러리

2.2 샘플 프로젝트 구성

샘플 프로젝트의 구성은 다음과 같습니다.



3. SmartChamp OpenAPI 동작

SmartChamp OpenAPI를 사용하는 주요 작업별로 설명합니다.

3.1 초기화 및 시작

SmartChamp OpenAPI를 사용하기 위해 초기화 및 필요한 델리게이트 등록하고, 서버와의 통신을 위해 접속 작업을 수행합니다. 샘플 앱에 적용된 코드는 다음과 같습니다.

```
[[SmartChampAPI instance] setDevMode:SERVER_MODE_DEV];
[[SmartChampAPI instance] setInitDelegate:self];
[[SmartChampAPI instance] startConnect];
```

setDevMode함수는 앱 작성 시 테스트 서버(개발용)에 접속해서 개발할 경우 호출하는 함수입니다. 실제 운영되는 서버에 접속할 것인지 테스트를 위한 개발용 서버에 접속할 것인지 선택합니다.

setInitDelegate는 startConnect하기 전에 호출해서 델리게이트를 등록해줘야 초기 작업에 대한 이벤트를 받을 수 있습니다.

startConnect 함수를 호출하게 되면 통신 모듈이 동작하여 서버 접속 및 암호화 채널 연결, 최신 파일 업데이트, 필요 정보 로드 작업을 수행합니다. 단계별 작업 및 INetInitDelegate를 통해 호출되는 이벤트 함수는 다음과 같습니다. 단계별 리턴되는 함수에서 성공/실패에 대한 정보를 주는 경우, 성공하면 다음 단계로 넘어가고, 실패하면 그 상태에서 중지됩니다. 각각의 함수에 대한 설명은 INetInitListenr 명세서 참고 바랍니다.

초기화 단계	INetInitDelegate	설명
접속	onSessionConnecting	접속 시작을 알림
	onSessionConnected	접속 결과 알림 성공/실패 알림
업데이트	onAPIVersionState	현재 사용되는 통신API의 버전을 체크하고, 업데이트가 필요한지 판단합니다. 반드시 업데이트가 필요한 경우 상위버전의 라이브러리가 적용된 앱이 필요한 상황이므로 진행이 중지되고, 앱 종료가 요구됩니다. onRequestedTerminate()가 호출됩니다.
	onUpdateDownloadState	API에서 사용되는 파일의 업데이트가 필요한 경우 진행상황에 대한 정보를 알립니다. 자세한 사항은 INetInitDelegate 명세서 참조.
종목코드 로드	onMasterLoadState	API에서 사용하는 파일을 읽는 작업 상태를 표시합니다.
완료	onInitFinished	초기 접속 작업 완료됨
기타	onRequiredRefresh	앱 사용 중에 네트워크 단절이나 앱 비활성상태 지속 등으로 접속이 끊어진 경우, 재접속이 자동으로 이루어

		어지며, 화면 갱신을 해야 현재 상태를 유지할 수 있으므로 필요한 경우 이 함수에서 화면 갱신을 수행해야 합니다.
	onRequestedTerminate	접속이 종료되어야 하는 경우 이 함수가 호출됩니다. 앱 종료 작업을 수행하면 됩니다.

3.2 종료

SmartChamp OpenAPI를 사용하는 앱이 종료되는 시점에 인스턴스 해제 작업을 호출합니다.

```
// OpenAPI 종료
[SmartChampAPI releaseOpenAPI];
```

별도의 결과 리턴이나 델리게이트 통한 상태 알림을 하지 않습니다.

3.3 로그인

SmartChamp OpenAPI의 서비스 호출이나 계좌 목록등을 얻으려면 로그인 작업 필요합니다. 주요 입력은 아이디, 아이디비밀번호, 공동인증서 DN, 공동인증서 비밀번호, 제휴사 코드입니다. 로그인 유형에 따라 입력값이 달라지며 아래 코드를 참조 바랍니다.

setLoginDelegate로 로그인 관련 델리게이트를 설정해야 로그인 결과를 수신할 수 있습니다. 로그인 시작 시 onLoginStarted()가 호출되고, 로그인이 성공하든 실패하든 결과는 onLoginResult로 호출됩니다.

```
[[SmartChampAPI instance] setLoginDelegate:self];
//로그인 시작
if (isSiselogin){
    [[SmartChampAPI instance] loginUser:idTextField.text userPassword:sEncryptIDPassword certPassword:@"" corp:[corpbox getSelectkey]];
}else if (isCertlogin){
    [[SmartChampAPI instance] loginUserWithCertDN:certDN certPassword:sPlainCert corp:[corpbox getSelectkey]];
}else{
    [[SmartChampAPI instance] loginUser:idTextField.text userPassword:sEncryptIDPassword certPassword:sPlainCert corp:[corpbox getSelectkey]];
}
```

3.4 서비스 조회

서비스(TR)는 앱에서 서비스에 대한 객체(SmartChampTran)를 생성하고, 필요한 서비스 정보를 로드(setQueryFile)하여 SmartChampTran이 서비스 정보를 사용할 수 있게 하고, 요청에 필요한 데이터를 구성(setDataValue)하고 서버에 요청(requestData)하고, ITranDataDelegate를 통해 요청 결과를 수신하는 방식입니다.

서비스를 사용하기 위해 초기화 작업과 Delegate를 등록합니다. ITranDataDelegate에 대한 자세한 내용은 ITranDataDelegate명세서 부분을 참고 바랍니다.

setTranId로 별도의 서비스 아이디를 지정할 수 있으며, 지정하지 않으면 setQueryFile에 지정한 이름이 기본값으로 사용됩니다. ITranDataDelegate의 함수가 호출될 때, sTranID에 서비스 아이디가 들어옵니다. 한 뷰에서 SmartChampTran를 여러 개 지정해서 사용할 경우 ITranDataDelegate를 함께 사용할 수 있으며, 이벤트 함수 호출될 때 어떤 서비스인지 구분하기 위해 sTranID를 사용할 수 있습니다.


```

@protocol ITranDataDelegate <NSObject>

// 서비스 요청 전에 호출
- (void)onTranBeforeRequest:(NSString*)sTranID nextQuery:(BOOL)isNextQuery;
// 서비스 요청 결과 수신
- (void)onTranDataReceived:(NSString*)tran msgCode:(NSString*)code msgText:(NSString*)text;
// 서비스 오류 수신
- (void)onTranError:(NSString*)tran msgCode:(NSString*)code msgText:(NSString*)text;
// 서비스 타임아웃 (서버 응답 없음)
- (void)onTranTimeout:(NSString*)tran;

```

위와 같이 초기화가 이루어진 후, 입력 값 지정 후 서버에 요청한다.

```

if (stscurTran == nil){
    stscurTran = [[SmartChampTran alloc] initWithTrCode:@"stscur" delegate:self];
}
[stscurTran setTranID:@"주식현재가"];
NSString *sFullCode = [[SmartChampAPI instance] getItemFullCode:sItemCode];
[stscurTran setDataValue:@"InRec1" fieldID:@"sCode" index:0 value:sFullCode];
if ([stscurTran requestData:NO] < 0)
{
    [[iToast instance] showText:@"주식 현재가 요청에 실패 하였습니다." duration:iToastDurationLong gravity:iToastGravityBottom];
}

```

요청에 대한 결과는 ITranDataDelegate를 통해서 받습니다. 다음은 데이터 수신해서 tran를 구분하고, 결과 데이터를 얻는 방식의 예시입니다.

```

- (void)onTranDataReceived:(NSString*)tran msgCode:(NSString*)code msgText:(NSString*)text
{
    if([tran isEqualToString:@"주식 호가"]){
        {
            [self processstpbid];
            [self requestHogaReal];
            [self requestPriceReal];
        }else if ([tran isEqualToString:@"주식현재가"]){
            [self processstscurTran];
        }else if ([tran isEqualToString:@"주식매수주문"]){

```

```

8
9     NSString *name = [stscurTran getDataValue:@"OutRec1" fieldID:@"sName" index:0]; // 종목이름
10    NSString *price = [stscurTran getDataValue:@"OutRec1" fieldID:@"lCPrice" index:0]; // 현재가
11    int attrPrice = [stscurTran getDataAttr:@"OutRec1" fieldID:@"lCPrice" index:0];
12    NSString *change = [stscurTran getDataValue:@"OutRec1" fieldID:@"lDiff" index:0]; // 전일 대비
13    char attrChange = [stscurTran getDataAttr:@"OutRec1" fieldID:@"lCPrice" index:0];
14    NSString *signT = [stscurTran getDataValue:@"OutRec1" fieldID:@"cPCheck" index:0]; // 대비 부호
15    float attrPrice = [stscurTran getDataValue:@"OutRec1" fieldID:@"lLstCPrice" index:0] floatValue];
16    NSString *volumn = [stscurTran getDataValue:@"OutRec1" fieldID:@"lVolume" index:0]; // 체결량
17

```

3.5 실시간 조회

실시간 서비스(REAL)를 초기화 작업과 Delegate를 등록합니다. Delegate를 등록합니다. IRealDataDelegate에 대한 자세한 내용은 IRealDataDelegate명세서 부분을 참고 바랍니다.

setRealId로 별도의 서비스 아이디를 지정할 수 있으며 기본값으로 사용됩니다. sRealId는 조회 서비스와 동일하게 동작합니다.

```

12
13 // 서비스 요청 전 호출
14 - (void)onRealBeforeRequest:(NSString*)sRealId;
15 // 서비스 응답 후 호출
16 - (void)onRealDataReceived:(NSString*)sRealId;
17

```

초기화가 되었으면 다음과 같이 서버에 실시간 데이터를 요청합니다.

```

- (void)requestPriceReal{
    if (stockPriceReal == nil)
    {
        stockPriceReal = [[SmartChampReal alloc] initWithTrCode:@"S00" delegate:self];
    }
    [stockPriceReal releaseReal];

    NSString *sFullCode =[stpbidTran getDataValue:@"OutRec1" fieldID:@"sCode" index:0];
    [stockPriceReal setDataValue:@"InBlock1" fieldID:@"sCode" index:0 value:sFullCode];
    [stockPriceReal requestReal];
}

```

서버에 요청 후 데이터가 갱신될 때마다 onRealDataReceived가 호출되며 여기서 데이터를 얻을 수 있습니다. 실시간 데이터가 더 이상 필요없는 경우 releaseReal을 호출하면 이전에 설정한 값으로 실시간 데이터 수신을 해제 할 수 있습니다.

4. SmartChamp OpenAPI 명세서

4.1 클래스 명세서

4.1.1 SmartChampAPI

SmartChamp OpenAPI를 사용하기 위한 기본이 되는 클래스이며, 이 클래스에 정의된 함수는 다음과 같습니다.

분류	반환	함수명	설명
초기화	void	instance	API 인스턴스 초기화
	void	releaseOpenAPI	API 인스턴스 해제
	void	setDevMode	운영/개발 접속 서버 선택
델리게이트 등록	void	setInitDelegate	접속 처리 결과 델리게이트 등록
	void	setLoginDelegate	로그인 결과 델리게이트 등록
	void	setLogoutDelegate	로그아웃 결과 델리게이트 등록
초기화시작	void	startConnect	초기 접속 시작
로그인 로그아웃	void	loginUser	로그인 요청
	void	logoutUser	로그아웃
사용자	NSString	getLoginUserId	사용자 아이디
	int	getLoginType	현재 로그인 유형 반환

계좌	void	checkAccPwd	계좌비밀번호 체크
	int	getAccountSize	계좌 개수 리턴
	NSString	getAccountNo	계좌번호 리턴
	NSString	getAccountName	계좌명 반환
	NSString	getAccountInfo	계좌정보 반환
인증서	Array	getUserCertList	사용자 인증서 리스트 반환
	NSString	getUserCertInfo	인증서 정보 리턴
	NSString	requestCertImportNumber	인증서 가져오기 랜덤번호 생성
	NSString	requestImportCert	인증서 가져오기 실행
	NSString	requestExportCert	인증서 내보내기 실행
	NSString	requestDeleteCert	인증서 삭제
종목	NSArray	getKospiCodeList	코스피 종목 리스트 반환
	NSArray	getKosdaqCodeList	코스닥 종목 리스트 반환
	NSArray	getStockCodeList	주식(코스피/코스닥) 종목 리스트 반환
	NSArray	getETFCodeList	ETF 종목 리스트 반환
	NSArray	getETNCodeList	ETN 종목 리스트 반환
	NSArray	getFuturesCodeList	선물 종목 리스트 반환
	NSArray	getOptionsCodeList	옵션 종목 리스트 반환
	NSArray	getOverseasStockCodeList	해외주식 종목 리스트 반환
	NSArray	getItemCodeList	종목 코드 리스트 반환(타입별)
	NSString	getItemCodeType	종목 시장구분 반환
	NSString	getItemFullCode	종목 표준 코드 반환
	NSString	getItemShortCode	종목 축약 코드 반환
	NSString	getItemMarketTypeText	시장 구분명 반환
	NSString	getItemCodeInfo	종목 정보 반환
키패드	NSString	getSecKeyPublicKey	보안키패드용 공개키를 리턴
암호화	NSData	encryptData	데이터 암호
	NSData	decryptData	데이터 복호

각각의 함수의 설명은 다음과 같습니다.

- 초기화

함수명	-(id)instance		
설명	SmartChamp OpenAPI초기화, 처음실행되는 AppDelegate에서 한번만 호출하면 됩니다		
리턴값	없음		
파라미터	-		
내용	API사용 전에 언제든지 한번은 호출해야 합니다. [SmartChampAPI instance]		

함수명	+(void)releaseOpenAPI	
설명	API 해제, 앱 종료 전에 호출하면 됩니다.	
리턴값	없음	
파라미터	없음	
내용	<p>앱 종료 시점에 한번 호출합니다.</p> <pre>// OpenAPI 종료 [SmartChampAPI releaseOpenAPI];</pre>	

함수명	(void)setDevMode:(NSString*)strDev	
설명	접속하는 서버를 선택하는 기능입니다.	
리턴값	없음	
파라미터	NSString strDev	운영/테스트 서버 선택
내용	<p>반드시 SmartChampAPI 초기화 전에 호출해야 합니다. 이 함수로 선택한 서버로 접속과 내부의 운영/테스트 선택 동작이 이루어 집니다.</p> <pre>[[SmartChampAPI instance] setDevMode:SERVER_MODE_DEV];</pre> <p>입력값은 SmartChampDefs에 정의되어 있습니다.</p> <pre>#define SERVER_MODE_DEV @"1" #define SERVER_MODE_REAL @"0"</pre>	

- 델리게이트 등록

함수명	- (void) setInitDelegate:(id <INetInitDelegate>)delegate	
설명	세션 초기화, 접속 상태, 초기 업데이트 상태 등에 대한 결과를 통보받기 위한 델리게이트를 지정합니다.	
리턴값	없음	
파라미터	INetInitDelegate Delegate	API 초기 접속 처리 결과를 받기 위한 델리게이트
내용	SmartChamp OpenAPI 초기화 작업 후 startConnect 호출 전에 델리게이트를 등록해야 하며, 1개의 인스턴스만 등록해야 합니다.	

```

    * 서버 접속 시작
    */
    - (void)onSessionConnecting;;

    /**
     * 서버 접속 완료
     * @param isSuccess 성공여부
     * @param strErrorMsg 에러메세지
     */
    - (void)onSessionConnected:(BOOL)isSuccess error:(NSString*)strErrorMsg;

    /**
     * API 버전체크
     * param nCheckState 업데이트 필요 유형 ( -1 : 오류, 0 : 필요없음, 1 : 권고, 2 : 강제업데이트 )
     * param sUpdateInfo
     */
    - (void)onAPIVersionState:(NSInteger)nCheckState update:(NSString*)sUpdateInfo;

    /**
     * 파일 다운로드 작업 상태
     * param nState 다운로드 상태 ( 0 : 진행중, 1 : 상태알림, 2 : 완료 )
     * param nIndex 진행상태
     * param nCount 전체갯수
     * param sMessage 메시지 ( 상태 알림 )
     */
    - (void)onUpdateDownloadState:(NSInteger)nState index:(NSInteger)nIndex count:(NSInteger)nCount message:

    /**
     * 마스터 로드 작업 상태
     * @param nState 로드 상태 ( 0 : 시작, 1 : 완료 )
     */
    - (void)onMasterLoadState:(NSInteger)nState;

    /**
     * 모든 초기화 작업 완료된 상태
     */
    - (void) onInitFinished;

    /**
     * 재접속
     */
    - (void) onRequiredRefresh;

```

델리게이트 내의 함수 호출에 대한 내용은 인터페이스 INetInitDelegate 설명 부분 참조 바랍니다.

함수명	(void) setLoginDelegate:(id <INetLoginDelegate>)delegate	
설명	로그인 요청에 대한 결과를 리턴 받기 위한 델리게이트 등록	
리턴값	없음	
파라미터	INetLoginDelegate Delegate	로그인 결과를 받기 위한 델리게이트
내용	요청한 로그인에 대한 시작과 로그인 작업 완료 시 성공여부와 실패 시 실패메시지를 받을 수 있습니다.	

```

    * 서버 접속 시작
    */
- (void)onSessionConnecting;;

/**
 * 서버 접속 완료
 * @param isSuccess 성공여부
 * @param strErrorMsg 에러메세지
 */
- (void)onSessionConnected:(BOOL)isSuccess error:(NSString*)strErrorMsg;

/**
 * API 버전체크
 * param nCheckState 업데이트 필요 유형 ( -1 : 오류, 0 : 필요없음, 1 : 권고, 2 : 강제업데이트 )
 * param sUpdateInfo
 */
- (void)onAPIVersionState:(NSInteger)nCheckState update:(NSString*)sUpdateInfo;

/**
 * 파일 다운로드 작업 상태
 * param nState 다운로드 상태 ( 0 : 진행중, 1 : 상태알림, 2 : 완료 )
 * param nIndex 진행상태
 * param nCount 전체갯수
 * param sMessage 메시지 ( 상태 알림 )
 */
- (void)onUpdateDownloadState:(NSInteger)nState index:(NSInteger)nIndex count:(NSInteger)nCount message:(NSString*)sMe

/**
 * 마스터 로드 작업 상태
 * @param nState 로드 상태 ( 0 : 시작, 1 : 완료 )
 */
- (void)onMasterLoadState:(NSInteger)nState;

/**
 * 모든 초기화 작업 완료된 상태
 */
- (void) onInitFinished;

/**
 * 재접속
 */
- (void) onRequiredRefresh;

```

델리게이트 내의 함수 호출에 대한 내용은 인터페이스 INetLoginDelegate 설명 부분 참조 바랍니다.

함수명	- (void) setLogoutDelegate:(id <INetLogoutDelegate>) delegate	
설명	로그아웃 요청 시, 로그아웃 결과를 리턴 받을 수 있는 델리게이트를 등록	
리턴값	없음	
파라미터	INetLogoutDelegate Delegate	로그아웃 결과를 받을 델리게이트
내용	로그아웃 요청에 대해 처리 완료를 받을 수 있다.	

```

/**
 * 로그인 시작됨
 */
- (void)onLoginStarted;

/**
 * 로그인 TR
 * @param isSuccess 성공여부
 * @param strErrorMsg 에러메세지
 */
- (void)onLoginResult:(BOOL)isSuccess error:(NSString*)strError;

```

델리게이트 내의 함수 호출에 대한 내용은 인터페이스 INetLogoutDelegate 설명 부분 참조 바랍니다.

- 초기 접속 시작

함수명	- (void)startConnect
설명	OpenAPI로 서버에 접속을 요청합니다.
리턴값	없음 처리 결과는 INetInitDelegate를 통해서 받습니다.
파라미터	없음
기타	초기화 및 델리게이트 등록이 완료 된 후, 서버에 접속을 시작합니다. 접속 이후, API버전체크 및 최신 업데이트가 필요한 파일 다운로드 및 초기 로그인 작업 등이 처리됩니다. 접속시작부터 완료 시점까지 단계 별로 INetInitDelegate를 통해서 처리 상태가 이벤트 함수를 통해 리턴됩니다.

- 로그인/로그아웃

함수명	- (void)loginUser:(NSString*)userid userPassword:(NSString*)userPW certPassword:(NSString*)certPW corp:(NSString*)corp; - (void)loginUserWithCertDN:(NSString*)certDN certPassword:(NSString*)certPW corp:(NSString*)corp;	
설명	로그인 유형에 따라 로그인 요청을 합니다.	
리턴값	없음 처리 결과는 INetLoginDelegate를 통해서 받습니다.	
파라미터	NSString userId	사용자 아이디 (대문자,숫자)
	NSString userPw	사용자 아이디비번
	NSString certPw	공동인증서 패스워드

	NSString certDN	공동인증서 DN (인증서 정보)
	NSString corp	제휴사 코드
내용	<p>SmartChamp OpenAPI에서 로그인은 조회(시세전용)용 로그인([사용자아이디+아이디비번])과 거래용 로그인([공동인증서DN+인증서비번] 또는 [아이디+아이디비번+공동인증서비번]) 2가지 방식으로 로그인 합니다.</p> <p>거래용 로그인 [공동인증서DN + 인증서비번] 방식이 일반적으로 사용되고, 공동인증서 1개에 여러 아이디가 생성된 경우, 공동인증서DN만으로 사용자아이디를 특정할 수 없으므로 [아이디+아이디비번+공동인증서비번] 방식으로 로그인해야 합니다.</p> <p>조회용 로그인 loginUser(sUserId, sUserPwd, "", sCorpCode)와 같이 공동인증서 비번을 입력하지 않게 되면 조회용 로그인으로 동작합니다.</p>	

함수명	(void) logoutUser;;	
설명	로그인 안한 상태(로그아웃) 상태로 전환합니다.	
리턴값	없음 처리가 완료되면 INetLogoutDelegate를 통해서 완료가 되었음을 받을 수 있습니다.	
파라미터	없음	
내용	로그인한 상태에서 호출 시 내부에 가지고 있는 로그인 정보를 삭제하고 서버에 로그아웃을 알립니다.	

● 사용자 정보

함수명	- (NSString*)getLoginUserID	
설명	현재 로그인한 사용자의 아이디를 얻습니다.	
리턴값	사용자 아이디	
파라미터	없음	
내용	현재 로그인한 사용자의 아이디를 리턴합니다.	

함수명	(LOGIN_MODE) getLoginType;										
설명	현재 로그인한 로그인 유형을 얻습니다.										
리턴값	로그인 유형										
파라미터	없음										
내용	로그인 유형은 SmartChampDefs클래스에 정의되어 있습니다.										
	<table border="1"> <thead> <tr> <th>유형정의</th><th>값</th><th>설명</th></tr> </thead> <tbody> <tr> <td>LO N_TYPE_NONE</td><td>0</td><td>로그인 안됨</td></tr> <tr> <td>LOGIN_TYPE_AUTO</td><td>1</td><td>자동로그인</td></tr> </tbody> </table>	유형정의	값	설명	LO N_TYPE_NONE	0	로그인 안됨	LOGIN_TYPE_AUTO	1	자동로그인	
유형정의	값	설명									
LO N_TYPE_NONE	0	로그인 안됨									
LOGIN_TYPE_AUTO	1	자동로그인									

			로그인 후 로그아웃 하지 않고, 앱을 종료한 경우 후에 앱 실행 시 이전 아이디로 로그인 됨 시세 조회만 가능
	LOGIN_TYPE_SISE	2	조회용 로그인 주문 서비스 호출은 불가하며, 시세조회 계좌조회만 가능
	LOGIN_TYPE_CRT	3	주문용 로그인 공동인증서를 통해 로그인 한 경우 주문 서비스 호출 가능

- 계좌

함수명	(void) checkAccPwd:(NSString*)strAccNo accPWD:(NSString*)strAccPwd delegate:(id<IAccountPwdCheckDelegate>)delegate	
설명	계좌번호에 대한 계좌비밀번호가 맞는지 체크합니다.	
리턴값	없음 체크 결과는 IAccountPwdCheckDelegate를 통해서 받습니다.	
파라미터	NSString strAccNo	계좌번호
	NSString strAccPwd	계좌비번
	IAccountPwdCheckDelegate Delegate	비밀번호 체크 결과를 리턴받을 델리게이트
내용	OpenAPI를 사용하여 작성하는 앱에서 계좌비번 체크가 필요한 경우 사용합니다. (void)onAccountPwdCheckResult:(BOOL)isSuccess accno:(NSString*)strAccNo message:(NSString*)strMessage	

함수명	- (NSInteger)getAccountSize		
설명	로그인한 사용자의 계좌갯수를 반환합니다..		
리턴값	계좌번호 개수		
파라미터	없음		
내용	사용자 계정의 계좌 중에 OpenAPI 사용 등록이 된 계좌의 개수를 가져옵니다. <div><pre>NSInteger nAccountSize = [[SmartChampAPI instance] getAccountSize]; NSMutableArray *arrayAccountNo = [NSMutableArray array]; NSMutableArray *arrayAccountName = [NSMutableArray array]; for (NSInteger i = 0; i < nAccountSize; i++) { NSString *sAccountNO = [[SmartChampAPI instance] getAccountNo:i]; NSString *sAccountName = [[SmartChampAPI instance] getAccountName:i]; [arrayAccountNo addObject:sAccountNO]; [arrayAccountName addObject:[NSString stringWithFormat:@"%s(%s)", sAccountNO, sAccountName]]; }</pre></div>		

함수명	-(NSString)getAccountName:(NSString*)strAccNo	
설명	계좌번호에 해당하는 계좌의 계좌명을 리턴합니다.	
리턴값	주어진 계좌번호의 계좌명	
파라미터	NSString strAccNo	계좌번호
내용	계좌명을 검색하여 리턴합니다.	

- 공동 인증서

함수명	- (NSArray*)getUserCertList	
설명	사용하는 앱에 저장된 공동인증서 정보 리스트를 리턴합니다.	
리턴값	인증서 리스트	
파라미터	없음	
내용	<p>인증서 리스트를 얻습니다. 인증서의 정보값이 NSString 구분자 형태로 리턴됩니다.</p> <pre> arrayCertList = [NSMutableArray arrayWithArray:[[SmartChampAPI instance] getUserCertList]]; NSString* strRet = [arrayCertList objectAtIndex:indexPath.row]; NSArray *arrayRet = [strRet componentsSeparatedByString:@":"]; NSString *strUserID = [arrayRet objectAtIndex:0]; // 사용자 명 NSString *strDN = [arrayRet objectAtIndex:1]; // 사용자 DN NSString *strUsage = [arrayRet objectAtIndex:2]; // 용도/구분 NSString *strIssuerName = [arrayRet objectAtIndex:3]; // 발급자명 NSString *strIssuer = [arrayRet objectAtIndex:4]; // 발급자DN NSString *strExpired = [arrayRet objectAtIndex:5]; // 만료여부 NSString *strExpreDate = [arrayRet objectAtIndex:6]; // 만료일 NSString *strSerialNum = [arrayRet objectAtIndex:7]; // 인증서 일련번호 </pre>	

함수명	- (NSString*)requestCertImportNumber	
설명	공동인증서 가져오기 1단계 - 랜덤번호 생성	
리턴값	성공여부`랜덤숫자1`랜덤숫자2`랜덤숫자3	
파라미터	없음	
내용	<p>공동인증서 가져오기 작업은 2단계로 이루어진다.</p> <p>1단계는 내보내기하는 단말에서 입력할 랜덤번호를 생성한다.</p> <p>2단계는 내보내기하는 단말에서 랜덤번호를 입력하고 내보내기 요청을 한 후에 가져오기 작업을 할 단말에서 가져오기 요청을 한다.</p> <p>이 함수는 1단계 작업을 위한 랜덤번호 생성하는 함수이다. 2단계 작업은 다음목록의 importCert함수를 사용한다.</p>	

	<pre> NSString *sSimple = [[SmartChampAPI instance] requestCertImportNumber]; NSArray *arraySimple = [sSimple componentsSeparatedByString:@"`"]; NSString *sNumber1 = nil; NSString *sNumber2 = nil; NSString *sNumber3 = nil; NSArray *arrayNumber = nil; if ([[arraySimple objectAtIndex:0] isEqualToString:@"Success"]){ NSString *simpleNumber = [arraySimple objectAtIndex:1]; sNumber1 = [simpleNumber substringToIndex:4]; sNumber2 = [simpleNumber substringWithRange:NSMakeRange(4, 4)]; sNumber3 = [simpleNumber substringWithRange:NSMakeRange(8, 4)]; arrayNumber = [NSArray arrayWithObjects:sNumber1, sNumber2, sNumber3, nil]; }else{ </pre> <p>requestCertImportNumber에 random으로 생성된 랜덤번호를 리턴 받는다. 이 번호를 내보내기하는 단말에서 입력하면 된다.</p> <p>리턴 값은 "" 로 구분자 처리 하며 첫번째에 성공 실패 여부가 있다. 실패일 경우 1 두 번째 구분자에 실패의 메시지가 있다..</p>
--	---

함수명	-(NSString) importCert
설명	공동인증서 가져오기 2단계 - 공동인증서 수신 후 로컬에 저장
리턴값	성공실패여부`메시지
파라미터	없음
내용	<p>공동인증서 가져오기 2단계 작업 중 2단계에 해당하는 함수이다.</p> <p>인증서 내보내기를 하는 단말에서 랜덤번호를 입력 후 내보내기 요청한 후에, 이 함수를 호출하여 단말에 인증서를 저장한다.</p> <pre> NSString *sResult = [[SmartChampAPI instance] importCert]; NSArray *arrayResult = [sResult componentsSeparatedByString:@"`"]; if ([[arrayResult objectAtIndex:0] isEqualToString:@"Success"]){ UIAlertController *alert = [UIAlertController alertController [self presentViewController:alert animated:YES completion:^{ }]]; }else{ </pre> <p>가져오기가 완료되면 리턴으로, 성공하면 구분자로 Success로 반환되면 지정된 공동인증서가 가져오기가 완료된 상태임, 실패(Failed)하면, 두번째 구분자에 실패사유가 저장되어 있음.</p>

함수명	- (NSString*) exportCert:(NSString*)strCertDN simpleCertNum:(NSString*)strSimpleCertNum
설명	인증서 내보내기 작업을 수행한다.

리턴값	NSString 성공실패 여부`메시지	
파라미터	NSString strCertDN	내보내기 할 공동인증서 DN
	NSString strSimpleCertNum	인증서 가져오기에서 생성된 랜덤번호
내용	공동인증서 내보내기는 가져오기를 수행할 단말에서 생성된 랜덤번호를 입력받아, 지정된 인증서를 중계서버로 전송한다.	

함수명	- (NSString*) deleteCert:(NSString*)strCertDN certPW:(NSString*)certPW	
설명	단말에 저장된 공동인증서를 삭제한다.	
리턴값	NSString 성공실패 여부`메시지	
파라미터	NSString strCertDN	삭제할 공동인증서 DN
	NSString certPW	삭제할 인증서의 비번
내용	로컬에 저장된 공동인증서를 삭제한다. 입력한 DN에 해당하는 인증서의 비밀번호를 검증하고, 통과되면 로컬의 파일을 삭제한다. 삭제 후 리턴값으로 결과를 받는다.	

● 종목 리스트 / 종목 정보

함수명	ArrayList< id <IStructItemCode> > getKospiCodeList()	
설명	코스피 종목코드 리스트를 얻는다.	
리턴값	이름으로 정렬된 코스피 종목 리스트	
파라미터	없음	
내용	코스피 종목 리스트를 반환합니다.	

함수명	ArrayList<NSString> getKodakCodeList()	
설명	코스닥 종목코드 리스트를 얻는다.	
리턴값	이름으로 정렬된 코스닥 종목 리스트	
파라미터	없음	
내용	코스닥 종목 리스트를 반환합니다. <pre>id <IStructItemCode> item = (id <IStructItemCode>)[arraySearchedStockItem ob [titleLabel setText:[item getCode]]; [nameLabel setText:[item getName]];</pre>	

함수명	ArrayList< id <IStructItemCode> > getStockCodeList()	
설명	국내주식 종목코드(코스피+코스닥) 리스트를 얻는다.	

리턴값	이름으로 정렬된 주식 종목 리스트	
파라미터	없음	
내용	국내 주식 종목 리스트를 반환합니다.	

함수명	ArrayList< id <IStructItemCode>> getETFCodeList()	
설명	ETF 종목코드 리스트를 얻는다.	
리턴값	이름으로 정렬된 ETF 종목 리스트	
파라미터	없음	
내용	ETF 종목 리스트를 반환합니다.	

함수명	ArrayList< id <IStructItemCode>> getETNCodeList()	
설명	ETN 종목코드 리스트를 얻는다.	
리턴값	이름으로 정렬된 ETN 종목 리스트	
파라미터	없음	
내용	ETN 종목 리스트를 반환합니다.	

함수명	ArrayList< id <IStructItemCode>> getFuturesCodeList()	
설명	국내지수선물 종목코드 리스트를 얻는다.	
리턴값	지수선물 종목 리스트	
파라미터	없음	
내용	지수선물 종목 리스트를 반환합니다.	

함수명	ArrayList<NSString> getOptionsCodeList()	
설명	국내 지수옵션 종목코드 리스트를 얻는다.	
리턴값	지수옵션 종목 리스트	
파라미터	없음	
내용	지수옵션 종목 리스트를 반환합니다.	

함수명	ArrayList< id <IStructItemCode>> getOverseasStockCodeList()	
-----	---	--

설명	해외주식 종목코드 리스트를 얻는다.	
리턴값	해외주식 종목 리스트	
파라미터	없음	
내용	해외주식 종목 전체(미국,중국,홍콩) 리스트를 반환합니다.	

함수명	ArrayList< id <IStructItemCode>> getItemCodeList(NSString strType)																																																									
설명	주어진 유형에 맞는 종목코드 리스트를 얻는다.																																																									
리턴값	종목 리스트																																																									
파라미터	NSString strType	종목 유형																																																								
내용	strType에 따른 종목 리스트를 반환한다. strType에 들어갈 수 있는 값은 다음과 같다. <table><tr><td>strType</td><td>유형 설명</td></tr><tr><td>KOSP</td><td>주식 - 코스피</td></tr><tr><td>KOSDAQ</td><td>주식 - 코스닥</td></tr><tr><td>STOCK</td><td>주식 - 코스피 + 코스닥</td></tr><tr><td>CD</td><td>주식 - CFD</td></tr><tr><td>ELW</td><td>ELW</td></tr><tr><td>ETN</td><td>ETN</td></tr><tr><td>ET</td><td>ET</td></tr><tr><td>WARRANTS</td><td>신주인수권</td></tr><tr><td>KONE</td><td>KNEX</td></tr><tr><td>KOTC</td><td>K-OTC</td></tr><tr><td>K20FUT</td><td>지수선물</td></tr><tr><td>K20OPT</td><td>지수옵션</td></tr><tr><td>MINIFUT</td><td>미니선물</td></tr><tr><td>MINIOPT</td><td>미니옵션</td></tr><tr><td>WEEKLYOPT</td><td>위클리옵션</td></tr><tr><td>KRX300FUT</td><td>KRX300선물</td></tr><tr><td>KSD150FUT</td><td>코스닥150선물</td></tr><tr><td>KSD150OPT</td><td>코스닥150옵션</td></tr><tr><td>STKFUT</td><td>주식선물</td></tr><tr><td>VFUTURE</td><td>변동성</td></tr><tr><td>TFUTURE</td><td>섹터 선물</td></tr><tr><td>CM</td><td>야간선물</td></tr><tr><td>BOND</td><td>채권</td></tr><tr><td>UPJONG</td><td>업종</td></tr><tr><td>WSTOC</td><td>해외주식 - 전체</td></tr><tr><td>WSTOCK_US</td><td>해외주식 - 미국 전체</td></tr><tr><td>WSTOCK_NY</td><td>해외주식 - 미국 - 뉴욕</td></tr></table>		strType	유형 설명	KOSP	주식 - 코스피	KOSDAQ	주식 - 코스닥	STOCK	주식 - 코스피 + 코스닥	CD	주식 - CFD	ELW	ELW	ETN	ETN	ET	ET	WARRANTS	신주인수권	KONE	KNEX	KOTC	K-OTC	K20FUT	지수선물	K20OPT	지수옵션	MINIFUT	미니선물	MINIOPT	미니옵션	WEEKLYOPT	위클리옵션	KRX300FUT	KRX300선물	KSD150FUT	코스닥150선물	KSD150OPT	코스닥150옵션	STKFUT	주식선물	VFUTURE	변동성	TFUTURE	섹터 선물	CM	야간선물	BOND	채권	UPJONG	업종	WSTOC	해외주식 - 전체	WSTOCK_US	해외주식 - 미국 전체	WSTOCK_NY	해외주식 - 미국 - 뉴욕
strType	유형 설명																																																									
KOSP	주식 - 코스피																																																									
KOSDAQ	주식 - 코스닥																																																									
STOCK	주식 - 코스피 + 코스닥																																																									
CD	주식 - CFD																																																									
ELW	ELW																																																									
ETN	ETN																																																									
ET	ET																																																									
WARRANTS	신주인수권																																																									
KONE	KNEX																																																									
KOTC	K-OTC																																																									
K20FUT	지수선물																																																									
K20OPT	지수옵션																																																									
MINIFUT	미니선물																																																									
MINIOPT	미니옵션																																																									
WEEKLYOPT	위클리옵션																																																									
KRX300FUT	KRX300선물																																																									
KSD150FUT	코스닥150선물																																																									
KSD150OPT	코스닥150옵션																																																									
STKFUT	주식선물																																																									
VFUTURE	변동성																																																									
TFUTURE	섹터 선물																																																									
CM	야간선물																																																									
BOND	채권																																																									
UPJONG	업종																																																									
WSTOC	해외주식 - 전체																																																									
WSTOCK_US	해외주식 - 미국 전체																																																									
WSTOCK_NY	해외주식 - 미국 - 뉴욕																																																									

	WSTOCK_NS	해외주식 - 미국 - 나스닥
	WSTOCK_AM	해외주식 - 미국 - 아멕스
	WSTOCK_CN	해외주식 - 중국 전체
	WSTOCK_SH	해외주식 - 중국 - 상해
	WSTCK_SZ	해외주식 - 중국 - 심천
	WSTOCK_HK	해외주식 - 홍콩 전체
	WUPJON	해외업종

함수명	- (NSArray*) getOverseasStockCodeList:(NSString*)nation exchange:(NSString*)exchange																						
설명	해외주식 종목에서 주어진 국가와 거래소에 해당하는 종목 리스트 반환																						
리턴값	해외주식 종목 리스트																						
파라미터	NSString strNation	국가 구분																					
	NSString strExchange	거래소 구분																					
내용	<p>원하는 국가와 거래소 구분값으로 종목 리스트 반환 구분값은 다음과 같다.</p> <table border="1"> <thead> <tr> <th>strNation</th><th>strExchang</th><th>종목</th></tr> </thead> <tbody> <tr> <td>US</td><td>NYS</td><td>미국 - 뉴욕</td></tr> <tr> <td>US</td><td>AMX</td><td>미국 - 아멕스</td></tr> <tr> <td>S</td><td>NAS</td><td>미국 - 나스닥</td></tr> <tr> <td>CN</td><td>SHS</td><td>중국 - 상해</td></tr> <tr> <td>CN</td><td>SZS</td><td>중국 - 심천</td></tr> <tr> <td>HK</td><td>HKS</td><td>홍콩</td></tr> </tbody> </table>		strNation	strExchang	종목	US	NYS	미국 - 뉴욕	US	AMX	미국 - 아멕스	S	NAS	미국 - 나스닥	CN	SHS	중국 - 상해	CN	SZS	중국 - 심천	HK	HKS	홍콩
strNation	strExchang	종목																					
US	NYS	미국 - 뉴욕																					
US	AMX	미국 - 아멕스																					
S	NAS	미국 - 나스닥																					
CN	SHS	중국 - 상해																					
CN	SZS	중국 - 심천																					
HK	HKS	홍콩																					

함수명	- (NSString*) getItemCodeType:(NSString*)code			
설명	종목의 종목유형을 반환한다.			
리턴값	종목유형			
파라미터	NSString strCode	종목코드 (단축코드)		
내용	종목을 분류하기 위한 정보를 리턴한다. 이 값으로 종목유형에 따라 서비스 호출이나 주문로직 등 별도 처리를 하기 위한 정보로 사용된다.			
	유형별 정의는 다음과 같다.			
	유형	설명	유형	설명
	A	주식	R	채권
	B	코스피	S	코스닥150선물
C	코스닥	T	국내 - 업종	

	D	ELW	U	유로50선물
	E	ETF	V	코스닥150옵션
	F	ETN	W	KRX300선물
	G	KNEX	X	위클리옵션
	H	K-OTC	a	해외주식
	I	신주인수권	c	홍콩주식
	J	지수선물	d	홍콩ETF
	K	미니선물	f	중국 - 상해
	L	지수옵션	g	중국 - 심천
	M	미니옵션	i	미국 - 뉴욕
	N	야간선물	j	미국 - 나스닥
	O	변동성	l	미국 - 아멕스
	P	섹터	m	미국 - 업종
	Q	주식선물		

함수명	- (NSString*) getItemFullCode:(NSString*)code	
설명	주어진 종목에 대한 표준코드를 리턴한다.	
리턴값	종목의 표준코드(full code)	
파라미터	NSString strCode	종목의 단축코드
내용	<p>단축종목코드에 대한 표준코드를 구한다.</p> <p>시세조회, 주문, 실시간 시세 요청 시에 표준코드를 입력으로 해야함. 이 함수를 이용해서 단축코드에서 표준코드 구해서 입력으로 사용함.</p>	

함수명	- (NSString*) getItemMarketTypeText:(NSString*)code	
설명	주어진 종목코드(단축코드)에 대한 시장유형 이름을 구한다.	
리턴값	시장유형명	
파라미터	NSString strCode	종목코드(단축코드)
내용	종목에 대한 시장유형 이름을 구한다. 정의는 다음과 같다.	
	시장유형명	유형 설명
	KOSPI	주식 - 코스피
	KOSDAQ	주식 - 코스닥
	ELW	ELW
	ETN	ETN
	ETF	ETF
	SRGT	신주인수권

	KONEX	KONX
	KOTC	K-OTC
	FUTURE	지수선물
	OPTION	지수옵션
	MINIUTURE	미니선물
	MINIOPTION	미니옵션
	WEEKOPTION	위클리옵션
	KRX300F	KRX300선물
	KDQ150F	코스닥150선물
	KDQ150_OPT	코스닥150옵션
	JFUTURE	주식선물
	VUTURE	변동성
	TFUTURE	섹터 선물
	CMEFUTURE	야간선물
	BOND	채권
	UPONG	업종
	WSTK_NYS	해외주식 - 미국 - 뉴욕
	WSTK_NAS	해외주식 - 미국 - 나스닥
	WSTK_AMX	해외주식 - 미국 - 아멕스
	WSTK_SS	해외주식 - 중국 - 상해
	WSTK_SZS	해외주식 - 중국 - 심천
	WSTK_HKS	해외주식 - 홍콩 - 주식
	WSK_HKETF	해외주식 - 홍콩 - ETF
	WUPJONG	해외업종

함수명	- (NSString*) getItemCodeInfo:(NSString*)code type:(NSString*)strType		
설명	주어진 종목의 정보를 반환한다.		
리턴값	종목정보		
파라미터	NSString strCode	종목코드(단축코드)	
	NSString strType	원하는 정보유형	
내용	종목의 세부 정보를 구할 때 사용한다.		
	종목 유형에 따라 strType이 달라질 수 있다.		
	종목분류	strType	설명
	공통	shcode	단축코드
		epcode	표준코드(폴코드)
		name	종목명(한글)
type		종목유형	

				getItemCodeType함수 참조
		marketttext		종목유형명 getItemMarketTypeText함수 참조
	선물옵션	callput		콜/풋 구분
		spread		스프레드 종목 여부 "1" - tue
		fullname		긴종목명
		atm		ATM구분
		basemarket		기초자산 코드
		optprice		옵션 행사가
		enddate		종료일
		multiplier		승수
	해외주식	ename		종목명(영문)
		symbolid		종목코드(단축코드)
		isincoe		
		upcodes		업종코드
		divcode		
		tradeunit		거래단위
		floatpoint		소수점 자리
		ticktype		틱 유형
		currency		통화
		bym		
		exchgcode		거래소 코드
		exchgname		거래소 이름
		exchgcode_eugene		유진 자체 거래소 코드
		nation		국가코드

함수명	- (NSData*) encryptData:(NSData*)szData	
설명	평문 데이터를 암호화	
리턴값	암호화된 문자열	
파라미터	NSData szData	평문 데이터
내용	API내부에서 사용하는 암호화 방식으로 암호문 생성 API를 사용하는 앱에서 데이터 암호화시 사용할 수 있음. 반드시 사용해야 하는 함수는 아니며 편의를 위해 제공함.	

함수명	- (NSData*) decryptData:(NSData*)szData		
설명	암호화된 NSData로 해독 NSData 생성		
리턴값	평문 데이터		
파라미터	NSData szData	암호화된 문자열	

내용	
----	--

4.1.2 SmartChampTran

조회(주문)용 서비스를 호출하기 위한 클래스, 호출하고자 하는 TR에 대해 1개체씩 정의하여 사용한다.

분류	반환	함수명	설명
초기화	void	initWithTrCode	인스턴스 초기화
	void	setQueryFile	qry파일 로드
	void	setTranId	TR id 지정
	NSString	getTranId	TR id 반환
데이터	void	setDataHeader	패킷 헤더에 포함시킬 데이터 지정
	void	setScreenNo	화면 번호 지정
	void	clearInputData	입력 데이터 초기화
	void	clearOutputData	출력 데이터 초기화
	void	setDataValue	TR 입력/출력 필드에 데이터 지정
	NSString	getDataValue	TR 입력/출력 필드에 존재하는 데이터 리턴
	int	getDataAttr	TR 출력 필드의 데이터 속성값 리턴 (시세 색상표시용)
요청	int	requestData	데이터 요청
	int	requestNextData	넥스트 데이터 요청
	BOOL	isMoreNextData	조회 후 연속조회를 위한 데이터가 존재하는지 여부
카운트	int	getDataCount	수신된 데이터 블록의 데이터 개수
메시지	String	getDataMsgText	응답 데이터 메시지
	NSString	getDataMsgCode	응답 데이터 메시지 코드

각각의 함수 설명은 아래와 같다.

- 초기화

함수명	- (id)initWithTrCode:(NSString*)trCode delegate:(id <ITranDataDelegate>)delegate	
설명	TRAN 서비스 호출을 위해 인스턴스 초기화	
리턴값	없음	
파라미터	ITranDataDelegate Delegate	서비스 조회에 대한 결과 델리게이트
	NSString trCode	서비스명 (서비스가 정의된 파일명)
내용	TRAN 서비스 호출을 위한 객체 초기화 trCode 가 입력된 함수는 초기화와 쿼리파일 로드를 한번에 수행한다. init를 호출하여 초기화한 경우, setQueryFile(strTrCode)로 쿼리파일 로드를 수행한 후 서비스를 호출해야 한다.	

```
// 주문 TR 초기화
m_tranStockBuy = [[SmartChampTran alloc] initWithTrCode:@"OTD1101U" delegate:self];
[m_tranStockBuy setTranID:@"주식매수주문"];

@protocol ITranDataDelegate <NSObject>

// 서비스 요청 전에 호출
- (void)onTranBeforeRequest:(NSString*)sTranID nextQuery:(BOOL)isNextQuery;
// 서비스 요청 결과 수신
- (void)onTranDataReceived:(NSString*)tran msgCode:(NSString*)code msgText:(NSString*)text;
// 서비스 오류 수신
- (void)onTranError:(NSString*)tran msgCode:(NSString*)code msgText:(NSString*)text;
// 서비스 타임아웃 (서버 응답 없음)
- (void)onTranTimeout:(NSString*)tran;
```

ITranDataDelegate에 sTranID는 별도로 지정하지 않으면 서비스명과 동일하게 들어온다. setTranID로 별도의 아이디를 지정하면 지정된 아이디로 델리게이트의 이벤트 함수가 호출된다. 예를 들어 stmst2 서비스에 대해 여러 instance를 생성한 경우(ex: m_tranCurrPrice1, m_tranCurrPrice2), 각각의 인스턴스를 구별하기 위해 setTranID로 구별되는 값을 지정할 수 있다(ex : 주식현재가1, 주식현재가2). ITranDataDelegate를 여러 인스턴스에 동시에 사용해도 sTranID에 setTranID로 지정한 아이디로 들어와서 어느 인스턴스에 대한 이벤트인지 구분할 수 있다.

함수명	void setQueryFile(NSString strTrCode)	
설명	TRAN 서비스 인스턴스에서 주어진 서비스의 정보를 파일에서 읽는다.	
리턴값	없음	
파라미터	NSString strTrCode	서비스명 (서비스가 정의된 파일명)
내용	초기화된 인스턴스에서 사용할 서비스의 정보를 로드한다. - (id)initWithTrCode:(NSString*)trCode delegate:(id<ITranDataDelegate>)delegate 를 통해서 초기화 했다면 이 함수를 호출하지 않는다. init를 통해서 초기화한 경우에만 호출할 수 있다.	

함수명	-(void)setTranID:(NSString*)strTranID	
설명	TRAN 서비스명 외에 별도의 TRAN서비스아이디 지정	
리턴값	없음	
파라미터	NSString *strTranID	지정할 TRAN서비스 아이디
내용	TRAN 서비스 인스턴스의 TranID는 별도로 지정하지 않으면 setQueryFile로 지정한 서비스 이름이 기본값으로 지정된다. TRAN서비스를 구분해야 할 필요가 있거나, 별도의 아이디를 지정하고자 할 경우, 이 함수로 서비스 아이디를 지정한다.	

함수명	-(NSString*) getTranID	
설명	서비스 인스턴스의 서비스 아이디를 구한다.	
리턴값	TRAN 아이디	
파라미터	없음	
내용	TRAN서비스 인스턴스의 아이디가 필요한 경우 호출한다.	

함수명	-(void)setDataHeader:(NSString*)strKey value(NSString*) strValue							
설명	TRAN서비스의 호출 시 요청데이터 헤더 부분에 지정되는 값을 지정할 수 있다.							
리턴값	없음							
파라미터	NSString strKey	지정할 데이터 이름(키)						
	NSString strValue	지정할 데이터						
내용	서비스 호출 전에 특정 예외처리를 위한 내용							
	<table><tr><th>시장유형명</th><th>유형 설명</th></tr><tr><td>DETAIL_CONT</td><td>조회 시 다음 데이터 조회 가능하게 설정 "1"</td></tr><tr><td>DETAIL_QRY_CNT</td><td>조회 시 데이터 건 수 지정 "20" 한번에 20건씩만 조회</td></tr></table>		시장유형명	유형 설명	DETAIL_CONT	조회 시 다음 데이터 조회 가능하게 설정 "1"	DETAIL_QRY_CNT	조회 시 데이터 건 수 지정 "20" 한번에 20건씩만 조회
	시장유형명	유형 설명						
DETAIL_CONT	조회 시 다음 데이터 조회 가능하게 설정 "1"							
DETAIL_QRY_CNT	조회 시 데이터 건 수 지정 "20" 한번에 20건씩만 조회							
체결내역, 시세추이와 같은 출력이 array로 동작하는 서비스에 대해 한번에 몇건씩 페이지 단위로 조회할 경우 위의 예와 같이 연속조회와 조회건수를 지정하여 사용한다.								

- 데이터

함수명	-(void)clearInputData	
설명	서비스 입력에 지정된 값을 클리어한다.	
리턴값	없음	
파라미터	없음	
내용	TRAN 서비스의 입력데이터를 클리어한다.	

함수명	-(void)clearOutputData	
설명	서비스 출력에 있는 데이터를 클리어한다.	
리턴값	없음	
파라미터	없음	
내용	TRAN 서비스의 출력데이터를 클리어한다.	

함수명	- (void)setDataValue:(NSString*)block fieldID:(NSString *)field index:(int)nIndex value:(NSString *)value	
설명	TRAN 서비스 블록의 데이터 필드에 값을 지정한다.	
리턴값	없음	
파라미터	NSString sBlockName	데이터 블록 이름
	NSString sFieldName	데이터 필드 이름
	int nRecIndex	데이터 인덱스
	Strng sData	지정할 데이터

내용	TRAN 서비스의 InBlock, OutBlock의 필드에 값을 지정한다. 블록이 single인 경우 nRecIndex는 항상 0이고, array형태인 경우 nRecIndex로 인덱스를 지정하여 데이터를 지정할 수 있다.
----	--

함수명	- (NSString*)getDataValue:(NSString*)block fieldID:(NSString*)field index:(int)nIndex;	
설명	TRAN 서비스 블록의 데이터 필드에 있는 값을 얻는다.	
리턴값	필드의 데이터	
파라미터	NSString sBlockName	데이터 블록 이름
	NSString sFieldName	데이터 필드 이름
	int nRecIndex	데이터 인덱스
내용	TRAN 서비스의 InBlock, OutBlock의 필드의 값을 얻는다. 블록이 single인 경우 nRecIndex는 항상 0이고, array형태인 경우 nRecIndex는 인덱스를 지정하여 원하는 row의 데이터를 얻을 수 있다.	

함수명	- (int)getDataAttr:(NSString*)block fieldID:(NSString*)field index:(int)nIndex	
설명	TRAN 서비스의 블록의 데이터 속성값을 얻는다.	
리턴값	데이터 속성값	
파라미터	NSString sBlockName	데이터 블록 이름
	NSString sFieldName	데이터 필드 이름
	int nRecIndex	데이터 인덱스
내용	<p>데이터의 속성값은 모든 필드에 있는 것은 아니며, 필요한 필드에서만 사용된다. 시세 서비스의 현재가, 등락, 등락률 등 상승/하락에 대해 표현색상이 달라지는 경우, 이 속성값으로 이를 결정해서 표현할 수 있다.</p> <p>TRAN 서비스의 InBlock, OutBlock의 필드에서 속성값을 얻는다. 블록이 single인 경우 nRecIndex는 항상 0이고, array형태인 경우 nRecIndex는 인덱스를 지정하여 원하는 row의 속성값을 얻을 수 있다.</p> <p>속성에 대한 정의 SmartChampDefs의 DATA_ATTR_XXX항목으로 판단할 수 있다.</p>	

● 요청

함수명	- (int)requestData
설명	TRAN 서비스 조회를 서버에 요청한다.
리턴값	요청 결과 (리턴이 < 0면 오류)

파라미터	없음	
내용	TRAN 서비스에 대해 조회를 서버에 요청한다. 요청전에 setDataValue로 InBlock에 데이터를 설정하고 requestData()를 호출하는 순서로 동작한다.	

함수명	-(int) requestNextData	
설명	TRAN 서비스 조회 후 다음 데이터가 존재하는 경우 다음 데이터를 요청한다.	
리턴값	요청 결과 (리턴이 < 0이면 오류)	
파라미터	없음	
내용	TRAN 서비스에 대해 다음 데이터 조회를 서버에 요청한다. 요청전에 다음 데이터가 있는지 isMoreNextData()를 호출하여 true인 경우에만 호출해야 한다. 데이터의 연속조회는 조회된 array형태의 데이터를 listview형태의 뷰에 표시 한 후 스크롤이 마지막 항목에 다다랐을 때, 다음 데이터가 있는 체크하여 있으면 다음데이터를 조회하고 조회된 데이터를 listview의 마지막에 추가하여 표시하는 형태로 동작한다.	

함수명	- (int)getDataCount:(NSString*)block	
설명	TRAN 서비스의 블록에 들어있는 데이터 건수를 얻는다.	
리턴값	블록의 데이터 건수	
파라미터	NSString block	블록명
내용	데이터 블록이 array인 경우, 블록 내에 몇건의 데이터가 있는지 얻는다.	

- 처리 결과 메시지

함수명	-(NSString*)getDataMsgText	
설명	TRAN 서비스 조회 후 조회데이터와 함께 수신한 메시지 정보를 얻는다.	
리턴값	메시지 텍스트	
파라미터	없음	
내용	메시지 정보는 ITranDataDelegate의 onTranDataReceived나 onTranError에서 파라미터로 전달되는 sMsgText값을 통해서 얻을 수 있으며, getDataMsgText를 통해서도 얻을 수 있다.	

함수명	-(NSString*)getDataMsgCode	
설명	TRAN 서비스 조회 후 조회데이터와 함께 수신한 메시지 코드를 얻는다.	
리턴값	메시지 코드	

파라미터	없음	
내용	메시지 코드는 ITranDataDelegate의 onTranDataReceived나 onTranError에서 파라미터로 전달되는 sMsgCode값을 통해서 얻을 수 있으며, getDataMsgCode를 통해서도 얻을 수 있다.	

4.1.3 SmartChampReal

실시간 시세 정보를 받기 위한 클래스, 실시간 서비스 1개에 각 1개의 클래스를 선언하여 사용한다.

분류	반환	함수명	설명
초기화	void	- (id)initWithTrCode:(NSString*)trCode delegate:(nonnull id<IRealDataDelegate>)delegate	인스턴스 초기화
	void	setQueryFile	qry파일 로드
	void	setRealId	실시간 id 지정
	NSString	getRealId	실시간 id 반환
데이터	void	setDataValue	실시간 입력/출력 필드에 데이터 지정
	NSString	getDataValue	실시간 입력/출력 필드에 존재하는 데이터 리턴
	int	getDataAttr	실시간 출력 필드의 데이터 속성값 리턴(시세 색상표시용)
요청/해제	void	requestReal	실시간 시세 등록 요청
	void	releaseReal	실시간 시세 해제 요청
카운트	int	getDataCount	수신된 데이터 블록의 데이터 개수

● 초기화

함수명	-(id)initWithTrCode:(NSString*)trCode delegate:(nonnull id<IRealDataDelegate>)delegate	
설명	REAL 서비스 호출을 위해 인스턴스 초기화	
리턴값	없음	
파라미터	IRealDataDelegate Delegate	실시간 데이터 요청/수신 이벤트 델리게이트
내용	<p>REAL 서비스 호출을 위한 객체 초기화</p> <p>initWithTrCode 를 호출하여 초기화한 후, trCode로 쿼리파일 로드를 수행한 후 실시간 요청 작업을 해야 한다.</p> <p>IRealDataDelegate에 sRealID는 별도로 지정하지 않으면 서비스명과 동일하게 들어온다. setRealId로 별도의 아이디를 지정하면 지정된 아이디로 델리게이트의 이벤트 함수가 호출된다.</p>	

--	--

함수명	-(void)setQueryFile(NSString*) strCode)	
설명	REAL 서비스 인스턴스에서 주어진 서비스의 정보를 파일에서 읽는다.	
리턴값	없음	
파라미터	NSString strCode	서비스명 (서비스가 정의된 파일명)
내용	초기화된 인스턴스에서 사용할 서비스의 정보를 로드한다. init를 통해서 초기화한 경우에만 호출할 수 있다.	

함수명	-(void)setRealId:(NSString*)strRealId	
설명	REAL 서비스명 외에 별도의 REAL서비스아이디 지정	
리턴값	없음	
파라미터	NSString strRealId	지정할 REAL서비스 아이디
내용	REAL 서비스 인스턴스의 RealId는 별도로 지정하지 않으면 setQueryFile로 지정한 서비스 이름이 기본값으로 지정된다. REAL서비스를 구분해야 할 필요가 있거나, 별도의 아이디를 지정하고자 할 경우, 이 함수로 서비스 아이디를 지정한다.	

함수명	-(NSString*)getRealId	
설명	서비스 인스턴스의 서비스 아이디를 구한다.	
리턴값	REAL 아이디	
파라미터	없음	
내용	REAL서비스 인스턴스의 아이디가 필요한 경우 호출한다.	

- 데이터

함수명	- (void)setDataValue:(NSString*)block fieldID:(NSString *)fieldID index:(int)index value:(NSString *)value	
설명	REAL 서비스 블록의 데이터 필드에 값을 지정한다.	
리턴값	없음	
fieldID	fieldID	fieldID
	NSString fieldID	데이터 필드 이름
	int index	데이터 인덱스
	NSString value	지정할 데이터
내용	REAL 서비스의 InBlock, OutBlock의 필드에 값을 지정한다. 블록이 single인 경우 nRecIndex는 항상 0이고, array형태인 경우 nRecIndex로 인덱스를 지정하여 데이터를 지정할 수 있다.	

함수명	-(NSString*)getDataValue:(NSString*)sBlockName fieldID:(NSString*)fieldID index:(int)index	
설명	REAL 서비스 블록의 데이터 필드에 있는 값을 얻는다.	
리턴값	필드의 데이터	
파라미터	NSString sBlockName	데이터 블록 이름
	NSString fieldID	데이터 필드 이름
	int index	데이터 인덱스
내용	REAL 서비스의 InBlock, OutBlock의 필드의 값을 얻는다. 블록이 single인 경우 index는 항상 0이고, array형태인 경우 index 인덱스를 지정하여 원하는 row의 데이터를 얻을 수 있다.	

함수명	- (unsigned char)getDataAttr:(NSString*)sBlockName fieldID:(NSString*)fieldID index:(int)index	
설명	REAL 서비스의 블록의 데이터 속성값을 얻는다.	
리턴값	데이터 속성값	
파라미터	NSString sBlockName	데이터 블록 이름
	NSString fieldID	데이터 필드 이름
	int index	데이터 인덱스
내용	<p>데이터의 속성값은 모든 필드에 있는 것은 아니며, 필요한 필드에서만 사용된다. 시세 서비스의 현재가, 등락, 등락률 등 상승/하락에 대해 표현색상이 달라지는 경우, 이 속성값으로 이를 결정해서 표현할 수 있다.</p> <p>REAL 서비스의 InBlock, OutBlock의 필드에서 속성값을 얻는다. 블록이 single인 경우 nRecIndex는 항상 0이고, array형태인 경우 nRecIndex는 인덱스를 지정하여 원하는 row의 속성값을 얻을 수 있다.</p> <p>속성에 대한 정의 SmartChampDefs의 DATA_ATTR_XXX항목으로 판단할 수 있다.</p>	

● 요청

함수명	-(int) requestReal	
설명	REAL 서비스 등록을 서버에 요청한다.	
리턴값	없음	
파라미터	없음	
내용	<p>REAL 서비스에 대해 서버에 등록을 요청한다.</p> <p>요청전에 setDataValue로 InBlock에 데이터를 설정하고 requestReal()를 호출하는 순서로 동작한다.</p>	

--	--

함수명	-(int) releaseReal
설명	REAL 서비스 등록을 해제하기 위해 서버에 요청한다.
리턴값	없음
파라미터	없음
내용	<p>REAL 서비스에 대해 서버에 등록했던 요청을 해제한다.</p> <p>요청시세 사용 했던 setDataValue로 지정된 값이 변경되지 않았다면 releaseReal만 해제해도 된다.</p> <p>지정되어 있지 않다면 다음과 같이 등록할 때 같이 처리한다.</p>

4.1.4 SmartChampDefs

SmartChamp Open API에 사용되는 상수 정의

- 로그인 유형

SmartChampAPI의 getLoginType()으로 얻는 값으로 현재 로그인 상태에 대한 값이다.

타입	정의	값	설명
int	LOGIN_TYPE_NONE	0	로그인 안됨
int	LOGIN_TYPE_AUTO	1	자동로그인 이전에 로그인한 경우, 다음 앱 실행시 저장된 아이디로 자동로그인 상태로 유지됨
int	LOGIN_TYPE_SISE	2	시세 조회용 로그인 공동인증서 없이 아이디/비번만으로 로그인한 경우
int	LOGIN_TYPE_CERT	3	거래용 로그인 공동인증서를 이용하여 로그인한 경우, 주문을 위해서는 거래용 로그인이 필수

- 화면 재조회 요청 사유

INetInitDelegate의 onRequiredRefresh로 화면 재조회 요청 시, 재조회가 일어난 이유를 함께 받는데 그 이유를 정의.

타입	정의	값	설명
int	REFRESH_REASON_NORMAL	0	앱이 비활성화되는 경우, 실시간 데이터를 수신

			하지 않기 위해 서버에 등록 해제를 요청함. 앱이 다시 활성화되면 데이터를 최신으로 유지하고 실시간 데이터를 다시 요청할 필요가 있음. 이 사유로 onRequiredRefresh가 호출되면 데이터 요청 및 실시간 등록을 수행해야 함
int	REFRESH_REASON_RECONNECT	1	앱 통신 단절 시 또는 장시간 비활성화 상태에 의한 단절 시 재접속이 발생함. 재접속이 완료 시 화면의 데이터 갱신 작업이 필요함. 처리 방식은 REFRESH_REASON_NORMAL경우와 동일함
int	REFRESH_REASON_CHANGED_OSSISE	2	해외주식 실시간 서비스는 1개의 단말에서만 사용 가능함. 앱 사용중 실시간 사용 상태가 변경된 경우 호출됨. 현재 화면이 해외주식 실시간 화면인 경우 데이터 재조회 수행이 필요함

● 서버 접속 모드

SmartChampAPI의 setDevMode로 서버 접속 모드를 지정할 수 있다. 서버 접속 모드 지정 시 입력하는 값을 정의.

타입	정의	값	설명
NSString	SERVER_MODE_REAL	"0"	운영서버, 앱 시작 시 운영서버로 접속할 것인지, 개발/테스트 서버로 접속할 것인지 지정할 경우 사용되는 값
NSString	SERVER_MODE_DEV	"1"	개발/테스트용 서버

4.2 인터페이스

4.2.1 INetInitDelegate

앱 실행 후 초기 접속, 버전처리, 종목정보 로드 등 초기 실행 관련한 상태를 리턴하는 델리게이트입니다.

반환	함수명	설명
void	onSessionConnecting	초기 접속 시작
void	onSessionConnected	접속 결과 리턴 (성공여부, 실패 메시지)
void	onAPIVersionState	"라이브러리 버전 체크 결과
void	onUpdateDownloadState	"버전처리 진행 상황
void	onMasterLoadState	마스터 로드 상태 (0 : 시작, 1 : 완료)
void	onInitFinished	초기화 완료

함수명	- (void)onSessionConnecting	
설명	SmartChampAPI의 startConnect() 호출시 접속이 시작됨을 알린다.	
리턴값	없음	
파라미터	없음	
내용	<p>서버 접속이 시작되었음을 알리는 이벤트</p> <pre> 9 0 - (void)onSessionConnecting 1 { 2 3 } 4 </pre>	

함수명	- (void)onSessionConnected:(BOOL)isSuccess error:(NSString*)strErrorMsg	
설명	서버에 접속이 되거나, 접속이 실패한 경우 호출된다.	
리턴값	없음	
파라미터	bool isSuccess	성공여부
	NSString strErrMsg	실패시 실패 내용
내용	<p>접속 작업이 완료된 후 접속에 대한 결과를 알리기 위해 호출된다. isSuccess가 true이면 자동으로 다음작업이 수행되고, false면 startConnect로 시작된 접속 처리가 중지된다.</p> <pre> 5 /** 6 * 서버 접속 완료 7 * @param isSuccess 성공여부 8 * @param strErrorMsg 에러메세지 9 */ 0 - (void)onSessionConnected:(BOOL)isSuccess error:(NSString*)strErrorMsg </pre>	

함수명	(void)onAPIVersionState:(NSInteger)nCheckState update:(NSString*)sUpdateInfo;	
설명	API 버전체크에 대한 결과 알림.	
리턴값	없음	
파라미터	int nCheckState	버전체크 결과

		< 0 : 버전체크 실패 (진행 중지됨) 0 : 정상 (다음 작업 자동 진행) 1 : 업그레이드 필요(다음 작업 자동 진행) 2 : 업그레이드 필수 상황 (진행 중지됨) 사용자에게 안내하고 앱 업데이트 유도
	NSString strMessage	체크 결과 메시지
내용	API 버전에 대한 비교 작업이 이루어 지고, 허용 범위 내의 버전은 통과되고, 오류가 발생했거나 반드시 업데이트가 필요한 상황이면, 접속 프로세스가 종료되고 사용자에게 알려야 한다.	

함수명	- (void)onUpdateDownloadState:(NSInteger)nState index:(NSInteger)nIndex count:(NSInteger)nCount message:(NSString*)sMessage	
설명	OpenAPI에서 필요한 파일(종목정보, 쿼리정보)을 다운로드 하는 경우에 진행 상황을 알리기 위해 호출됨	
리턴값	없음	
파라미터	int nState	업데이트 진행 상태 0 – 업데이트 작업 중(체크,다운로드) 1 – 진행상황 알림 2 – 완료됨
	int nIndex	다운로드 중인 파일의 순번
	int nCount	총 다운로드 파일 수
	NSString sMessage	1 – 진행상황 알림의 경우 내용
내용	파일 다운로드에 대한 알림 제공	

함수명	(void)onMasterLoadState:(NSInteger)nState	
설명	OpenAPI에서 사용할 종목정보 로드 작업 완료를 알림	
리턴값	없음	
파라미터	int nState	의미없음
내용	종목정보 파일을 로드하는 작업을 수행하고 작업이 완료되면 알리는 이벤트	

함수명	-(void) onInitFinished	
설명	초기 접속 작업이 완료되었음을 알림	
리턴값	없음	
파라미터	없음	
내용	앱 시작 시 초기 접속 작업 요청하고 접속작업이 완료되고, OpenAPI를 사용하여 조회/실시간 요청이 가능한 상태가 된 것을 알리는 이벤트.	

--	--

4.2.2 INetLoginDelegate

로그인 작업에 대한 시작과 결과 정보를 리턴하기 위한 델리게이트입니다.

반환	함수명	설명
void	onLoginStarted	로그인 작업 시작 알림
void	onLoginResult	로그인 결과 리턴 (성공여부, 실패 메시지)

함수명	-(void) onLoginStarted	
설명	로그인 요청 시 로그인 작업 시작을 알림	
리턴값	없음	
파라미터	없음	
내용	loginUser 함수 호출 시 로그인 작업 시작된 경우 알림	

함수명	- (void)onLoginResult:(BOOL)isSuccess error:(NSString*)strErrMsg	
설명	로그인 작업 완료 후 결과 알림	
리턴값	없음	
파라미터	BOOL isSuccess	로그인 성공 여부
	NSString strErrMsg	로그인 오류 메시지
내용	주어진 login정보로 로그인 작업 수행하고 그 결과를 리턴	

4.2.3 INetLogoutDelegate

로그아웃 요청시 로그아웃 작업 완료를 알리기 위한 델리게이트.

반환	함수명	설명
void	onLogoutFinished	로그아웃 완료

함수명	-(void)onLogoutFinished	
설명	로그아웃 완료 알림	
리턴값	없음	
파라미터	없음	
내용	logoutUser() 호출 시 로그아웃 작업이 완료되면 호출된다.	

--	--

4.2.4 IAccountPwdCheckDelegate

계좌 비밀번호 체크에 대한 결과 리턴을 위한 델리게이트.

반환	함수명	설명
void	onAccountPwdCheckResult	계좌 비밀번호 체크 결과 리턴

함수명	-(void)onAccountPwdCheckResult:(BOOL)isSuccess accno:(NSString*)strAccNo message:(NSString*)strMessage	
설명	계좌번호와 계좌비번으로 검증 요청 한 후 받는 이벤트	
리턴값	없음	
파라미터	BOOL isSuccess	비번 체크 성공 여부
	NSString strAccNo	검증요청한 계좌번호
	NSString strMessage	오류 메시지
내용	계좌번호와 키패드에서 입력받은 계좌비번으로 checkAccPwd함수 호출 시 결과를 리턴 받는다.	

4.2.5 ITranDataDelegate

조회 서비스 호출 시 호출되기전 시점과 결과 수신 후 데이터 처리를 위한 델리게이트.

반환	함수명	설명
void	onTranBeforeRequest	데이터 요청 전 호출되는 이벤트, 호출전 필요한 작업을 여기서 수행
void	onTranDataReceived	데이터 수신 호출되는 이벤트, 결과 데이터를 처리 하는 작업을 여기서 수행
void	onTranError	호출한 서비스의 오류 발생 시 호출되는 이벤트
void	onTranTimeout	호출한 서비스가 정해진 시간안에 응답이 오지 않는 경우 발생하는 이벤트

함수명	- (void)onTranBeforeRequest:(NSString*)sTranID nextQuery:(BOOL)isNextQuery	
설명	TRAN 서비스 호출 시 서버로 요청 데이터 전송되기 전에 호출됨	
리턴값	없음	
파라미터	NSString sTranID	TRAN 서비스 아이디
	BOOL isNextQuery	연속조회 여부
내용	TRAN 서비스를 호출하면 인블럭(입력데이터)에 설정된 값으로 요청정보를 생성하게 되	

	<p>는데, 데이터 생성 전에 호출된다.</p> <p>requestData 호출 전에 setDataValue로 입력 데이터 설정하거나, 이 이벤트 함수에서 설정하거나 상관없이 동작한다.</p>
--	---

함수명	- (void)onTranError:(NSString*)tran msgCode:(NSString*)code msgText:(NSString*)text	
설명	TRAN 서비스 요청 후 오류가 발생한 경우 호출됨	
리턴값	없음	
파라미터	NSString sTranID	TRAN 서비스 아이디
	NSString sErrCode	오류 메시지 코드
	NSString sErrMsgText	오류 메시지 내용
내용	TRAN 서비스 데이터를 서버에 요청한 후 오류가 발생한 경우 호출됨	

함수명	-(void) onTranTimeout:(NSString*) sTranID	
설명	TRAN 서비스 요청 후 결과 데이터가 정해진 시간안에 도착하지 않는 경우	
리턴값	없음	
파라미터	NSString sTranID	TRAN 서비스 아이디
내용	서비스 조회 요청에 응답이 없는 경우 호출됨	

4.2.6 IRealDataDelegate

실시간 서비스 등록 시 등록 요청 전 시점과 실시간 데이터 수신 시 데이터 처리를 위한 델리게이트.

반환	함수명	설명
void	onRealBeforeRequest	실시간 등록 호출 시, 등록 요청 서버에 전송하기 전에 호출됨
void	onRealDataReceived	서버에서 실시간 데이터 도착 시 호출되는 이벤트

함수명	-(void)onRealBeforeRequest:(NSString*) sRealId	
설명	실시간 서비스 등록 요청 시 서버에 요청 올리기 전에 호출됨	
리턴값	없음	
파라미터	NSString sRealId	실시간 서비스 아이디
내용	실시간 서비스 등록 요청 시 서버에 요청이 올라가기 전에 데이터를 구성하는데 그 전에 호출됨. 여기서 원하는 종목을 설정하거나, requestReal호출하기 전에 설정하거나 상관없이 동작한다.	

--	--

함수명	-(void) onRealDataReceived:(NSString*) sRealId	
설명	실시간 데이터 수신 시 호출됨	
리턴값	없음	
파라미터	NSString sRealId	실시간 서비스 아이디
내용	<p>실시간 데이터 수신 시 호출됨</p> <p>실시간 데이터는 array 형태의 데이터이며, 종목체결 같은 경우 한번에 1개 이상의 데이터가 들어올 수 있다. getDataCount로 OutBlock에 들어있는 데이터 개수를 얻은 후 필요한 경우 개수 만큼 처리가 필요하다.</p>	

4.2.7 인증서 관련 함수

공동인증서 가져오기 수행 시 가져오기를 위한 랜덤 번호 생성 정보와 가져오기 결과를 리턴 하기 위한 델리게이트.

반환	함수명	설명
NSInteger	getUserCertListCount	공동인증서 저장 갯수
NSArray	getUserCertList	공동인증서 리스트 배열
NSString	requestCertImportNumber	인증서 가져오기 랜덤 번호 생성

함수명	requestCertImportNumber	
설명	인증서 가져오기를 위해 생성된 랜덤번호를 받는다.	
리턴값	NSString	
	Failed`[errorcode]message	실패`[에러코드]에러메세지
	Success`random number	성공`생성된 랜덤번호
내용	인증서 가져오기를 위해 requestCertImportNumber를 호출하면 인증서 관리 모듈에서 랜덤 번호를 생성하여 돌려준다.	

	<pre> 23 24 NSString *sSimple = [[SmartChampAPI instance] requestCertImportNumber]; 25 NSArray *arraySimple = [sSimple componentsSeparatedByString:@"`"]; 26 NSString *sNumber1 = nil; 27 NSString *sNumber2 = nil; 28 NSString *sNumber3 = nil; 29 NSArray *arrayNumber = nil; 30 if ([[arraySimple objectAtIndex:0] isEqualToString:@"Success"]){ 31 NSString *simpleNumber = [arraySimple objectAtIndex:1]; 32 33 sNumber1 = [simpleNumber substringToIndex:4]; 34 sNumber2 = [simpleNumber substringWithRange:NSMakeRange(4, 4)]; 35 sNumber3 = [simpleNumber substringWithRange:NSMakeRange(8, 4)]; 36 37 arrayNumber = [NSArray arrayWithObjects:sNumber1, sNumber2, sNumber3]; 38 }else{ </pre>
--	--

함수명	(NSString*)importCert	
설명	인증서 가져오기	
리턴값	NSString	
	Failed`인증서 복사가 실패하였습니다.	실패일 경우 반환 메시지
	"Success`정상적으로 인증서 복사가 완료되었습니다.	성공했을 경우 반환 메시지
내용	인증서 내보내기 하는 단말에서 랜덤 번호 입력 후 내보내기를 수행한 경우, importCert를 호출하여 단말에 저장할 수 있다. 내려받아 저장하기가 완료되면 성공 수신 여부가 반환됨. 수신된 공동인증서는 strCertDN 변수에 DN값이 들어있다.	

4.2.8 인증서 삭제

반환	함수명	설명
NSString	- (NSString*) deleteCert:(NSString*)strCertDN certPW:(NSString*)certPW	공동인증서 삭제

함수명	(NSString*) deleteCert:(NSString*)strCertDN certPW:(NSString*)certPW	
설명	공동인증서 삭제	
리턴값	성공 실패 여부	
파라미터	Failed	실패
	Success	성공
내용	인증서 DN과 인증서 비밀번호를 통하여 인증서를 삭제한다.	