**openMSync**

**User’s Manual**

**Release 1.0**

Inervit Co., Ltd

Copyright (C) Inervit openMSync 2012

Inervit CO., Ltd.

All rights reserved.

No part of this product may be reproduced in any form or by any means without expressed written permission from:

**Inervit CO., Ltd.**

IDAM Bldg. 231-16

NonHyun-Dong, KangNam-Gu, Seoul Korea

Tel: +82-2-2015-0660

Fax: +82-2-2015-0610

Inervit openMSync, Inervit mSync, and Inervit openML are registered trademarks of Inervit CO., Ltd.

All other names are trademarks or registered trademarks of their respective owners.

목차

[0. About This Manual 5](#_Toc341706096)

[1. 개요 6](#_Toc341706097)

[1.1. mSync 소개 6](#_Toc341706098)

[1.2. 응용 분야 7](#_Toc341706099)

[1.3. mSync 구조 및 기능 7](#_Toc341706100)

[2. mSync 설치 및 환경 설정 10](#_Toc341706101)

[2.1. 시스템 환경 10](#_Toc341706102)

[2.2. 설치 및 응용 10](#_Toc341706103)

[2.2.1. 설치 방법 10](#_Toc341706104)

[2.2.2. 폴더 구성 10](#_Toc341706105)

[2.3. 환경설정 11](#_Toc341706106)

[2.4. ODBC 설정 11](#_Toc341706107)

[2.4.1. MS-SQL 서버 설정 12](#_Toc341706108)

[2.5. mSync 서버 실행 15](#_Toc341706109)

[2.6. mSync 서버 종료 16](#_Toc341706110)

[3. mSync 응용 시스템 개발 방법 17](#_Toc341706111)

[3.1. 응용 시스템 개발 과정 17](#_Toc341706112)

[3.1.1. 데모 프로그램 설명 17](#_Toc341706113)

[3.1.2. 테이블 스키마 17](#_Toc341706114)

[3.2. 서버 스크립트 작성 방법 18](#_Toc341706115)

[3.2.1. 데이터베이스 서버 연결 18](#_Toc341706116)

[3.2.2. 서버 스크립트 작성 19](#_Toc341706117)

[3.3. 사용자 등록 23](#_Toc341706118)

[3.4. 클라이언트 프로그램 작성 방법 24](#_Toc341706119)

[4. mSync 도구 및 API 27](#_Toc341706120)

[4.1. mSync admin 도구 27](#_Toc341706121)

[4.1.1. mSync admin 도구 실행 27](#_Toc341706122)

[4.1.2. 기본 메뉴 설명 27](#_Toc341706123)

[4.1.3. 세부 메뉴 설명 33](#_Toc341706124)

[4.1.4. Log Manager 메뉴 53](#_Toc341706125)

[4.2. mSync clientAPI 57](#_Toc341706126)

[4.2.1. 개요 57](#_Toc341706127)

[4.2.2. API 57](#_Toc341706128)

[A. 스크립트 작성 시 DBMS별 유의사항 58](#_Toc341706129)

[B. 제한 사항 60](#_Toc341706130)

1. About This Manual

Inervit openMSync (이하 mSync) 사용자 매뉴얼은 mSync 시스템을 이용하여 DB 동기화 응용 시스템을 개발하는 방법을 설명하는 지침서이다. 이 문서는 mSync 제품의 구조 및 기능에 대해 설명하며 데이터베이스 서버에 접근하는 방법, 서버 스크립트를 작성하는 방법, mSync 클라이언트로 제공되는 API를 이용하여 클라이언트 프로그램을 작성하는 방법 등을 설명한다. 또한, mSync 서버를 실행하고 mSync 응용 시스템을 운용하는 데에 필요한 제반 사항에 대해 설명한다.

본 매뉴얼을 사용하는 개발자는 ODBC에 대한 개념 및 설정 방법을 숙지하고 기본적인 프로그래밍 기술을 습득하고 있으며 SQL 질의어를 생성할 수 있다고 가정한다.

* 제 1장에서는 mSync 시스템의 소개 및 구조와 그 기능에 대해 설명한다. mSync 제품으로 제공되는 mSync 시스템, 사용자의 환경에 맞게 개발한 mSync 응용 시스템의 개념에 대해 설명하며 mSync 제품의 응용 분야를 소개한다.
* 제 2장에서는 mSync 시스템의 설치에 필요한 시스템 환경 및 설치 방법, 폴더 구성, ODBC와 시스템 테이블들을 생성하는 방법에 대해 설명한다.
* 제 3장에서는 mSync 응용 시스템의 개발 과정을 설명하며 서버 스크립트 작성 방법과 클라이언트 프로그램 작성 방법에 대해 설명한다.
* 제 4장에서는 mSync admin 도구 및 mSync 클라이언트로 제공되는 API에 대해 설명한다.
* 기타 사항으로 데이터베이스 서버에 따른 스크립트 작성시 유의 사항과 제한 사항에 대해 언급한다.

# 개요

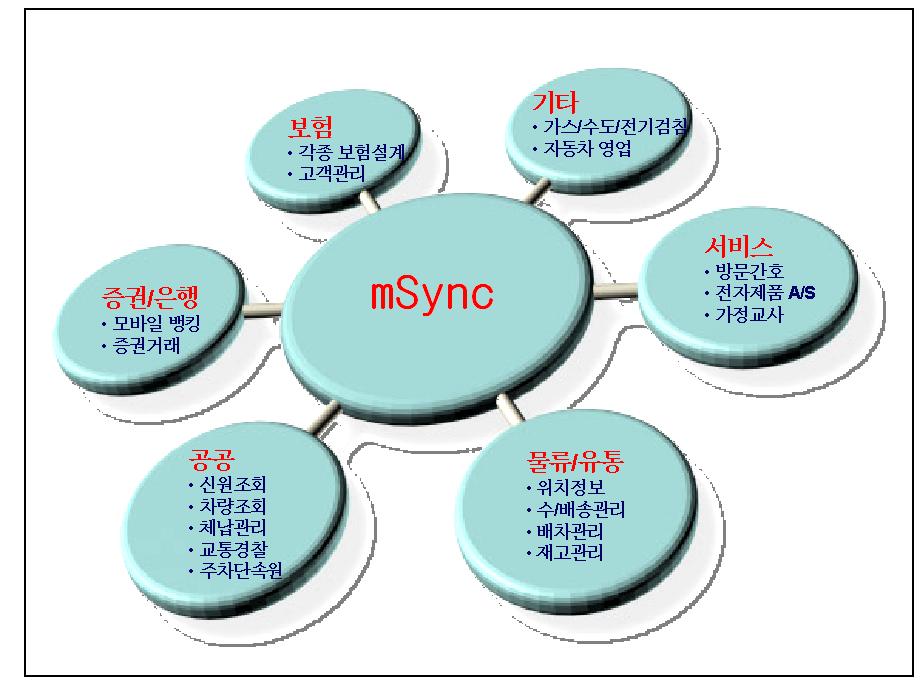
## mSync 소개

현재는 무선 통신의 기술과 모바일 기기들의 성능 확장으로 유선망을 중심으로 하는 기업의 컴퓨팅 환경이 무선망으로 확대되고 있다. 이에 따라 기존에는 기업들의 데이터베이스 서버에서 주로 이루어졌던 데이터 처리들이 모바일 기기에서도 필요하게 되었고 데이터베이스 서버와 모바일 기기 간의 데이터 교환이 필수적으로 요구되고 있다. mSync는 데이터베이스 서버와 모바일 기기의 데이터 교환을 효율적으로 지원하는 미들웨어이다.

mSync는 모바일 비즈니스 환경에서 기존의 데이터베이스 서버에 저장된 기업 데이터나 파일들을 주어진 비즈니스 로직에 따라 모바일 기기에 다운로드하며, 모바일 기기에서 변경되거나, 새로 추가된 데이터는 다시 데이터베이스 서버에 업로드하는 기능을 제공한다. 또한, 모바일 기기의 사용자를 등록하여 인증에 성공한 사용자만이 동기화할 수 있도록 한다.

PDA, 스마트 폰과 같은 모바일 기기에 탑재되는 Mobile DBMS (예, openML)는 PIMS, GIS 등과 관련된 다양한 모바일 응용 프로그램에 적용 가능하며 디스크 없는 환경에서도 DBMS의 기능을 적용할 수 있도록 각 모바일 기기의 환경에 맞게 변경하여 제공된다. 또한, mSync는 모바일 기기들과 cubrid, Oracle, MySQL, MS-SQL, Access 등 이종의 데이터베이스들과 양방향으로 데이터를 동기화할 수 있는 기능을 한다.

## 응용 분야

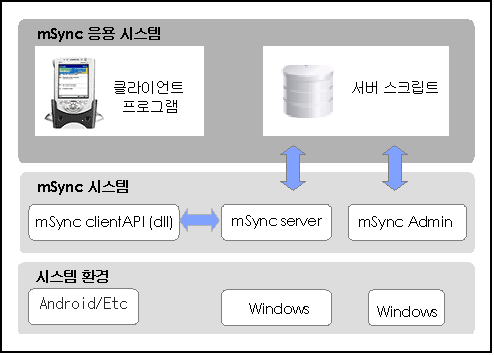


<그림 1-1> mSync의 응용분야

mSync는 각종 보험설계 및 고객을 관리하는 보험 분야, 모바일 뱅킹이나 증권 거래, 신원이나 차량 조회, 체납관리, 주차 단속 등의 공공분야, 위치정보나 수/배송 관리, 배차관리 및 재고 관리 등의 물류/유통 분야, 전자제품 A/S나 방문 간호 등의 서비스 분야와 같이 다양한 모바일 비즈니스 환경에 적용될 수 있다.

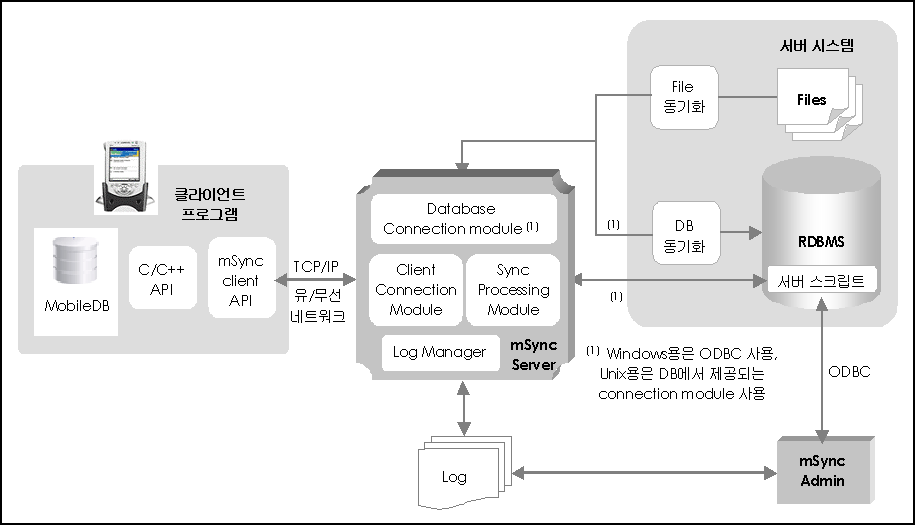
## mSync 구조 및 기능

mSync는 크게 서버와 클라이언트로 구성되며 서버 부분에서는 mSync 서버 프로그램을 제공하고 클라이언트 부분에는 mSync 클라이언트 API set을 제공한다. 또한, mSync 서버의 기본 기능들을 관리할 수 있도록 mSync admin 도구를 제공하여 서버 스크립트를 작성하거나 mSync 서버의 사용자를 등록할 수 있다. 이렇게 mSync에서 제공되는 mSync 서버, mSync 클라이언트 API set, mSync admin 도구를 mSync 시스템이라 하고 비즈니스 로직에 맞춰 서버 스크립트와 클라이언트 프로그램을 구성한 시스템을 mSync 응용 시스템이라 한다. <그림 1-2>은 시스템 관점에서 본 mSync 구조를 표현한다.



<그림 1-2> mSync 구조도

<그림 1-3>는 mSync의 세부 구조도를 표현한 그림으로 크게 네 부분으로 나누어 설명하도록 한다.



<그림 1-3> mSync 세부 구조도

* 클라이언트 프로그램 : mSync 클라이언트 API set을 이용하여 작성된 프로그램을 말하며(본 그림에서는 Mobile DB도 Mobile DB의 C/C++ API 모듈 포함) mSync 클라이언트와 mSync 서버 사이는 TCP/IP를 이용한 유/무선 네트워크로 연결되어 데이터 및 파일의 동기화를 수행한다.
* 서버 시스템 : 기존의 기업 정보를 파일 또는 데이터베이스로 저장 관리하는 서버 즉, 데이터베이스 서버를 말하며 이곳에는 mSync 시스템 테이블들이 저장된다(mSync 시스템 테이블에 대해서는 2.3 장에서 설명한다).
* mSync 서버 : mSync 서버는 데이터베이스 서버에 대한 연결을 담당하는 database connection module, mSync 클라이언트와 연결을 담당하는 client connection module, 동기화의 전반적인 과정을 담당하는 sync processing module, 로그를 관리하는 log manager의 네 부분으로 구성된다. Database connection module은 탑재되는 시스템이 Windows 기반에서 ODBC를 통해 데이터베이스 서버에 연결한다. Sync processing module은 미리 작성된 스크립트 정보에 맞춰 동기화의 전반적인 과정을 담당하게 된다. Client connection module은 mSync 클라이언트 API set인 라이브러리와 TCP/IP를 이용하여 연결되어 데이터를 주고 받는 부분을 수행한다.
* mSync admin 도구 : mSync 서버를 사용할 모바일 기기 사용자 등록 및 서버 스크립트 작성, mSync 서버의 로그 모니터링 기능을 제공한다.

# mSync 설치 및 환경 설정

## 시스템 환경

mSync 시스템을 사용하기 위해서는 다음과 같은 사양이 요구된다.

* mSync 서버 :Windows 2000/NT/XP, Windows Vista, Windows 7
* mSync admin : Windows 기반 (ODBC 필요)
* mSync 클라이언트 : Android, Windows 기반. (WinCE, embedded Visual C++ 라이브러리 제공으로 지원 범위 확대 가능)
* 지원되는 데이터베이스 서버 : MS-SQL, Cubrid, Oracle, Sybase, mySQL, DB2, Access

## 설치 및 응용

### 설치 방법

mSync 서버는 Windows 기반 서버에 설치 가능하며 기본적으로 압축 파일 형태로 제공된다.

따라서 압축을 풀기만 하면 적용 준비가 완료된다.

### 폴더 구성

|  |  |  |  |
| --- | --- | --- | --- |
|  | Server | bin | Admin.exe mSync.exe mSyncDBCTL.dll  (관리 도구 및 mSync 서버) |
| Script | DBMS에 mSync 운영 관련된 System 테이블을 생성하는 script 파일들 |
| Client | Android | libmSyncClientJNI.so 등과 같은 Android용 library |
| Windows | Windows용 library |

<그림 2-3> mSync 폴더 구성 – windows 기반

## 환경설정

mSync 서버는 서버 스크립트를 이용하여 동기화 작업을 수행하며 mSync 응용 시스템에서 필요한 모든 정보를 자체의 시스템 테이블들에 저장한다. 따라서, mSync 시스템을 사용하기 전에 이 테이블들을 생성해줘야 한다. 본 제품을 설치하면 mSyncServer 폴더 안에 ***scripts***라는 폴더가 생성되며, 여기에 각 데이터베이스 서버에 대한 시스템 테이블을 만들 수 있는 스크립트 파일이 제공되므로(Sybase인 경우 mSync\_Sybase.sql 이고 MSSQL인 경우 mSync\_MSSQL.sql, 언급되지 않은 일반 DB의 경우에는 mSync.sql) mSync admin 도구의 실행에 앞서 시스템 테이블을 넣을 데이터베이스를 생성하고 적절한 스크립트 파일을 이용하여 아래와 같은 시스템 테이블들을 생성해준다.

* openMSYNC\_version
* openMSYNC\_DSN
* openMSYNC\_table
* openMSYNC\_script
* openMSYNC\_user

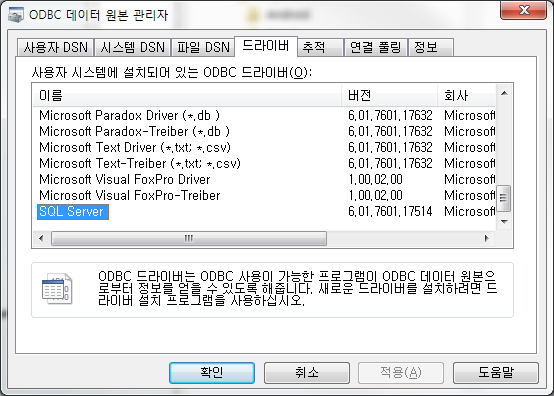
## ODBC 설정

ODBC(Open Database Connectivity)는 데이터베이스를 액세스하기 위한 표준 개방형 응용 프로그램 인터페이스로서 mSync 서버(Windows 기반)나 mSync admin 도구에서 데이터베이스 서버를 액세스할 때 필요하다. 이를 위해서는 액세스할 각 데이터베이스마다 별도의 모듈이나 드라이버가 필요하며 여기서는 Oracle, Access, MS-SQL 등 DBMS 서버에 대해 설정하는 방법에 대해 설명한다. 제어판에서 ODBC를 설정할 수 있다. 이렇게 ODBC를 설정하는 방법에 대해서는 각각의 DB마다 약간씩 차이가 있으나 큰 어려움 없이 설정할 수 있기 때문에 여기서 개별적으로 설명하지는 않는다.

본 설명서에서는 이후 다루게 될 예제를 위한 설정을 기준으로 MS-SQL Server에 대해서만 설명을 하도록 한다.

제어판에서 “***데이터 원본(ODBC)***”을 선택해서 “**ODBC 데이터 원본 관리자**”를 열면 <그림 2-4>와 같은 화면을 볼 수 있다. 여기서 필요한 DBMS용 드라이버가 있는지 확인할 수 있다.

가능하다면 “***시스템 DSN***” 탭에서 “***추가***” 버튼을 눌러 각 데이터베이스 서버에 맞는 ODBC를 설정하도록 한다.

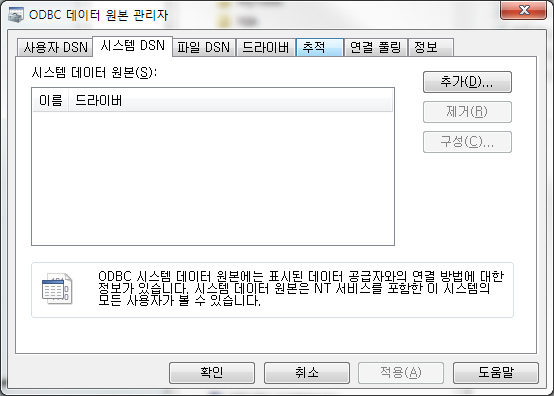


<그림 2-4 > ODBC 데이터 원본 관리자 화면

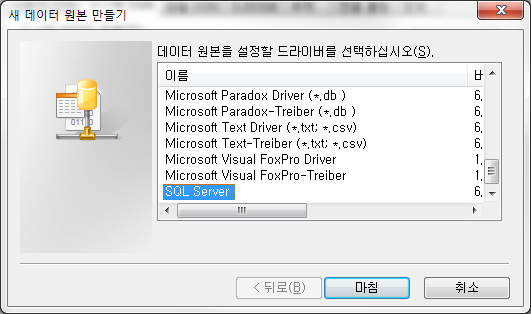
### MS-SQL 서버 설정

여기서는 SQL Server를 설치한 후 설정하는 방법에 대해 설명한다.

1. <그림 2-5-a>에서 “***추가***”를 클릭하여 SQL Server를 선택하고 “***마침***”을 클릭한다.

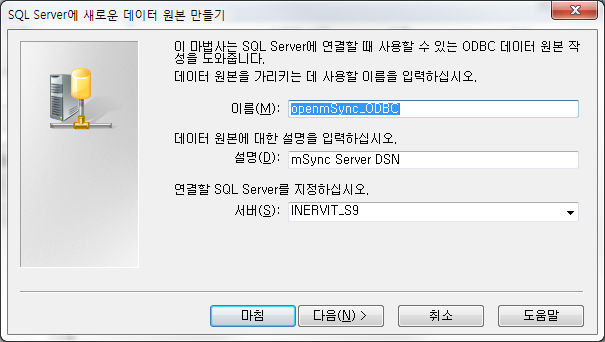


<그림 2-5-a > 새 데이터 원본 만들기 화면1



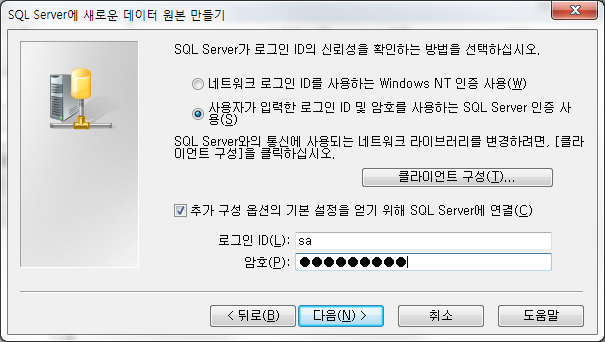
<그림 2-5-b> 새 데이터 원본 만들기 화면2

1. 첫번째 입력 상자에는 DSN(Data Source Name)을 입력하고, 두 번째는 간단하게 요약설명을 적으며 세 번째는 서버의 위치를 입력한다. 만일 SQL Server가 네트워크상에 있다면 그 주소를 찾아 적거나 선택하면 되고 “***다음 >***” 버튼을 클릭한다.



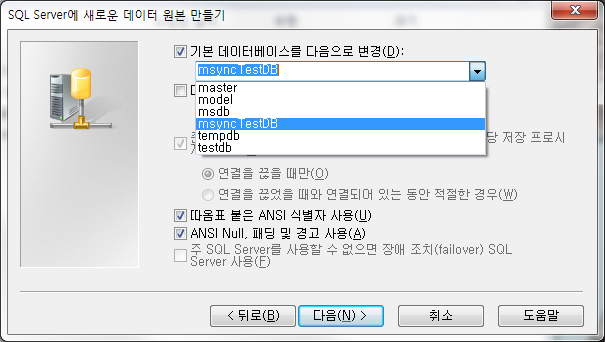
<그림 2-5-c> DSN 정보 입력 화면

1. 다음으로 SQL Server에 연결하는 방법을 지정하는데 SQL Server 접근 시에 어떤 방법을 사용하고 ID와 암호는 어떤 것을 사용할 것인지를 설정한다.



<그림 2-5-d> 로그인 ID와 암호를 지정하는 화면

1. 모든 설정이 끝나면 “***다음 >***” 버튼을 클릭하여 기본 데이터베이스를 설정하고 다시 “***다음 >***” 버튼을 클릭한 뒤 “***마침***”을 클릭하면 ODBC 설정이 끝나게 되며, “***마침***”을 누른 뒤에 데이터 원본 테스트를 할 수도 있다.



1. 설정 이후에도 <그림 2-5-a > 화면에서 “***구성***”을 눌러서 설정 수정을 할 수 있다.

## mSync 서버 실행

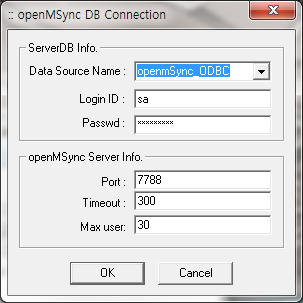
여기서는 mSync 서버의 실행에 대해 설명하도록 한다.

mSync의 실행은 설치 후 또는 mSync 응용 시스템의 개발을 마친 후에 할 수도 있다.

mSync 서버를 수행하기 전에 ODBC 드라이버를 설정해야 한다. 또한 Admin 도구를 이용해서 필요한 설정을 해 두어야 실제 클라이언트와의 통신을 통해 동기화 작업을 지원할 수 있다. (Admin 도구에 대해서는 별도로 다루기로 한다.)

mSync.exe 프로그램을 실행시키면 <그림 2-6>와 같은 화면이 보여진다. 2.4 장에서 생성한 시스템 테이블의 데이터베이스에 대한 DSN을 선택한 뒤 “***OK***” 버튼을 클릭하면 지정된 port에 서버가 구동되며 mSync 서버의 아이콘이 트레이에 추가된다.

* Data Source Name : 현 시스템에 설정되어 있는 ODBC의 Data Source Name을 보여준다. 시스템 테이블들이 있는 데이터베이스의 DSN을 선택( <그림2-5-c>에서 설정한 DSN 선택)
* Login ID : 데이터베이스 서버 접속을 위한 DB 사용자 ID를 입력
* Passwd : 사용자의 암호를 입력
* Port : mSync 서버를 띄울 Port 번호를 입력
* TimeOut: mSync서버와 클라이언트간의 소켓 timeout 시간 설정
* Max users: mSync Server에 동시 접속 가능한 클라이언트 개수 설정



<그림 2-6> 데이터베이스 서버 연결 화면

## mSync 서버 종료

트레이에 있는 mSync 서버의 아이콘에 오른쪽 마우스 버튼을 클릭하면 <그림 2-7>과 같은 메뉴가 보이고 여기서 “Exit”를 선택하면 종료된다.



<그림 2-7> mSync 서버 종료화면

# mSync 응용 시스템 개발 방법

2장에서 mSync 시스템을 설치하고 mSync 서버를 실행/종료하는 방법에 대해 설명하였다. 이제 비즈니스 로직에 맞춰 mSync 응용 시스템을 개발하는 방법에 대해 설명한다.

## 응용 시스템 개발 과정

mSync 응용 시스템을 개발하는 순서는 다음과 같다.

1. 동기화 할 데이터베이스 서버의 테이블들과 파일들을 확인한다.
2. mSync admin 도구를 이용하여 서버 스크립트를 작성하거나 파일들을 등록하고 모바일 기기의 사용자를 등록한다.
3. mSync admin 도구에서 설정한 스크립트에 맞춰 클라이언트 프로그램을 작성한 후 전체 응용 시스템을 구동한다.

3.2장 ~ 3.4장에서는 데모 프로그램을 예제로 하여 mSync 응용 시스템을 개발하는 방법을 설명한다.

### 데모 프로그램 설명

mSync에서 제공하는 데모 프로그램(InervitM\_Demo)은 간단한 주소록 관리 프로그램으로 사용자 아이디, 이름, 회사명, 핸드폰 번호, 전화번호, 주소 등을 입력하고 수정할 수 있는 클라이언트 프로그램이다. 각 레코드는 openML에 저장되며, openML의 client API set도 포함하고 있다(자세한 사항은 openML 매뉴얼 참조).

데이터베이스 서버에 레코드를 추가, 변경하거나 모바일 기기의 mSync 클라이언트에서 레코드를 추가, 변경한 후 동기화 버튼을 클릭하면 각 변경 내용이 데이터베이스 서버와 모바일 기기의 openML에 반영되어 동기화가 이루어졌음을 확인할 수 있다.

### 테이블 스키마

표 3-1은 데모 프로그램에서 사용하는 테이블들의 스키마로 데이터베이스 서버의 테이블 이름은 customer\_demo이고 클라이언트 데이터베이스의 테이블 이름은 customer\_table이다. 서버 테이블은 아래 구조에 맞게 데이터베이스 서버에 만들어 줘야 하고 클라이언트 테이블은 데모 프로그램에서 자동으로 Inervit Mobile DB에 생성한다.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 서버 테이블 | |  | 클라이언트 테이블 | |
| contact\_place | |  | contact | |
| ID | Int | Primary key | id | int |
| username | varchar(32) | username | varchar(32) |
| Company | varchar(255) |  | Company | varchar(255) |
| Name | varchar(255) |  | Name | varchar(255) |
| E\_mail | varchar(255) |  | E\_mail | varchar(255) |
| Company\_phone | varchar(20) |  | Company\_phone | varchar(20) |
| Mobile\_phone | varchar(20) |  | Mobile\_phone | varchar(20) |
| address | Varchar(255) |  | address | Varchar(255) |
| mTime | DateTime |  | mTime | DateTime |
| dflag | Char(1) |  |  |  |

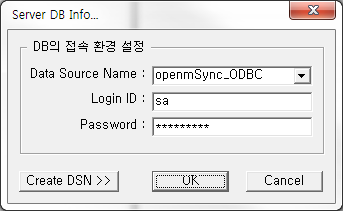
<표 3-1> 데모 프로그램 테이블 스키마

## 서버 스크립트 작성 방법

### 데이터베이스 서버 연결

mSync admin 도구를 실행하여 서버 스크립트를 작성하기 위해서 먼저 데이터베이스 서버로 연결 환경을 설정해주어야 한다. 툴바의 Connection Setup 버튼( 또는 Ctrl+O )을 클릭하여 2.3장에서 생성한 시스템 테이블에 대한 연결 환경을 설정한다. 만약 DSN이 설정되지 않은 경우에는 “***Create DSN >>***“ 버튼을 클릭하여 설정한다. (자세한 사항은 2.4장 참조)

* + Data Source Name : 현 시스템에 설정되어 있는 ODBC의 Data Source Name을 보여주므로 앞서 생성한 시스템 테이블의 DSN을 선택
* Login ID : 데이터베이스 서버 접속을 위한 DB 사용자 ID
* Password : 사용자의 암호



<그림 3-1> 데이터베이스 서버 연결 화면

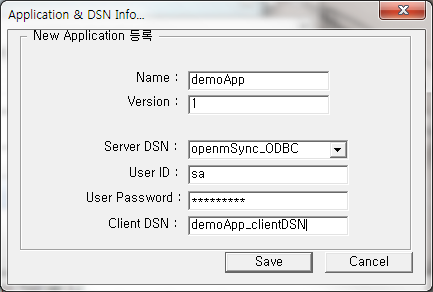
### 서버 스크립트 작성

mSync admin 도구를 실행하여 툴바의 Sync Script 등록 버튼( 또는 Ctrl+S )를 선택하여 서버 스크립트를 작성하는 방법에 대해 설명한다.

클라이언트 프로그램(이하 application이라 한다)이 어떤 데이터베이스 서버(DSN)에 있는 테이블의 어느 필드들을 동기화 작업에 사용할 것인지 지정하기 위해 서버 스크립트를 만들어야 하는데, 먼저 application과 해당 데이터베이스를 등록하고 테이블이나 스크립트를 추가하도록 한다. 하나의 application에서 여러 데이터베이스 서버를 액세스하고자 하면 추후에 “***Add DSN***” 메뉴를 통해 데이터베이스를 추가하면 된다.

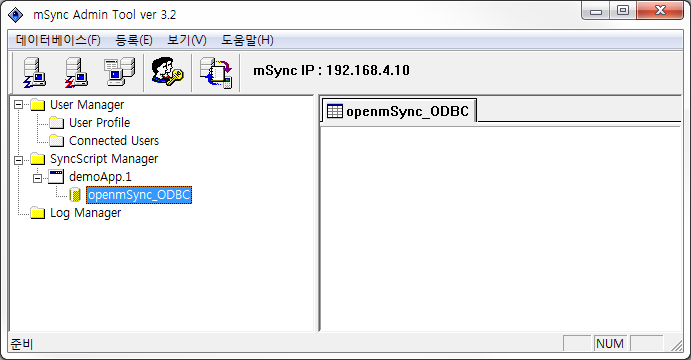
1. Application과 Version, DSN 등록
   * Name : application의 이름

* Version : application의 버전으로 1부터 50까지의 정수 값
* Server DSN : 동기화 작업을 할 테이블들이 있는 데이터베이스 서버의 DSN
* User ID : 데이터베이스 서버 접속을 위한 DB 사용자
* User Password : 사용자의 암호
* Client DSN : 모바일 기기에서 인식하는 DSN 값으로 Server DSN과 같아도 되며 단말기와 별도로 서버의 DSN 값을 수정할 수 있도록 Client DSN을 제공한다.



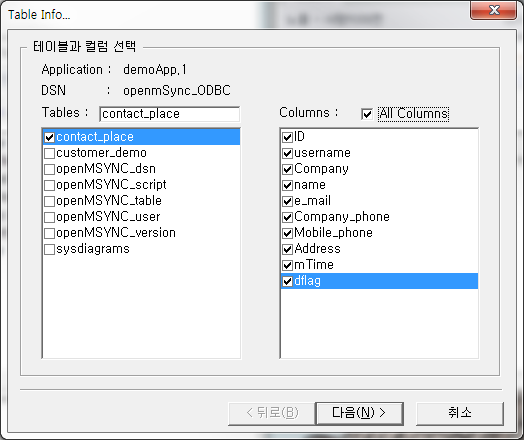
<그림 3-2> Application과 DSN을 지정하는 화면

1. 그림 3-4와 같이 openmSync\_ODBC라는 DSN을 사용하는 demoApp라는 application이 등록되었음을 알 수 있다. 여기서 DSN(openmSync\_ODBC)명에 오른쪽 마우스버튼을 클릭하면 해당 메뉴들이 보이는데 “***Add Tables***”를 선택하여 서버 스크립트 작성을 계속 진행한다. (다른 메뉴는 4장 참조)



<그림 3-4> Application과 DSN이 등록된 화면

1. 그림 3-5와 같은 화면에서 미리 설정한 Application과 DSN 정보가 표시되고 좌측에는 해당 DSN의 테이블 리스트가 보인다. 그 중 클라이언트 프로그램에서 동기화 할 contact\_place를 선택하고 동기화와 관련없는 컬럼을 제외한 나머지 컬럼을 선택한 뒤 “***다음***” 버튼을 클릭한다.



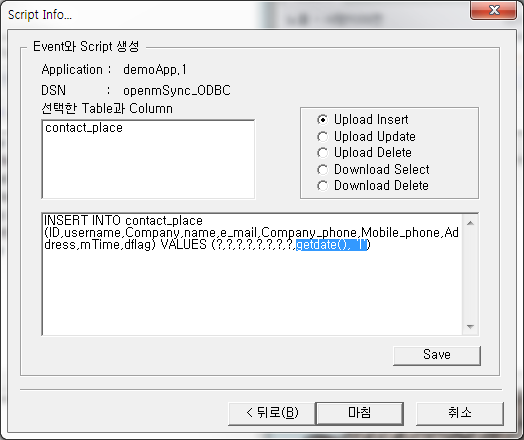
<그림 3-5> 테이블 및 컬럼 선택

1. 이제 마지막으로 이벤트에 해당하는 스크립트를 작성한다 (이벤트에 대한 자세한 설명은 4.1.3.3장 테이블 추가 항목 참조). Upload\_Insert를 선택하면 insert에 대한 스크립트의 레이아웃이 보여진다. 그림 3-6와 같이 스크립트를 고쳐 “***save***” 버튼을 클릭하면 이벤트에 대한 스크립트가 저장된다. 마찬가지 방법으로 표 3-2의 내용과 같이 각각의 스크립트를 작성해서 저장한다.

(스크립트는 전체 시스템 운영 등과 관련해서 구성하게 반드시 모든 스크립트를 만들어야 하는 것은 아니다. 여기서는 예제를 설명하기 위해 전체 스크립트를 모두 만들어 두었다.)

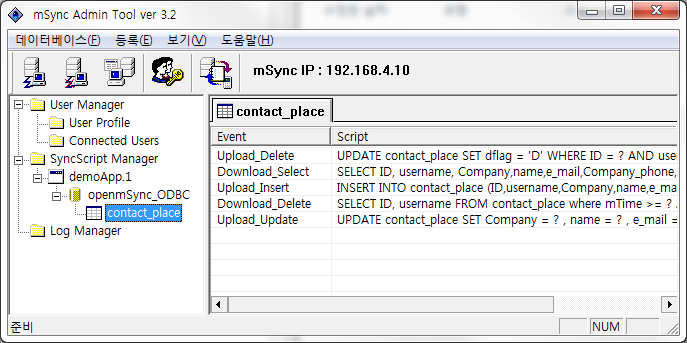
|  |
| --- |
| Upload\_Insert :  INSERT INTO contact\_place (ID, username, Company, name, e\_mail, Company\_phone,  Mobile\_phone, Address,mTime, dflag ) VALUES (?,?,?,?,?,?,?,?,getdate(), 'I')  Upload\_Update :  UPDATE contact\_place SET Company = ? , name = ? , e\_mail = ? , Company\_phone = ? , Mobile\_phone = ? , Address = ? , mTime = getdate(), dflag = 'U' WHERE id = ? AND username = ?  Upload\_Delete:  UPDATE contact\_place SET dflag = 'D' WHERE ID = ? AND username = ?  Download\_Select :  SELECT ID, username, Company,name,e\_mail,Company\_phone,Mobile\_phone,Address  FROM contact\_place where mTime >= ? AND dflag <> 'D'  Download\_Delete:  SELECT ID, username FROM contact\_place where mTime >= ? AND dflag = 'D' |

<표 3-2> 서버 스크립트



<그림 3-6> 스크립트 추가 화면

1. “***마침***” 버튼을 클릭하면 그림 3-7과 같은 화면이 나오고 openmSync\_ODBC 데이터베이스의 contact\_place 테이블에 대한 스크립트 다섯 개가 작성된 것을 알 수 있다.

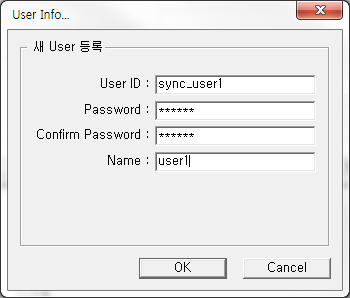


<그림 3-7> 스크립트 저장 완료 화면

## 사용자 등록

3.1장에서 설명한 바와 같이 서버 스크립트의 작성을 마쳤으면 모바일 기기의 사용자를 등록해 주어야 mSync 응용 시스템을 사용할 수 있다. mSync admin 도구 툴바의 User 등록 버튼 (  또는 Ctrl+U )을 클릭하여 mSync 서버를 통해 데이터베이스 서버에 접근하는 사용자를 등록하고 정보를 관리한다.

* User ID : 동기화 작업을 허용할 모바일 기기의 사용자 ID
* Password : 사용자의 암호
* Name : 사용자 이름 (사용자 관리를 위해 필요한 것으로 입력하지 않아도 된다.)



<그림 3-8> 모바일 기기 사용자 등록 화면

본 예제에서는 두 개의 모바일 기기간의 동기화를 사용하기 위해 sync\_user1과 sync\_user2 두 명을 등록한다.

## 클라이언트 프로그램 작성 방법

이제 서버 스크립트에 맞춰 클라이언트 프로그램을 작성하는 방법에 대해 설명한다.

다음 설명하는 예제는 Android App.인 JSyncDemo의 코드 중 일부를 발췌해서 정리 후 보여주는 것이다.

전체 코드 및 세부 사항은 해당 코드를 참고하도록 한다.

여기서는 주요 함수의 사용 방법이나 함수 호출 순서에 대해서 설명할 수 있도록 재 편집된 코드이다.

|  |
| --- |
| 1 mSyncClientJNI.mSync\_InitializeClientSession(logfile\_name, mTimeout );  2 mSyncClientJNI.mSync\_ConnectToSyncServer( mIP, mPort);  3 mSyncClientJNI.mSync\_AuthRequest(user, pwd, "demoApp", g\_Version);  4 SYNC\_TABLE();  5 mSyncClientJNI.mSync\_Disconnect(mSyncClientJNI.NORMAL); |

Line 1: 모바일 기기의 소켓을 초기화, 로그파일 설정, 네트워크 타임아웃 설정한다. 로그파일을 설정한 후 필요시 mSync\_ClientLog (char \*format,...)함수를 이용하여 로그 내용을 남길 수 있다.

Line 2: 서버의 IP와 PORT를 입력하여 mSync 서버와 연결

Line 3: 사용자 ID와 암호는 3.3장 <그림 3-8>에서 등록한 sync\_user1이고 application과 version은 3.2.2장 <그림 3-2>에서 작성한 정보인 “demoApp” 과 1로 각각 설정한다

mSync\_AuthRequest() 함수의 리턴값을 검사해서 적절하게 대응해야 한다.

주요 리턴값에 대해서는 다음과 같은 의미를 갖는다.

UPGRADE\_FLAG : 버전 업그레이드가 되었다. 이 경우 App을 새로 받아야 할 것이다.

NACK\_FLAG : 사용자 인증에 실패했다. 등록된 사용자가 아니거나 암호가 맞지 않은 경우이다.

ACK\_FLAG : 인증 절차가 정상 완료되었다.

Line 4: 실제 동기화를 처리하는 내용이 들어가게 되며 여기서는 SYNC\_TABLE()함수를 호출해서 처리한다.

Line 5: mSync 서버와의 접속을 해제한다. 이때 정상적인 종료인 경우에는 NORMAL를, 비정상적인 경우(에러를 만났을 때)에는 ABNORMAL를 파라미터로 넘기도록 한다.

다음은 SYNC\_TABLE()함수의 구성이다.

|  |
| --- |
| 1 mSyncClientJNI.mSync\_SendDSN(“demoApp\_clientDSN”);  2 mSyncClientJNI.mSync\_SendTableName(nSyncMode, ServerTable);  3 Do\_Upload(INSERT\_FLAG);  4 Do\_Upload(UPDATE\_FLAG);  5 Do\_Upload(DELETE\_FLAG);  6 mSyncClientJNI.mSync\_UploadOK(download\_param);  7 while( (bData = mSyncClientJNI.mSync\_ReceiveDownloadData2()) != NULL )  8 {  9 Do\_Download\_Select(bData);  10 }  11 while( (bData = mSyncClientJNI.mSync\_ReceiveDownloadData2()) != NULL )  12 {  13 Do\_Download\_Delete(bData);  14 } |

Line 1: <그림 3-2>의 Client DSN 명을 지정해서 서버의 어느 데이터 베이스를 사용할 지 결정한다.

mSync 서버에서 ODBC를 설정하면서 지정하는 openmSync\_ODBC와는 다른 것이라는 점에 주의해야 한다.

Line 2: 동기화 대상 테이블과 동기화 대상 범위를 지정한다.

대상 테이블은 <그림 3-5>에서 지정한 Tables의 이름을 사용한다.

동기화 대상 범위는 ALL\_SYNC와 MOD\_SYNC로 나누어지는데 ALL\_SYNC는 전체 레코드를 대상으로 하고 MOD\_SYNC는 마지막 동기화 했던 시점 이후의 레코드에 대해서만 동기화 시도를 한다는 점이 다르다. 또한, 이 동기화 mode값의 설정은 Download 수행 시 관여되는 것이다.

Line 3~5: Upload할 대상이 있다면 Do\_Upload()를 호출해서 처리한다. 이때 각각 INSERT\_FLAG,

UPDATE\_FLAG, DELETE\_FLAG로 처리할 대상 레코드의 속성을 지정할 수 있으며 그 순서에는 상관이 없다. Do\_Upload() 함수 내에서는 mSyncClientJNI.mSync\_SendUploadData(flagStr, DataStr)를 이용하는데 복수개의 레코드를 (전체 길이가 HUGE\_STRING\_LENGTH보다는 작아야 한다) DataStr에 지정된 format으로 구성해서 전송하게 되며, 데이터가 계속해서 존재하면 동일과정을 반복해서 처리한다. 이때 필드값들의 순서에 주의해야 한다.

Line 5: 더 이상 upload 할 것이 없음을 서버 쪽에 알려주며, 동시에 Download할 때 필요한 정보도 함께 전달한다. 만약, upload할 것이 없었다고 하더라도 download 전에 mSync\_UploadOK() 함수는 반드시 호출되어야 한다.

이 함수의 파라미터로는 다운로드 스크립트에서 지정한 파라미터의 순서에 맞춰 메시지와 동일한 포맷으로 전달한다.

Line 7~10 : mSync\_UploadOK() 이후에 mSync\_ReceiveDownloadData2()는 Download\_Select 에 대한 동작이다. 즉, 서버에서 Download\_Select script에 따른 결과를 받아오게 된다.

코드에서 보듯이 NULL을 리턴 할 때까지 반복해서 받아오며(한번에 받은 데이터에도 여러 개의 레코드가 들어 있을 수 있다) Do\_Download\_Select() 함수를 통해서 처리하도록 한 형태의 코드이다.

Line 11~14: Download\_Select에 대한 처리가 모두 끝난 후에 다시 한번

mSync\_ReceiveDownloadData2()를 호출하게 되면 Download\_Delete에 해당하는 레코드를 얻어 올 수 있게 된다. 처리 형태는 Download\_Select와 같은 개념이다.

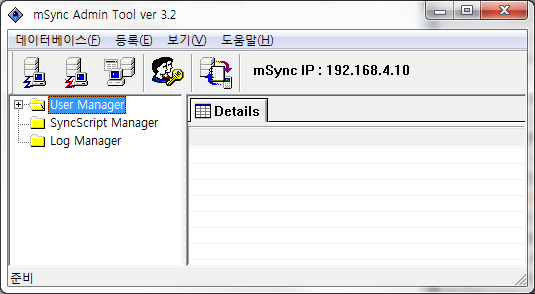
한 데이터베이스 서버의 여러 테이블들을 동기화 하려면 SYNC\_TABLE() 함수의 2~14라인이 반복되고 다른 데이터베이스 서버의 테이블들을 동기화 하려면 SYNC\_TABLE() 함수내의 1~14 라인이 반복된다.

# mSync 도구 및 API

## mSync admin 도구

### mSync admin 도구 실행

Admin.exe 프로그램을 실행시키면 그림 4-1과 같은 화면이 보이며 각 메뉴에 대한 소개는 다음과 같다. mSync IP는 mSync 서버와 동일 시스템에서 mSync admin을 실행시킬 경우에 mSync 서버의 IP를 나타낸다.



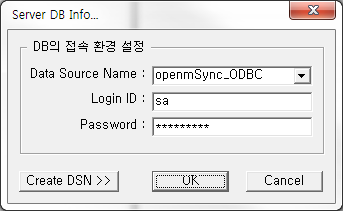
<그림 4-1> mSync admin 도구 기본 화면

### 기본 메뉴 설명

#### 연결 환경 설정 ( 또는 Ctrl+O )

Admin 도구를 사용하기 위해서는 먼저 2.3 장에서 생성한 시스템 테이블이 있는 데이터베이스 서버에 대한 접속 환경을 설정해 주어야 한다. 만약 DSN이 설정되지 않은 경우에는 “***Create DSN >>***“ 버튼을 눌러 DSN을 먼저 설정한다. (자세한 사항은 2.4 장 참조)

* + Data Source Name : 현 시스템에 설정되어 있는 ODBC의 Data Source Name을 보여주므로 앞서 생성한 시스템 테이블의 DSN을 선택
* Login ID : 데이터베이스 서버 접속을 위한 DB 사용자 ID
* Password : 사용자의 암호



<그림 4-2> 데이터베이스 서버 연결 화면

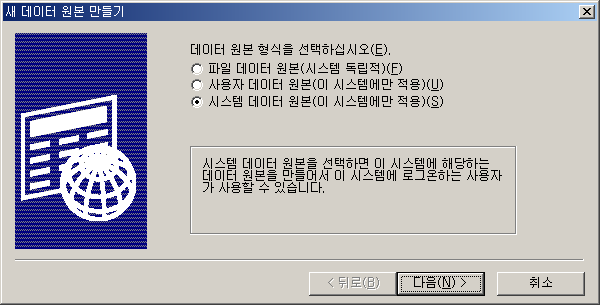
#### 연결 환경 해제 ( 또는 Ctrl+D )

시스템 테이블의 데이터베이스에 대한 연결 환경이 바뀌거나 재 접속하고자 하는 경우 먼저 환경을 해제해야 하며 환경 설정이 해제된 후에는 사용자나 스크립트의 내용을 볼 수 없으므로 접속 환경을 다시 설정해줘야 한다. (4.1.2.1 장 참조)

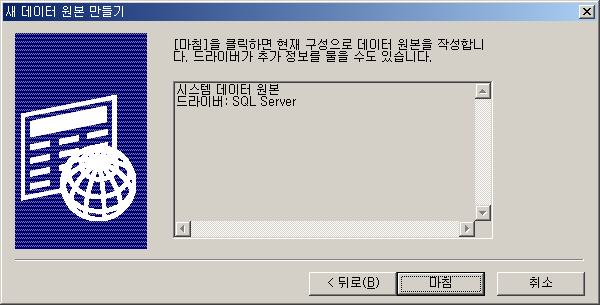
#### DSN 생성 ( 또는 Ctrl+C )

ODBC는 데이터베이스로 접근하기 위한 인터페이스로 여기서는 MS-SQL Server의 설정 방법에 대해 설명한다. (Admin 도구를 실행하기 전에 제어판을 통해 설정할 수도 있다. 2.4장 참조)

1. “***DSN 생성***”을 선택한 뒤 나오는 그림 4-3-a과 같은 화면에서 시스템 데이터 원본을 선택하고 “***다음 >***”을 클릭하면 그림 2-8-a와 같은 화면이 보여진다.. 여기에서 SQL Server를 선택하고 “***다음***”을 클릭하면 그림 4-3-b와 같은 화면이 보이며 “***마침***”을 클릭하면 계속하여 정보를 입력하는 화면이 보여진다.



<그림 4-3-a> 신규 데이터 원본 만들기 화면



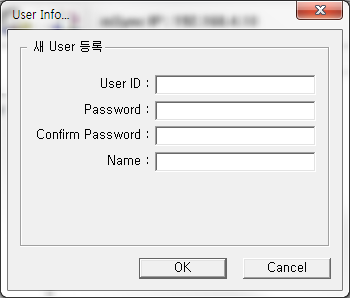
<그림 4-3-b> 신규 데이터 원본 만들기 화면

1. 이어지는 화면은 2.4.3장의 ②~④과정과 동일하며 ODBC의 정보를 입력하고 테스트 하는 화면으로 진행된다.

#### 사용자 등록 ( 또는 Ctrl+U )

mSync 서버를 통해 데이터베이스 서버에 접근하여 데이터를 동기화하는 모바일 기기 사용자의 정보를 등록 및 관리하며 모든 등록, 변경 메뉴는 연결 환경(4.1.2.1 장)을 설정한 후에 사용할 수 있다.

* + User ID : 동기화 작업을 허용할 모바일 기기의 사용자 ID
* Password : 사용자 암호
* Name : 사용자 이름 (사용자 관리를 위해 필요한 것으로 입력하지 않아도 된다.)

****

<그림 4-4> 모바일 기기 사용자 등록 화면

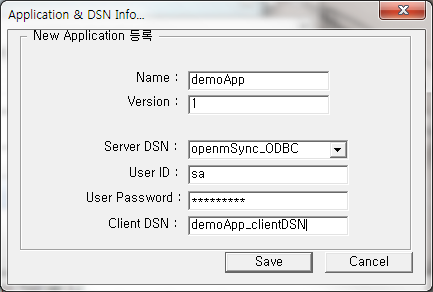
#### Sync Script 등록 ( 또는 Ctrl+S )

Application이 어떤 데이터베이스 서버(DSN)에 있는 테이블의 어느 필드들을 동기화 작업에 사용할 것인지 지정하기 위해 서버 스크립트를 만들어야 하는데 일단 Sync Script 등록을 통해서 application과 데이터베이스를 등록하고 테이블이나 서버 스크립트를 등록하는 과정을 진행한다. (테이블이나 스크립트의 추가는 4.1.3.3 장과 4.1.3.4 장 참조)

하나의 application에서 여러 데이터베이스 서버를 액세스하고자 하면 추후에 “***Add DSN***” 메뉴를 통해 데이터베이스를 추가하면 된다.

① Application과 Version, DSN 등록

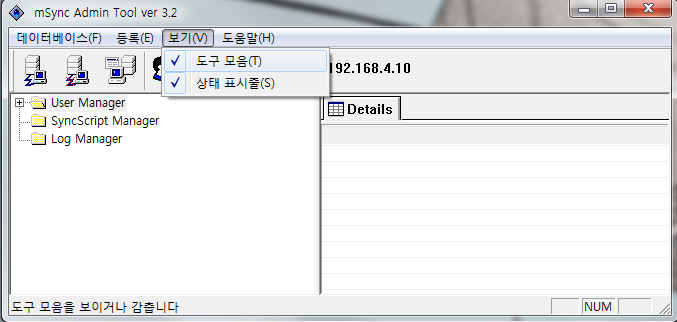
* + Name : application의 이름
* Version : application의 버전으로 1부터 50까지의 정수 값
* Server DSN : 동기화 작업을 할 테이블들이 있는 데이터베이스 서버의 DSN
* User ID : 데이터베이스 서버 접속을 위한 DB 사용자 ID
* User Password : 사용자의 암호
* Client DSN : 모바일 기기에서 인식하는 DSN 값으로 Server DSN과 같아도 되며 단말기와 별도로 서버의 DSN 값을 수정할 수 있도록 Client DSN을 제공한다.



<그림 4-5> Application과 DSN을 지정하는 화면

#### 보기 메뉴

보기 메뉴에서는 “도구모음”과 “상태표시줄”의 Display 여부를 설정할 수 있다.



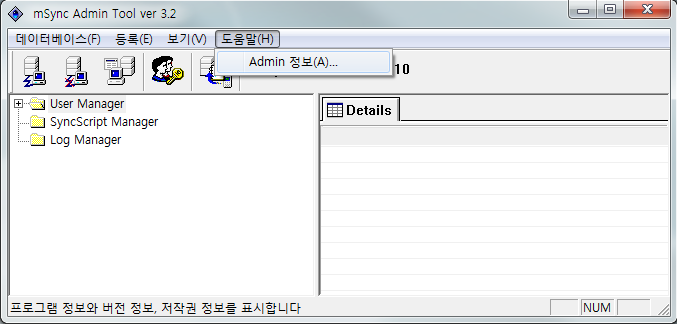
도구모음 창

도구모음 창

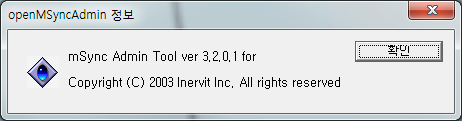
<그림 4-6> 보기 메뉴

#### 도움말 메뉴

도움말 메뉴에서는 <그림 4-8>과 같이 Admin 툴의 버전 정보 등을 제공해 준다.



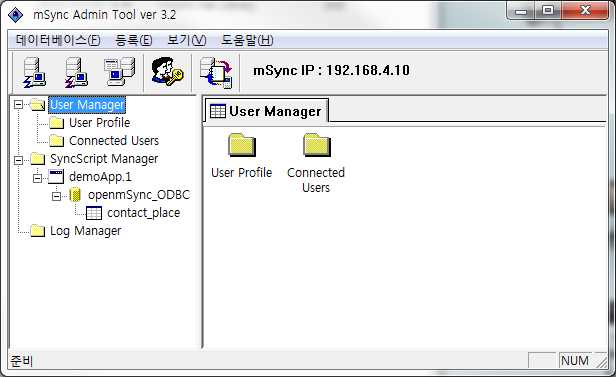
<그림 4-7> 도움말 메뉴



<그림 4-8> Admin 툴 버전 정보

### 세부 메뉴 설명

그림 4-9은 4.1.2.1 장을 통해 연결 환경을 설정한 후에 SyncScript Manager를 펼친 화면이다. Admin 도구에서 관리할 수 있는 것은 크게 사용자, SyncScript, 그리고 동기화 작업에 대한 로그로 나누어 볼 수 있으며 이 세가지가 왼쪽 메뉴 창에 나타난다.

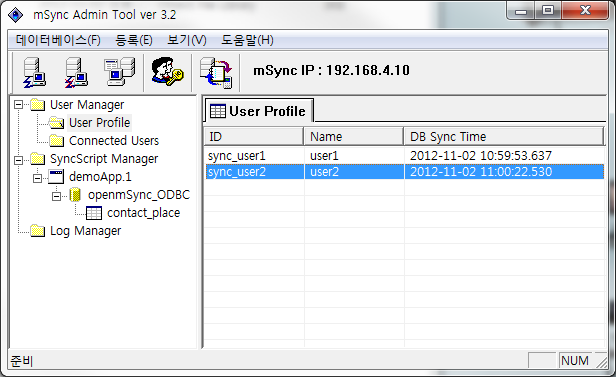


<그림 4-9> DB 연결 설정 후 화면

#### User Manager 메뉴

##### 사용자 정보 보기

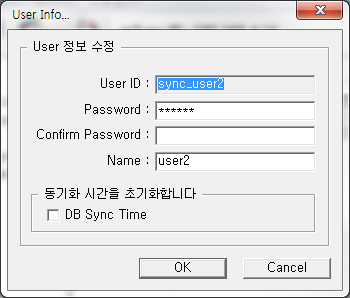
User Manager의 서브 메뉴 중 ***User Profile***을 선택하면 등록되어 있는 사용자의 ID, 이름, DB 에 대한 최근 동기화 시간을 볼 수 있고 정보 창에 나타나는 항목들 중 하나를 선택해서 오른쪽 마우스 버튼을 클릭하면 사용자 정보를 수정(Modify User)하거나 삭제(Delete User)할 수 있다.



<그림 4-10> 사용자 정보 보기 화면

* 사용자 정보 수정 (Modify User)

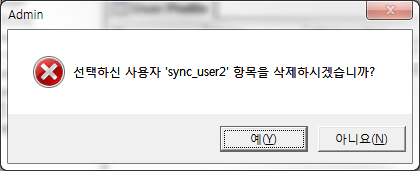
사용자 정보 수정은 ID를 제외한 값들이 수정 가능하다. 또한, 동기화 시간(DB Sync Time)을 초기화 함으로써 해당 사용자가 전체 데이터를 다시 동기화 하도록 할 수 있다.



<그림 4-11> 사용자 정보 수정 화면

* 사용자 정보 삭제 (Delete User)

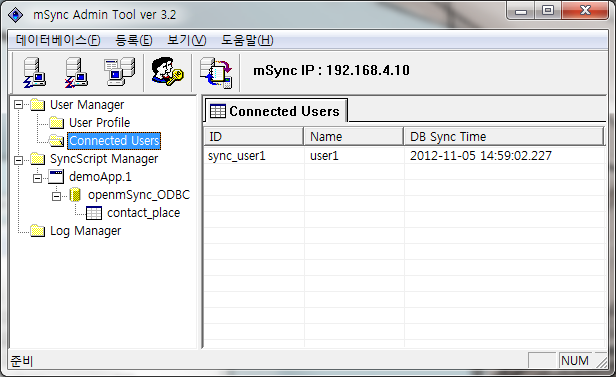
사용자 정보 삭제를 선택하면 아래와 같이 확인하는 화면이 나오고 “***예***”를 선택하게 되면 해당 사용자에 대한 정보가 삭제되며 정보 창에서도 항목이 삭제된다.



<그림 4-12> 사용자 정보 삭제 화면

##### 현재 접속중인 사용자 정보 보기

User Manager의 서브 메뉴 중 ***Connected Users***를 선택하면 현재 동기화 진행중인 사용자들의 정보를 볼 수 있으며 아래 그림과 같은 화면이 보여진다.

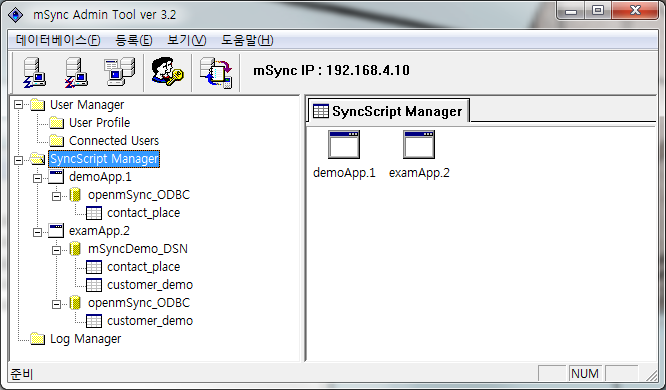


<그림 4-13> 동기화 진행중인 사용자 정보 보기 화면

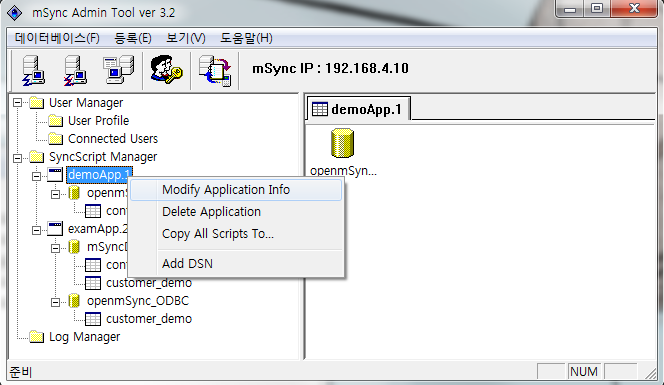
#### Application 메뉴

mSync 서버에서 관리하는 application의 구조는 <그림 4-14>과 같아서 application이 사용하는 데이터베이스 서버와 그에 속해 있는 테이블들이 트리 구조를 이루게 된다. 하나의 application은 하나의 데이터베이스, 하나의 테이블들로 동기화 작업이 이루어질 수도 있고, 여러 테이블들을 포함하는 여러 개의 데이터베이스로 동기화 작업이 이루어질 수도 있다. <그림 4-14>에서 demoApp라는 application은 버전 1이고 openmSync\_ODBC라는 DSN으로 연결되는 데이터베이스 서버의 contact\_place 테이블에 대한 동기화를 수행하도록 설정되고 있음을 볼 수 있다. 반면에 examApp라는 application은 두 개의 DSN을 사용하고 이때 mSyncDemo\_DSN에서는 2개의 테이블을 사용하고 있음을 볼 수 있다.

이와 같이 하나의 Application은 여러 개의 DSN이 연결 될 수 있고, 하나의 DSN에는 여러 개의 table이 연결되어 구성될 수 있다.



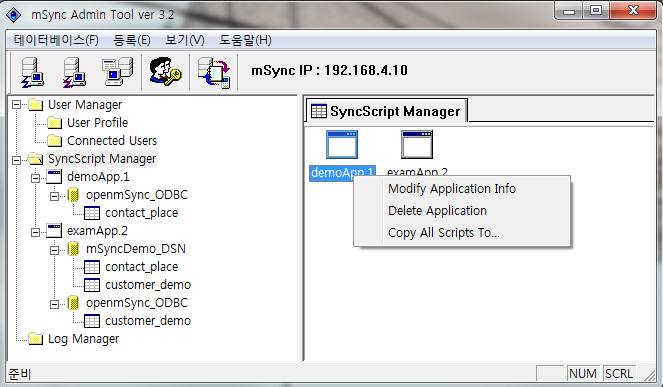
<그림 4-14> Application의 구조



<그림 4-15> Application 메뉴 – 메뉴 창 화면

<그림 4-15>은 메뉴 창의 application 중 하나를 선택하여 오른쪽 마우스 버튼을 클릭했을 때의 화면으로 application의 관리에 필요한 4개의 메뉴가 나타난다. 이 메뉴를 통해 application의 정보를 수정(Modify Application Info)하거나 삭제(Delete Application)할 수 있고 다른 application으로 스크립트를 모두 복사(Copy All Scripts To…)할 수도 있으며 데이터베이스(Add DSN)를 추가 할 수도 있다.

메뉴 창의 SyncScript Manager를 클릭하면 오른쪽 정보 창에는 전체 application들의 아이콘이 보이게 되는데 <그림 4-16>는 그 중 하나의 application을 선택하여 오른쪽 마우스 버튼을 클릭했을 때의 화면으로 이 메뉴를 통해 application의 정보를 수정(Modify Application Info) 하거나 삭제(Delete Application)할 수 있고 역시, 다른 application으로 스크립트를 모두 복사(Copy All Scripts To…)할 수 있다. 단, 데이터베이스 추가는 메뉴 창의 application 메뉴를 통해서만 가능하다.



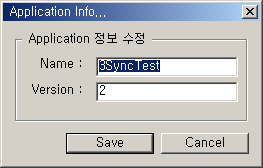
<그림 4-16> Application 메뉴 – 정보 창 화면

* Application 정보 수정 (Modify Application Info)

Application의 정보는 application의 이름과 버전으로 구성되며 이 두 개의 값을 수정할 수 있다. 단, 버전은 1부터 50까지의 정수만 가능하다.

여기서 버전 정보가 증가하면 디바이스용 클라이언트도 갱신된 버전을 받아서 사용해야 한다.

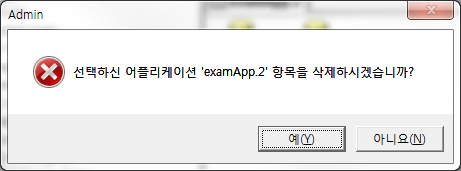
* + Name : 새로운 application의 이름
* Version : Application의 새 버전 값



<그림 4-17> Application 정보 수정 화면

* Application 정보 삭제 (Delete Application)

Application 정보 삭제를 선택하면 아래와 같이 확인하는 화면이 나오고 “***예***”를 선택하게 되면 해당 application에 등록되어 있던 모든 데이터베이스, 테이블, 스크립트, 파일 정보가 삭제되며 메뉴 창에서도 항목이 삭제된다.



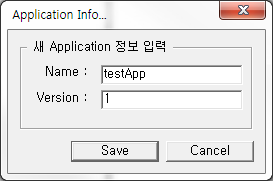
<그림 4-18> Application 정보 삭제 화면

* Application의 스크립트 전체 복사 (Copy All Scripts To…)

기존 application과 비슷한 데이터들을 동기화 하면서 지정해야 할 테이블의 개수가 많은 application을 개발할 경우 비슷한 서버 스크립트를 작성하는 데에 비슷한 작업을 반복해야 하는 것은 번거로운 일이다. 따라서 mSync admin 도구는 하나의 application에 속해 있는 모든 데이터베이스, 테이블, 스크립트 정보를 새 application으로 복사하는 기능을 제공하며 이 기능을 이용하여 기존의 application으로부터 스크립트 전체를 복사한 뒤 필요 없는 데이터베이스나 테이블, 스크립트는 삭제하고 필요한 정보들은 추가하여 편리하게 스크립트를 작성할 수 있다.

복사할 application을 선택 후 오른쪽 버튼을 눌러 나온 메뉴 중에 ***Copy All Scripts To…***를 선택하면 <그림 4-19>과 같은 화면이 보이며 여기에서 새로운 application의 이름과 버전을 입력하면 된다.

* + Name : 복사될 새로운 application의 이름
* Version : Application의 버전 값 (1부터 50까지의 정수 값)

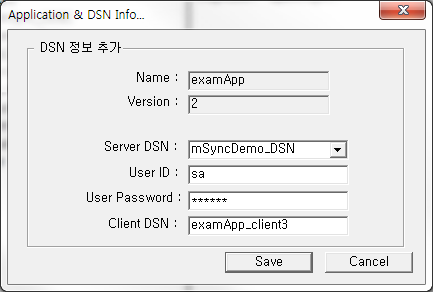


<그림 4-19> 복사될 application의 정보 입력

* 데이터베이스 추가 (Add DSN)

4.1.2.5 장에서 application을 등록할 때에는 기본적으로 하나의 데이터베이스만이 등록되므로 다른 데이터베이스의 테이블을 동기화 하고자 하는 경우에는 데이터베이스의 정보(DSN)를 추가해야 한다. 아래는 ***Add DSN*** 메뉴를 선택한 그림으로 Name과 Version 항목은 이미 선택한 application의 이름과 버전이 들어가며 수정 불가능하다.

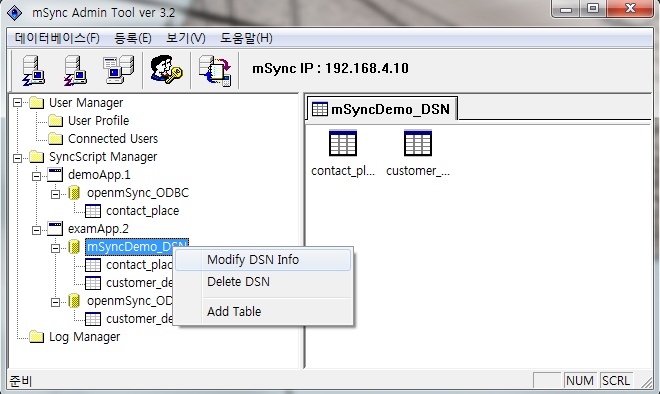
* Server DSN : 추가할 테이블들이 있는 데이터베이스의 DSN을 선택
* User ID : 데이터베이스 서버 접속을 위한 DB 사용자
* User Password : 사용자의 암호
* Client DSN : 4.1.2.5 장에서 설명한 것과 같이 단말 응용 프로그램에서 인식하는 DSN 값이므로 Server DSN과 같거나 혹은 별도의 다른 이름으로 입력



<그림 4-20> 데이터베이스 추가 화면

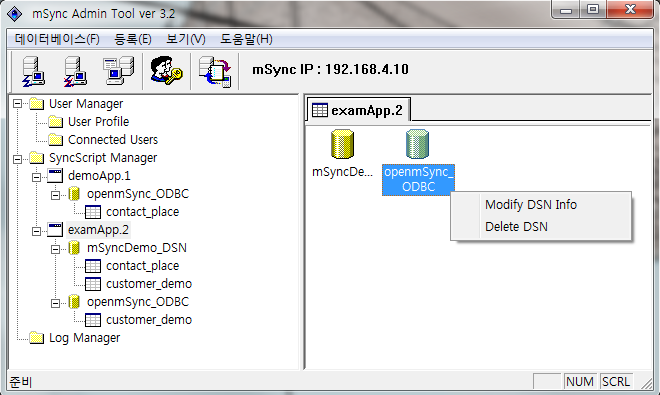
#### 데이터베이스(DSN) 메뉴

<그림 4-21>은 메뉴 창의 DSN 중 하나를 선택하여 오른쪽 마우스 버튼을 클릭했을 때의 화면으로 DSN의 관리에 필요한 세 개의 메뉴가 나타난다. 이 메뉴를 통해 DSN의 정보를 수정(Modify DSN Info)하거나 삭제(Delete DSN)할 수 있고 테이블을 추가(Add Table)할 수 있다.



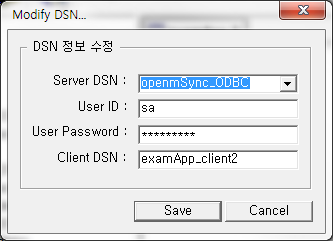
<그림 4-21> 데이터베이스 메뉴 – 메뉴 창 화면

메뉴 창의 application 하나를 클릭하면 오른쪽 정보 창에는 선택한 application에 속해 있는 전체 DSN들의 아이콘이 보이게 되는데 <그림 4-22>은 그 중 하나의 DSN을 선택하여 오른쪽 마우스 버튼을 클릭했을 때의 화면으로 이 메뉴를 통해 DSN의 정보를 수정(Modify DSN Info)하거나 삭제(Delete DSN)할 수 있다. 단, 테이블 추가는 메뉴 창의 DSN 메뉴를 통해서만 가능하다.



<그림 4-22> 데이터베이스 메뉴 – 정보 창 화면

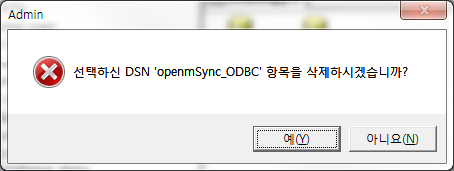
* DSN 정보 수정 (Modify DSN Info)
* Server DSN : 변경할 데이터베이스의 DSN을 선택한다.
* User ID : 데이터베이스 서버 접속을 위한 DB 사용자 ID
* User Password : 사용자의 암호
* Client DSN : 4.1.2.5 장에서 설명한 것과 같이 단말 응용 프로그램에서 인식하는 DSN 값이므로 Server DSN과 같거나 혹은 별도의 다른 이름으로 입력한다.



<그림 4-23> DSN 정보 수정 화면

* DSN 정보 삭제 (Delete DSN)

DSN 정보 삭제를 선택하면 <그림 4-24>과 같이 확인하는 화면이 나오고 “***예***”를 선택하게 되면 해당 DSN에 등록되어 있던 모든 테이블, 스크립트 정보가 삭제되며 메뉴 창에서도 항목이 삭제된다.

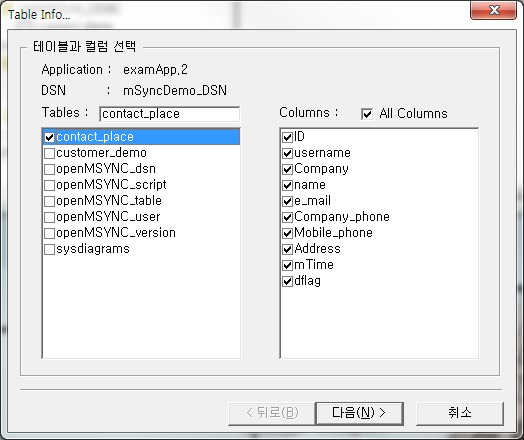


<그림 4-24> DSN 정보 삭제 화면

* 테이블 추가 (Add Table) : 스크립트 정의도 포함

앞서 설명한 바와 같이, 4.1.2.5 장에서 SyncScript를 등록하게 되면 application과 데이터베이스 또는 파일 레벨까지만 등록이 되며 데이터베이스에 속하는 테이블의 스크립트를 지정하기 위해서는 해당 DSN을 선택한 뒤 ***Add Table*** 메뉴를 선택하여야 한다.

<그림 4-25>에서 왼쪽 리스트에는 선택한 DSN에서 접근할 수 있는 테이블들이 나타나므로 먼저 테이블을 한 개 선택하면, ***“Tables:***” 라고 되어 있는 부분에 같은 이름이 등록된다. 이것은 클라이언트쪽에서 인식할 테이블 이름으로 서버의 테이블 이름과 동일해도 되고 만약 서버의 테이블 이름이 길다면 클라이언트쪽에서는 간단하게 이름을 줄여서 사용할 수 있도록 수정한다. 다음으로는 선택한 테이블의 어느 필드들을 동기화에 사용할 것인지 체크하며 만약, 모든 필드를 다 사용할 경우에는 “***All Columns***“ 항목을 체크하면 된다. 위와 같이 세 단계를 거쳐서 테이블의 필드까지 선택한 다음에는 “***다음 >***” 버튼을 클릭한다.

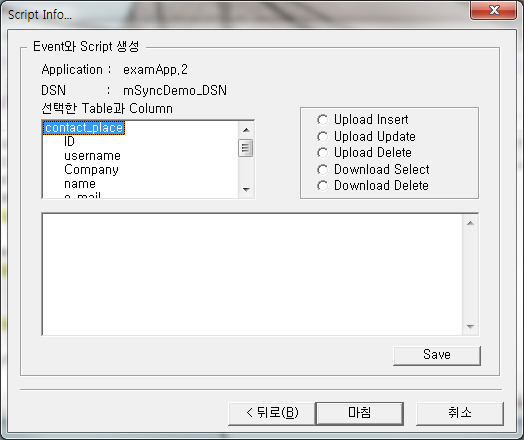


<그림 4-25> 테이블 선택 화면

다음으로 지정할 것은 실제 동기화에서 mSync 서버가 사용할 스크립트이다. 동기화는 크게 업로드와 다운로드 두 단계로 나뉘며 세부적으로는 다섯 이벤트로 나뉜다. 업로드는 클라이언트 쪽에서 변경된 사항을 서버에 반영하는 단계이고 다운로드는 서버에서 변경된 내용을 클라이언트 쪽에 전송하는 단계를 말한다.

* Upload\_Insert : 클라이언트 쪽 DB에서 삽입된 내용을 데이터베이스 서버에 어떻게 반영할 것인지를 정의하는 이벤트
* Upload\_Update : 클라이언트 쪽 DB에서 수정된 내용을 데이터베이스 서버에 어떻게 반영할 것인지를 정의하는 이벤트
* Upload\_Delete : 클라이언트 쪽 DB에서 삭제된 내용을 데이터베이스 서버에 어떻게 반영할 것인지를 정의하는 이벤트
* Download\_Select : 데이터베이스 서버에서 다운로드 할 레코드들을 정의하는 이벤트
* Download\_Delete : 데이터베이스 서버에서 삭제 처리된 레코드들을 정의하는 이벤트

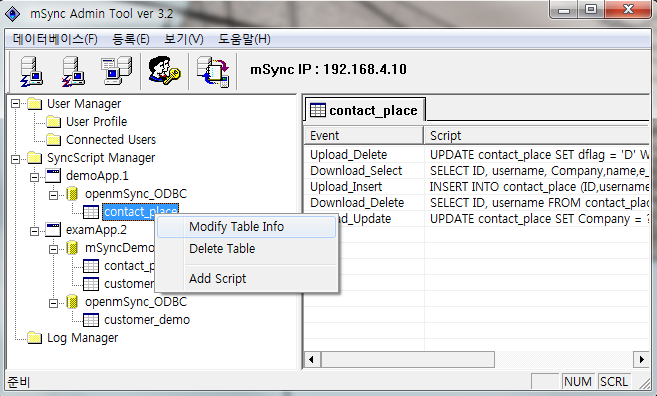
각 이벤트를 선택하면 아래 창에 기본적인 스크립트가 생성되며 비즈니스 로직에 맞게 편집하여 “***Save***” 버튼을 눌러 저장하면 된다. 다른 테이블도 추가하고 싶으면 “***< 뒤로***” 버튼을 눌러 다른 테이블을 선택하고(위에서 정의한 과정과 동일) <그림 4-26>와 같은 화면에서 해당 테이블의 스크립트를 정의하여 저장하면 된다. (<그림 4-26>처럼 “***선택한 Table과 Column***” 항목의 “***emp3***”을 더블 클릭하면 선택한 필드 리스트를 볼 수 있다.)



<그림 4-26> 스크립트 지정 화면

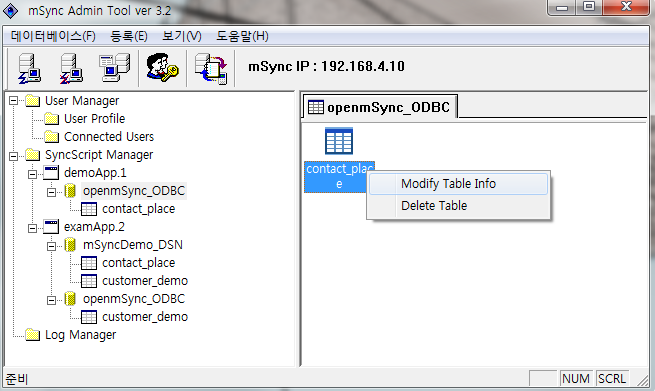
#### Table 메뉴

<그림 4-27>은 메뉴 창의 테이블 중 하나를 선택하여 오른쪽 마우스 버튼을 클릭했을 때의 화면으로 테이블의 관리에 필요한 세 개의 메뉴가 나타난다. 이 메뉴를 통해 테이블의 정보를 수정(Modify Table Info)하거나 삭제(Delete Table)할 수 있고 스크립트를 추가(Add Script)할 수 있다.



<그림 4-27 테이블 메뉴 – 메뉴 창 화면

메뉴 창의 DSN을 클릭하면 오른쪽 정보 창에는 선택한 DSN에 속해 있는 전체 테이블들의 아이콘이 보이게 되는데 <그림 4-28>은 그 중 하나의 테이블을 선택하여 오른쪽 마우스 버튼을 클릭했을 때의 화면으로 이 메뉴를 통해 테이블의 정보를 수정(Modify Table Info)하거나 삭제(Delete Table)할 수 있다. 단, 스크립트 추가는 메뉴 창의 테이블 메뉴를 통해서만 가능하다.

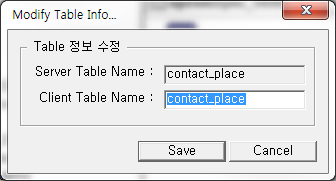


<그림 4-28> 테이블 메뉴 – 정보 창 화면

* 테이블 정보 수정 (Modify Table Info)

테이블의 정보는 서버의 테이블 이름과 클라이언트에서 인식하는 테이블 이름이 있는데 서버의 테이블 이름은 수정이 불가능하며 클라이언트 쪽 테이블 이름만이 수정 가능하다.

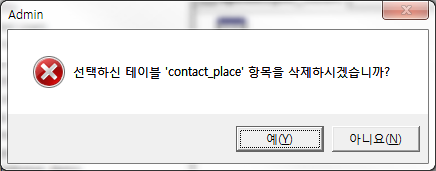
* Client Table Name : 클라이언트 쪽에서 인식하는 테이블 이름으로 Server Table Name과 같을 수도 있고 만약 이 값이 길 경우 줄여서 사용할 수도 있다.



<그림 4-29> 테이블 정보 수정 화면

* 테이블 정보 삭제

테이블 정보 삭제를 선택하면 <그림 4-30>와 같은 화면이 나오고 “***예***”를 선택하게 되면 해당 테이블에 등록되어 있던 모든 스크립트 정보가 삭제되며 메뉴 창에서도 항목이 삭제된다.

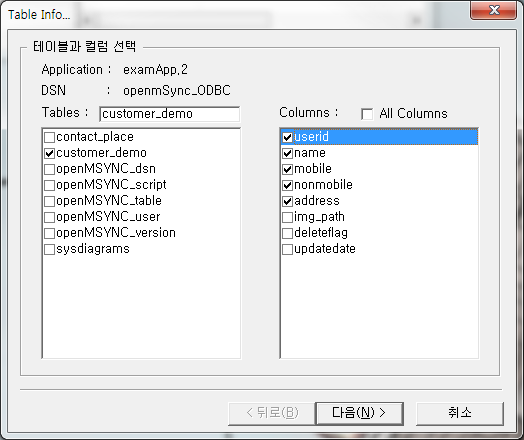


<그림 4-30> 테이블 정보 삭제 화면

* 스크립트 추가

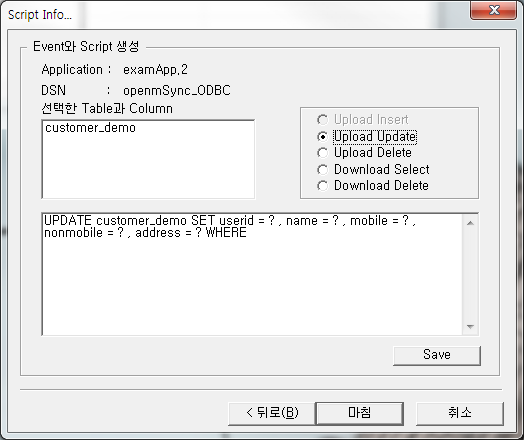
4.1.3.3 장의 테이블 추가 시에 지정한 스크립트 외에 스크립트를 추가하고 싶으면 해당 테이블을 선택한 뒤 ***Add Script*** 메뉴를 선택한다.

<그림 4-28>와 같은 화면에서 왼쪽 리스트에는 해당 DSN에서 접근할 수 있는 테이블들이 나타나는데 선택한 테이블이 체크된 상태로 보이므로 어느 필드들을 동기화에 사용할 것인지 체크 한 뒤 “***다음 >***” 버튼을 클릭한다.



<그림 4-31> 스크립트 추가 화면 - ①

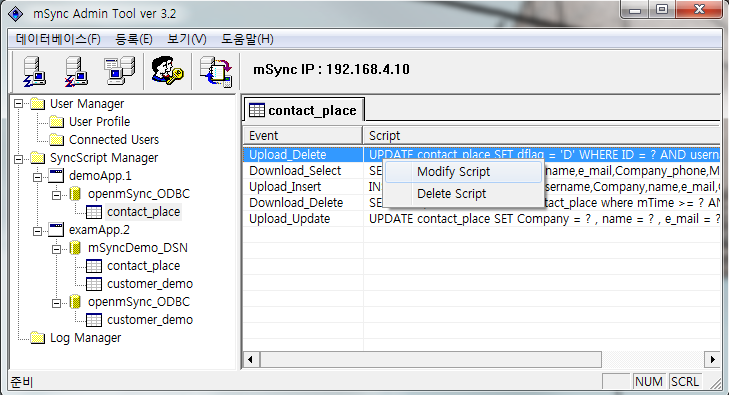
4.1.3.3 장에서 설명한 바와 같이 스크립트의 이벤트를 선택하고 기본적으로 제공되는 스크립트를 편집하여 저장하면 되며 아래 <그림 4-32>과 같이 이미 지정되어 있는 스크립트는 선택이 불가능하게 되어 있다.



<그림 4-32> 스크립트 추가 화면 - ②

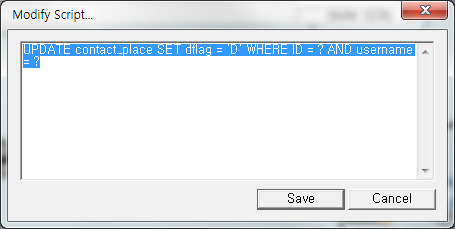
#### Script 메뉴

메뉴 창의 테이블 하나를 클릭하면 오른쪽 정보 창에는 선택한 테이블에 속해 있는 전체 스크립트의 내용이 보이게 되는데 <그림 4-33>는 그 중 하나의 스크립트를 선택하여 오른쪽 마우스 버튼을 클릭했을 때의 화면으로 이 메뉴를 통해 스크립트의 정보를 수정(Modify Script)하거나 삭제(Delete Script)할 수 있다.

<그림 4-33> 스크립트 메뉴 – 정보 창 화면

* 스크립트 정보 수정 (Modify Script)

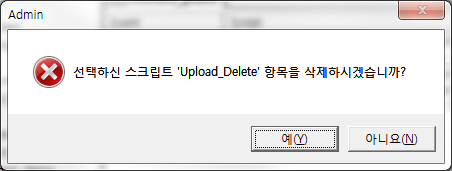
편집 창에 기존 스크립트 정보가 나타나므로 원하는 데이터 셋을 적용할 수 있도록 적절하게 스크립트를 수정한 뒤 “***Save***” 버튼을 눌러 저장한다.



<그림 4-34> 스크립트 수정 화면

* 스크립트 정보 삭제 (Delete Script)

스크립트 정보 삭제를 선택하면 <그림 4-35>와 같이 확인하는 화면이 나오고 “***예***”를 선택하게 되면 정보 창에서도 항목이 삭제된다.

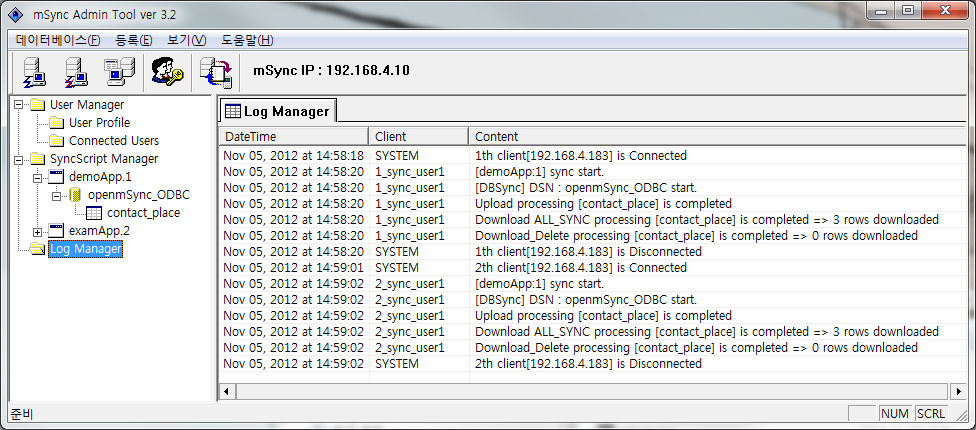


<그림 4-35> 스크립트 삭제 화면

### Log Manager 메뉴

#### 화면

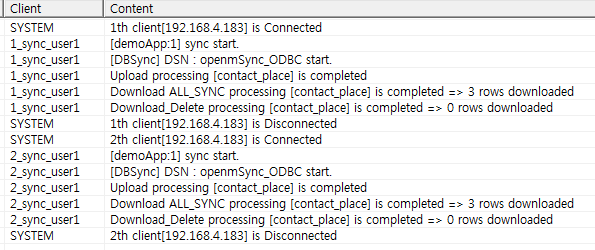
<그림 4-36>은 mSync 서버의 로그를 보여준다.



<그림 4-36> mSync 서버 로그 화면

#### 로그 메시지 설명

<그림 4-37>에서 mSync 서버의 로그 메시지를 보면 로그가 남겨지는 날짜 및 시간, 로그를 남기는 주체(SYSTEM이나 사용자 ID), 로그 내용의 세 부분으로 나뉜다. 클라이언트가 동기화를 요청하면 먼저 클라이언트의 IP와 함께 몇 번째 클라이언트가 접속되었는지 SYSTEM이 “1th client[192.168.4.183] is Connected”을 로그에 남기고 이후 사용자의 동기화 진행 과정이 나타난다.



<그림 4-37> mSync 서버 로그 내용

demoApp라는 application(version은 1)의 동기화가 시작되고 openmSync\_ODBC라는 데이터베이스에 연결되어 contact\_place라는 테이블에 대해 동기화 한 것을 알 수 있다.

다음 <표 4-1>은 mSync 서버에서 남기는 로그 메시지들을 정리한 것으로 굵은 문자열은 mSync 응용 시스템에 맞는 값으로 대체된다. 분류 값을 이용하여 <표 4-2>에서 해결책을 찾아 오류 사항을 수정한다.

|  |  |  |
| --- | --- | --- |
| 분류 | 로그 내용 | |
| ① | SYSTEM | There is some problems in socket() |
| SYSTEM | There is some problems in bind() |
| SYSTEM | There is some problems in listen() |
| SYSTEM | Start Up mSync ver 3.1.0.1 [port: **7777**]: FAIL. |
| ② | SYSTEM | Stopped openMSync™ |
| SYSTEM | Stop openMSync™ |
| ③ | SYSTEM | Start Up mSync ver 3.1.0.1 [port:**7777**] |
| SYSTEM | 최대 사용자 수는 **100**명입니다. |
| ④ | SYSTEM | 13th client[**192.168.123.124**] is Connected |
| SYSTEM | 13dth client[**192.168.123.124**] is Disconnected |
| SYSTEM | 13dth client[**192.168.123.124**] is Disconnected => Error Occurred |
| SYSTEM | Disconnect error : -1 |
| ⑤ | SYSTEM [1] | Connect to System DB[**openmSync\_ODBC**] : FAIL... |
| 2\_**sync\_user1** | Connect to User DB[**openmSync\_ODBC**] : FAIL...; |
| ⑥ | 2\_**sync\_user1** | [Authenticate User]Data Format is Wrong |
| 2\_**Inervit\_1** | [Authenticate User]Invalid User : Password is wrong! |
| 2\_**sync\_user1** | [Authenticate User]There is no user ID[**Inervit\_3**] |
| 2\_**sync\_user1** | [Authenticate User]System Error |
| ⑦ | 2\_**sync\_user1** | 업그레이드를 위한 Application **'demoApp'**의 버전 **'2'**이 존재합니다. |
| ⑧ | SYSTEM [1] | Application **' demoApp '**의 버전 **'1'**에 대한 정보를 찾을 수 없습니다. |
| ⑨ | SYSTEM [1] | Allowed maximun users = **100** |
| ⑩ | 2\_**sync\_user1** | [SyncTest:1] sync start. |
| 2\_**sync\_user1** | [DBSync] DSN : **openmSync\_ODBC** start. |
| 2\_**sync\_user1** | Upload processing [**examTable**] is completed |
| 2\_**sync\_user1** | Download ALL\_SYNC processing [**examTable**] is completed => **4** rows downloaded |
| 2\_**sync\_user1** | Download MOD\_SYNC processing [**examTable**] is completed => **4**rows downloaded |
| 2\_**sync\_user1** | Download\_Delete processing [**examTable**] is completed => **4** rows downloaded |
| 2\_**sync\_user1** | Upload\_Insert : **2** rows affected |
| 2\_**sync\_user1** | Upload\_Delete : **2** rows affected |
| 2\_**sync\_user1** | Upload\_Insert : **2** rows affected |
| ⑪ | SYSTEM | Thread Creation ERROR |
| ⑫ | 2\_**sync\_user1** | There is No script for Upload\_Update |
| 2\_**sync\_user1** | There is No script for Upload\_Delete |
| 2\_**sync\_user1** | There is No script for Upload\_Insert |
| 2\_**sync\_user1** | There is no Script for Download |
| 2\_**sync\_user1** | The size of data in DB is **4**. But the size of data[**abcde**] from **sync\_user1** is **5** |
| 2\_**sync\_user1** | The format of data is wrong. There is no Record Delimeter. |
| 2\_**sync\_user1** | [SCRIPT ERROR] PDA로부터 받은 parameter의 개수가 script에 정의된 parameter의 개수와 맞지 않습니다. |
| 2\_**sync\_user1** | There is no DSN of **'sys4’** which is defined in client |
| ⑬ | SYSTEM[1] | Network Communication Error [Receive] |
| SYSTEM[1] | Network Communication Error [Receive:Timeout] |
| SYSTEM[1] | Network Communication Error [Receive:Message is cut off] |
| 2\_**sync\_user1** | Network Communication Error [Receive] |
| 2\_**sync\_user1** | Network Communication Error [Receive:Timeout] |
| 2\_**sync\_user1** | Network Communication Error |
| 2\_**sync\_user1** | Network Communication Error [Invalid Flag:’**P**’] |
| ⑭ | SYSTEM[1] | Memory Allocation Error : size = **3000** bytes |
| ⑮ | 2\_**sync\_user1** | DBERROR**[-1:HY000]:[Oracle][ODBC][Ora]ORA-00923: FROM 키워드가 있어야할 곳에 없습니다** |

<표 4-1> 로그 분류

|  |  |
| --- | --- |
| 분류 | 해결책 |
| ① | mSync 서버를 띄울 때의 에러 메시지로 auth.ini 파일이 mSync 서버와 같은 폴더에 있는지, 제대로 된 파일인지, 설치가 허용된 서버인지, 이미 mSync 서버가 떠 있는지를 확인해 본다. |
| ② | mSync 서버의 종료를 나타내는 메시지 |
| ③ | mSync 서버가 실행됐을 때의 정상적인 로그 메시지 |
| ④ | 클라이언트가 연결되고 연결 해제 되었음을 알리는 정상적인 로그 메시지이며 클라이언트의 동기화 중에 에러가 발생한 경우 에러가 발생했음을 알린다.. |
| ⑤ | System DB는 mSync 서버를 띄울 때의 DSN 또는 SID이고 User DB는 application의 DSN 또는 SID 값이며 연결이 실패하였음을 나타내는 메시지로 DSN이나 SID의 값이 정확한지 확인한다. |
| ⑥ | Application에서 mSync\_AuthRequest()의 파라미터로 전달한 값 중 패스워드가 잘못되었거나 등록되지 않은 사용자거나 DB 에러가 났음을 나타내는 메시지로 패스워드나 사용자가 제대로 등록되었는지 확인하고 데이터베이스 서버의 상태를 확인한다. |
| ⑦ | 동기화 하려는 application의 업그레이드할 버전이 있다는 것을 알리는 정상적인 로그 메시지 |
| ⑧ | 동기화 하려는 application의 정보를 찾을 수 없으므로 해당 application이 제대로 등록되어 있는지 확인한다. |
| ⑨ | 최대 user수를 초과했을 때의 메시지 |
| ⑩ | 동기화의 진행 과정을 출력하는 정상적인 로그 메시지 |
| ⑪ | 클라이언트 접속 연결이 되었으나 이를 처리할 thread를 생성하지 못함. |
| ⑫ | Upload의 스크립트가 존재하지 않는다거나 download의 스크립트가 존재하지 않는다는 메시지로 upload/download를 수행하기 위해서는 해당 스크립트를 작성해줘야 한다. 또한, DB의 스키마에 지정되어 있는 사이즈보다 클라이언트로부터 받은 데이터의 사이즈가 크거나 그 포맷이 제대로 되어 있지 않은 경우, 또 파라미터의 개수가 다른 경우 로그 메시지를 확인하고 해당 사항을 확인한다. 클라이언트에서 정의한 DSN이 없다는 것은 application 등록 시에 지정한 client DSN 값이 application에서 mSync\_SendDSN()으로 제대로 전달하는지 확인한다. |
| ⑬ | 클라이언트와 mSync 서버 사이의 네트워크 에러 메시지로 application에서 규칙대로 API를 호출하는지 확인한다. |
| ⑭ | 메모리 할당 에러로 사이즈를 확인하고 담당자에 문의한다. |
| ⑮ | 데이터베이스 서버에 대한 에러로 스크립트 작성이 잘못 되었거나 데이터베이스에 대한 액세스가 실패할 경우 출력되며 해당 메시지는 데이터베이스 서버에서 얻어오므로 메시지를 보고 적절한 수정을 한다. |

<표 4-2> 해결책

## mSync clientAPI

### 개요

1.3 장에서 언급한 바와 같이, mSync 응용 시스템 중 mSync 클라이언트 쪽은 API set을 제공하며 여기서는 mSync 클라이언트 API에 대해서 기술한다. C/C++로 개발하는 경우에는 mSync 클라이언트 개발 시 mSyncclient.h를 include해야 한다.

JAVA로 개발 할 때는 mSyncClientJNI.java를 포함해 줘야 한다.

### API

“설명서\_openMSYNC\_JAVA\_API”를 참고하도록 한다.

1. 스크립트 작성 시 DBMS별 유의사항

4.1.2.5 장이나 4.1.3.3 장, 4.1.3.4 장, 4.1.3.5 장에서와 같이 스크립트를 작성할 때에는 다음과 같은 점들을 유의해야 한다.

다음은 서버의 데이터베이스에 따른 유의 사항을 설명한다.

* Oracle, MySQL
* Upload의 데이터에 timestamp field 값을 넣어 줄 경우 SYSDATE 함수를 사용
* Oracle은 datetime type사용불가
* MS Access
* script의 where 절에 datetime field(예를 들어 syncTime이라고 하자)를 사용할 경우 syncTime = ?라고 쓰는 대신 *syncTime = Cdate(‘?’)*로 써 준다.
* Upload의 데이터에 datetime field 값을 넣어 줄 경우 Now() 함수를 사용한다.
* **다른 DB들과 달리 script를 작성할 때에 data type을 고려하여 스트링 계열인 경우 ‘?’를 써 주고 숫자 계열인 경우 ?를 써 준다.**

|  |
| --- |
| 예) employee table이 id(int), name(char), date(datetime)으로 구성되어 있을 경우,   * + 1. download script 작성시   다른 DB(MS-SQL) : select id, name from employee where date >= ? 로 작성  ACCESS DB : select id, name from employee where date >= Cdate(‘?’)로 작성   * + 1. upload\_insert script 작성시   다른 DB(Oracle) : insert into employee values(?,**?**, SYSDATE)로 작성  ACCESS DB : insert into employee values(?, **‘?’**, Now())로 작성 |

* MS-SQL Server, Sybase
* Upload의 데이터에 datetime field 값을 넣어 줄 경우 getdate() 함수를 사용
* Timestamp type사용 불가
* CUBRID
* Upload의 데이터에 datetime 또는 timestamp field 값을 넣어 줄 경우 SYSDATE 함수를 사용
* **Accss와 마찬가지로 script를 작성할 때에 data type을 고려하여 스트링 계열인 경우 ‘?’를 써 주고 숫자 계열인 경우 ?를 써 준다. Nchar/nvarchar type인 경우 n’?’를 써준다.**

|  |
| --- |
| 예) employee table이 id(int), name(char), address(nchar), date(timestamp)으로 구성되어 있을 경우,   * upload\_insert script 작성시   다른 DB(Oracle) : insert into employee values(?,**?**, ?, SYSDATE)로 작성  CUBRID DB : insert into employee values(?, **‘?’, n’?’**, SYSDATE)로 작성 |

* **ODBC 함수 관련하여 잘못된 결과를 return 하므로 이를 수정하기 위해 broker 설정을 다음과 같이 변경한다.**
  + - **$CUBRID/conf/cubrid\_broker.conf에 'cci\_default\_autocommit=off'로 설정**
    - **cubrid broker restart.**
* DB2
* Upload의 데이터에 datetime 또는 timestamp field 값을 넣어 줄 경우 CURRENT\_DATE 함수를 사용

.

1. 제한 사항

현재 mSync 클라이언트와 mSync 서버 사이에 주고 받는 데이터의 버퍼 크기가 8156byte으로 정해져있다. 만일, 동기화 할 테이블의 한 레코드 사이즈가 8156byte보다 클 경우 HUGE\_STRING\_LENGTH를 조정하여 mSync 제품을 다시 생성해야 한다.

Binary 형태의 필드에 대해서는 동기화 지원을 하지 않는다. 따라서 BLOB 등의 type은 지원되지 않음에 유의해야 한다.