

목 차

4. 데이터 소스.....	4
4.1. 데이터베이스 접속.....	4
4.1.1. ODBC 접속.....	4
4.1.2. 데이터베이스 직접 접속.....	5
4.1.3. ADO 접속.....	6
4.1.4. RDS 접속.....	7
4.1.5. Data Server 접속.....	8
4.1.6. NET 접속.....	8
4.1.7. SAP 접속.....	9
4.1.8. XMLAGENT 접속.....	10
4.1.9. FILEAGENT 접속.....	11
4.1.10. SAPAGENT 접속.....	11
4.1.11. 접속 확인.....	11
4.2. 쿼리 정의.....	12
4.2.1. 쿼리 종류 선택.....	13
4.2.2. SQL 자유편집.....	14
4.2.3. 다중 DB 사용.....	16
4.2.4. 저장 프로시저 사용.....	17
4.2.5. 결과 레코드 셋.....	18
4.2.6. EJB 함수 사용.....	18
4.2.7. Prepared Statement 사용.....	19
4.2.8. 쿼리를 실행하여 컬럼명 가져오기.....	19
4.2.9. 쿼리결과 재사용.....	19
4.2.10. 서브 리포트 사용.....	20
4.2.11. 쿼리 작성 탭 이용.....	21
4.2.12. SAP 쿼리 정의.....	25
4.2.13. FILEAGENT 쿼리 정의.....	28
4.2.14. 웹 서비스 설정.....	29
4.2.15. iBatis, plugin 연동 보고서 쿼리 정의.....	31

4.3. 필드 쿼리 정의.....	31
4.4. XML 스키마 지정	33
4.4.1. 스키마 종류 지정	33
4.4.2. 스키마 파일 지정	34
4.4.3. 루트요소 지정	36
4.4.4. XML 구조를 문서에 포함	36
4.4.5. XMLAGENT 스키마 지정	36
4.5. XML 쿼리 정의	38
4.5.1. SQL 쿼리 정의와 XML 쿼리 정의 비교	38
4.5.2. 쿼리이름 과 최상위 요소	39
4.5.3. 필드 정의	39
4.5.4. 조건정의	41
4.5.5. 정렬/기타 정의	42
4.5.6. XSLT 보기.....	44
4.5.7. 결과 레코드 보기	45
4.5.8. 생성된 XSLT 살펴보기	45
4.5.9. 웹서비스 연동 보고서인 경우	47
4.6. 파일 접속.....	48
4.6.1. 파일 접속 탭	48
4.6.2. 필드 정의 탭	50
4.6.3. 가상 필드 정의 탭	53
4.6.4. XML 파일 접속 대화상자 이용하기.....	54
4.7. Multi-Pass SQL 필드 정의	55
4.7.1. 키 필드 정의 탭	55
4.7.2. 필드 정의 탭	56
4.8. 사용자 정의 함수.....	58
4.9. 서브 페이지 데이터 설정	58
4.10. 전처리/후처리 지정	59
4.11. 데이터 셋 연결	60
4.11.1. 필드 지정 탭	62
4.11.2. 참조 필드 탭	63
4.11.3. 요약 작성 탭	64

4.11.4. 요약 조건 탭	66
4.12. 스크립트	68
4.12.1. 변수명	69
4.12.2. 변수형	69
4.12.3. 변수 리스트	69
4.12.4. 스크립트 편집 창	70
4.12.5. 스크립트 설명	70
4.12.6. 스크립트 참조	70
4.12.7. 버튼들의 기능	71
4.12.8. 스크립트 편집 창의 팝업메뉴.....	72
4.13. XPlatform DataSet 접속	73
4.13.1. XML 스키마 지정.....	73
4.13.2. 쿼리 정의	74
4.13.3. 미리 보기기	75
4.13.4. XPlatform Runtime 과 데이터 연동 (보고서 실행 타임).....	76

4. 데이터 소스

이 장에서는 데이터 소스 메뉴에 포함된 기능에 대한 설명들을 수록했습니다. 데이터 소스 메뉴에서는 데이터베이스 접속 등의 데이터 소스와의 접속 방법 및 데이터 셋 연결 등 전반적인 데이터 소스에 대한 여러 가지 기능들을 제공하고 있습니다.


4.1. 데이터베이스 접속

[데이터 소스]-[데이터베이스 접속]을 선택하거나 단축키 <Ctrl+T>를 누르면 데이터베이스 접속 대화상자가 나타납니다.

Designer 에서 데이터베이스에 접속할 수 있는 방법은 **ODBC 접속**과 **데이터베이스 직접 접속**, **ADO 접속**, **RDS 접속**, **Data Server 접속**, **NET 접속**, **SAP 접속**, **XMLAGENT 접속**, **FILEAGENT 접속**, **SAPAGENT 접속** 등이 있습니다. 문서에 지정한 데이터베이스 접속 방법에 따라 데이터베이스 접속 대화상자 모양은 각기 다르지만, 데이터베이스 접속 이후의 과정은 대부분 동일합니다.

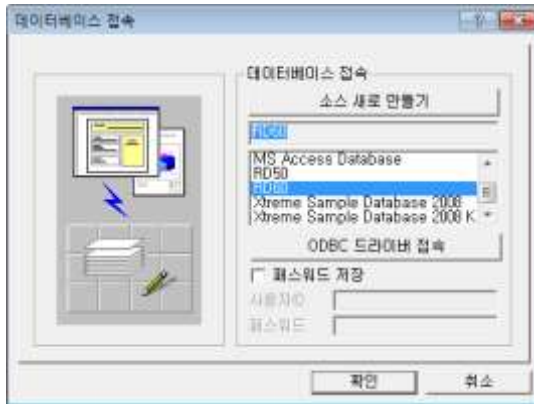
데이터베이스 에 접속된 상태에서는 [데이터 소스]-[데이터베이스 접속]메뉴가 [데이터베이스 접속 해제]로 변경되며, [데이터 소스]-[데이터베이스 접속 해제]를 선택하면 데이터베이스 접속이 해제됩니다. 데이터베이스 접속이 정상적으로 해제된 경우에는 아래의 그림과 같은 메시지가 나타납니다.




 단, 유니코드 버전에서는, 데이터베이스 접속에서 **ODBC 접속**과 **데이터베이스 직접 접속**, **RDS 접속** 기능은 제공하지 않습니다.

4.1.1. ODBC 접속

ODBC 접속의 경우엔 현재 등록되어 있는 데이터베이스(**DSN**) 목록을 보여주는 데이터베이스 접속 대화상자가 나타납니다. 데이터베이스에 접속을 하려면 데이터베이스 소스 명을 두 번 누르거나 데이터베이스 소스 명을 선택한 후, **ODBC 드라이버 접속** 버튼을 누릅니다.

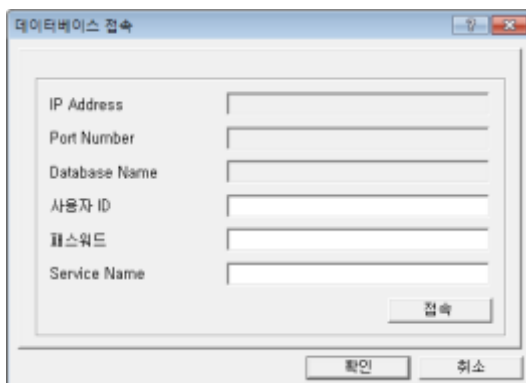



 **ODBC Data Source Manager(ODBC 데이터 원본 관리자)** 란 데이터베이스의 소스 관리를 사용자가 쉽게 하기 위해서 만들어진 프로그램으로 시스템에 ODBC Driver 를 설치하고 난 후, ODBC Data Source 를 설정할 수 있습니다. 이는 Microsoft Windows 제어판의 32bit-ODBC 를 실행하여도 같은 대화상자가 나타납니다.

4.1.2. 데이터베이스 직접 접속

데이터베이스 직접 접속은 **Database Network Library** 를 이용하여 해당 데이터베이스에 직접 접속하는 방법입니다. 사용하는 데이터베이스(**ORACLE, SYBASE, SRP, UNISQL, MI** 등)에 직접 접속하기 위해 **IP Address, Port Number, Database Name, 사용자 ID, 패스워드, Service Name** 을 입력할 수 있는 데이터베이스 접속 대화상자를 사용합니다.

아래의 그림은 데이터베이스 접속방법을 **Oracle8** 로 선택한 후, **[데이터 소스]-[데이터베이스 접속]**을 실행한 경우의 데이터베이스 접속 대화상자입니다. 사용자 ID, 패스워드, Service Name 을 입력하고 접속 버튼을 누릅니다.

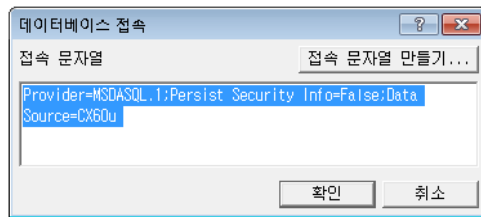


 데이터베이스 직접접속을 할 경우, 주의해야 하실 점은 **접속하고자 하는 데이터베이스의 윈도우용 클라이언트 소프트웨어를 먼저 설치**하셔야 한다는 점입니다. 보통 윈도우용 클라이언트 소프트웨어를 설치하면

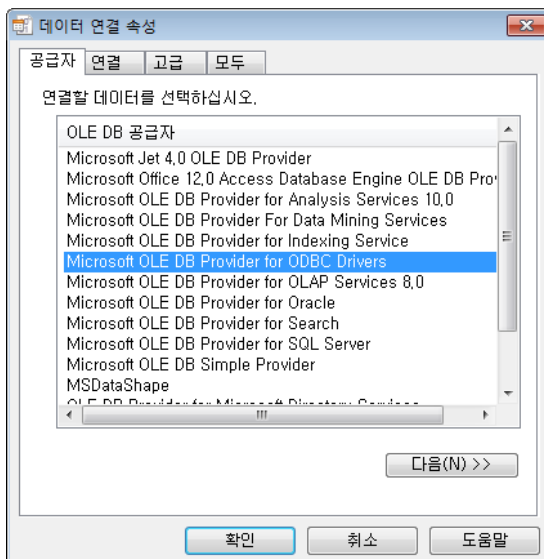
Database Network Library 가 설치됩니다. 또한 해당 클라이언트 소프트웨어를 실행하여 데이터베이스 접속을 위한 서비스명 등을 먼저 만들어야 합니다.

4.1.3. ADO 접속

ADO 접속은 OLE DB Provider 를 통해 데이터를 가져오기 위한 접속방법입니다.



ADO 개체를 통한 OLE DB 접속을 하기 위해서는 데이터베이스 접속을 위한 접속문자열 정보가 필요합니다. 접속 문자열은 사용자가 직접 입력하거나 **접속 문자열 만들기** 버튼을 클릭하여 아래의 그림과 같은 **데이터 연결 속성(Data Link Properties)** 대화상자를 띄워서 작성할 수 있습니다.



□ 공급자(Provider)

OLE DB 공급자 리스트가 나열되면 그 중 하나를 선택합니다.

□ 연결(Connection)


연결하고자 하는 데이터베이스 이름, 데이터베이스에 로그인 하기 위한 사용자 이름, 패스워드 등을 입력합니다.

□ 고급(Advanced)

네트워크 설정 및 액세스 권한 설정 등을 입력합니다.

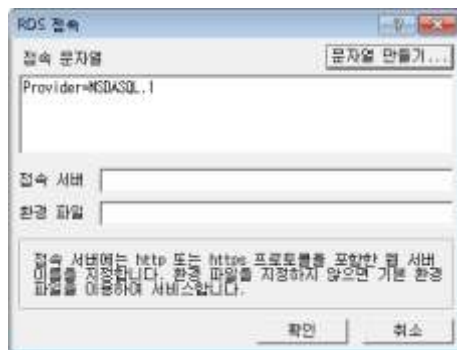
□ 모두(All)

기타 초기화 속성들을 입력합니다.

 ADO(ActiveX Data Object)란 MS사의 새로운 데이터 접속 개체로서 대부분의 데이터 형태를 접속해서 처리할 수 있도록 지원합니다. Crownix Report는 ADO 개체를 통한 OLE DB 접속을 지원함으로써 비정형 데이터를 포함한 다양한 데이터 소스에 접근이 가능합니다.

4.1.4. RDS 접속

RDS 접속은 RDS(Remote Data Service)를 통해 데이터를 가져오기 위한 접속 방법입니다.



□ 접속 문자열


데이터베이스 연결을 위한 연결문자열 정보를 입력합니다.

□ 접속 서버

데이터를 서비스할 서버 이름을 입력합니다. 프로토콜(http/https)을 포함한 서버 이름을 입력하고 입력하지 않으면 로컬에 있는 서비스를 이용합니다.

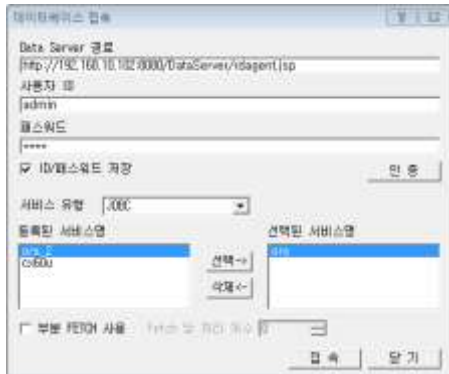
□ 환경 파일

서버에 설치되어 있는 환경 파일 이름을 입력합니다. 입력하지 않으면 기본 환경 파일을 이용하여 서비스합니다.

 RDS(Remote Data Service)란 웹 환경에서 데이터베이스를 조작하기 위한 MS사의 기술입니다. ADO는 데이터 소스에 바로 연결하여 작업(two-tier)하는 반면 RDS는 IIS 웹 서버를 거쳐서 데이터 소스에 연결합니다.

4.1.5. Data Server 접속

Data Server 접속은 **Crownix ERS Data Server** 를 통해 데이터를 가져오는 접속 방법입니다. **Crownix ERS Data Server** 사용에 대한 자세한 사항은 서버 모듈과 함께 제공되는 매뉴얼을 참고 하십시오.



Data Server 경로에 **Crownix ERS Data Server** 의 웹 경로를 입력합니다. 런타임 시 **Data Server** 경로 설정은 파라미터 옵션 중 **/rf** 나 **/rfn** 을 사용합니다.



/rf 나, **/rfn** 파라미터에 대한 자세한 설명은, **부록 3 Crownix Report 파라미터** 에서 **3.1.80. /rf** 나, **3.1.89. /rfn** 의 내용을 참조하십시오.

Data Server 경로를 입력하고 사용자 **ID**, **패스워드**를 입력한 후, **인증** 버튼을 누릅니다. 인증에 성공하면 **Data Server** 에 등록되어 있는 서비스 유형, 서비스 명들이 나열됩니다. 접속 하려는 서비스 유형, 서비스 명을 선택하고 접속을 누릅니다.

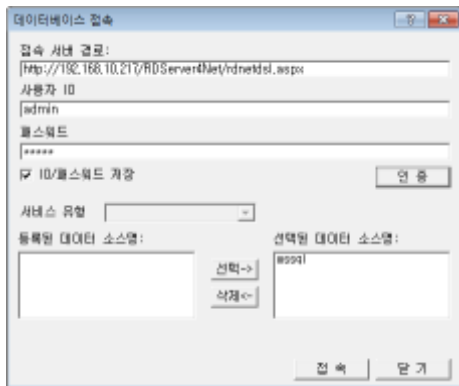
부분 FETCH 사용이란, 한 번에 대량의 데이터를 가져와야 하는 경우에 네트워크 환경에 따라 데이터 패치가 불가능해지는 것을 막기 위해 **FETCH** 당 처리개수에서 지정한 개수만큼 쿼리 결과를 나눠서 가져오는 기능입니다. 단, 이 기능은 **Oracle DB** 를 사용하는 경우에 한해 제공됩니다.




Data Server 에 등록 되어 있는 서비스 유형 목록이 표시 됩니다. 이 문서에서 서비스 유형을 별도 언급 하지 않고 **Data Server** 접속방식 으로 설명 되어 있는 경우, 서비스 유형은 **JDBC** 가 됩니다.

4.1.6. NET 접속

NET 접속은 **Crownix ERS Data Server for .NET** 를 통해 데이터를 가져오는 접속 방법입니다. **Crownix ERS Data Server for .NET** 사용에 대한 자세한 사항은 서버 모듈과 함께 제공되는 매뉴얼을 참고하십시오.



접속 서버 경로에 **Crownix ERS Data Server for .NET** 의 웹 경로를 입력합니다. 접속 서버 경로를 런타임 시에 설정하고 싶은 경우에는 **Data Server 접속**에서와 마찬가지로 파라미터 옵션 중 **/rf** 나 **/rfn** 을 사용합니다.

 **/rf** 나 **/rfn** 파라미터에 대한 자세한 설명은, **부록 3 Crownix Report 파라미터에서 3.1.80. /rf** 나 **3.1.89. /rfn** 의 내용을 참조하십시오.

접속 서버 경로를 입력하고 사용자 ID, 패스워드를 입력한 후, 인증 버튼을 누릅니다. 인증에 성공하면 서버에 등록되어 있는 데이터소스 명들이 **등록된 데이터 소스명** 창에 나열되며 그 중 접속을 원하는 데이터소스 명들을 선택한 후, 확인을 누르면 접속이 이루어집니다.

4.1.7. SAP 접속

SAP 접속은 **SAP R/3 함수**를 호출한 결과 셋을 데이터로 이용하기 위한 접속 방법으로 **SAP** 접속 방식 보고서 양식 파일을 작성하기 위해서는 **SAPGUI** 가 설치되어 있어야 합니다.





□ 시스템 내역 사용

System - SAP Logon 에 설정한 시스템 내역 입니다.

Client - SAP R/3 에 접속을 위한 Client 정보입니다.

UserId - SAP R/3 에 접속을 위한 사용자 ID 입니다.

Password - SAP R/3 에 접속을 위한 사용자 패스워드 입니다.

Language - SAP R/3 에 접속을 위한 언어 설정 입니다.

□ Connection Parameters 사용

SAP R/3 에 접속하기 위해 Connection Parameters 를 사용합니다.

□ 보고서 작성시 Data Server 사용

보고서 실행 시 **Crownix ERS Data Server (JAVA/.NET)** 를 통해 SAP R/3 시스템에 접속하고 데이터를 가져오고 싶은 경우에는 보고서 작성시 **Data Server** 사용을 체크한 후, **Data Server** 경로와 서비스 명을 입력합니다.



단, SAP 접속은 Crownix Report 6.0 for SAP 버전에서만 사용이 가능합니다.

4.1.8. XMLAGENT 접속

XMLAGENT 접속은 **Crownix ERS Data Server** 를 통해 XML 데이터를 가져오는 접속 방법입니다. **Designer** 에서는 보고서 작성시 필요한 XML 데이터 정보를 서버를 통해 얻어오고, 실제 **Viewer** 에서는 서버 접속 없이 XML 데이터를 통해 리포팅을 수행합니다.

접속 방법은 **Data Server** 접속과 동일하며, 다만, 부분 **FETCH** 사용, **FETCH** 당 처리 개수 등

의 기능은 제공되지 않습니다. (**AGENT** 선택 옵션을 **NET-LW** 로 선택시엔 **NET** 접속과 동일.)

XMLAGENT 접속 선택시, 함께 선택해야 하는 **AGENT** 선택 옵션은 총 4 가지로, **Data Server-LW, NET-LW, Data Server-PW, Data Server-AF** 가 있으며, 옵션명에 뒤에 붙는 글자는 XML 데이터 생성을 위해 서버와 연동된 외부 모듈을 의미합니다. **LW** 는 **LiveWorks**, **PW** 는 **ProWorks**, **AF** 는 **AnyFrame** 입니다.



Data Server-PW 나 **Data Server-AF** 사용시 **Viewer** 에서 XML 데이터를 가져오기위해 사용하는 파라미터는 **/rserverxmlcparam** 나 **/rserverxmliparam** 파라미터로, 각 파라미터에 대한 자세한 설명은, **부록 3 Crownix Report 파라미터**에서 **3.1.253. /rserverxmlcparam** 나 **3.1.254. /rserverxmliparam** 의 내용을 참조하십시오.

4.1.9. FILEAGENT 접속

FILEAGENT 접속은 **Crownix ERS Data Server** 를 통해 FILE 데이터를 가져오는 접속 방법입니다. **Designer** 에서는 보고서 작성시 필요한 FILE 데이터 정보를 서버를 통해 얻어오고, 실제 **Viewer** 에서는 서버 접속 없이 FILE 데이터를 통해 리포팅을 수행합니다.

접속 방법은 **Data Server** 접속과 동일하며, 다만, 부분 **FETCH** 사용, **FETCH** 당 처리 개수 등의 기능은 제공되지 않습니다.

FILEAGENT 접속 선택시, 함께 선택해야 하는 **연결 옵션** 선택 옵션은 현재 **TR** 한 가지만 선택이 가능합니다. **TR** 옵션은 서버를 통해 Transation 단위의 데이터들을 조합하여 필요한 데이터를 조합하고자 할 때 사용합니다.

4.1.10. SAPAGENT 접속

SAPAGENT 접속은 **Crownix ERS Data Server** 를 통해 **SAP R/3** 함수를 호출한 결과 셋을 가져오는 접속 방법입니다. **Designer** 에서는 보고서 작성시 필요한 데이터 정보를 서버를 통해 얻어오고, 실제 **Viewer** 에서는 서버 접속 없이 **SAP R/3** 함수를 호출한 결과 셋을 통해 리포팅을 수행합니다.

접속 방법은 **AGENT** 선택 옵션을 **Data Server** 로 선택한 경우는 **Data Server** 접속과 동일하며, **AGENT** 선택 옵션을 **NET** 으로 선택한 경우는 **NET** 접속과 동일합니다. 다만, 부분 **FETCH** 사용, **FETCH** 당 처리 개수 등의 기능은 제공되지 않습니다.



단, **SAPAGENT** 접속은 **Crownix Report 6.0 for SAP** 버전에서만 사용이 가능합니다.

4.1.11. 접속 확인

모든 설정을 하고 나서 접속에 성공하면 **접속 성공 메시지**가 나타납니다. 데이터베이스 접속 중에 오류가 발생하면 해당 오류 메시지가 나타나고 이어서 **접속 실패 메시지**가 나타납니다. 이 경우엔 설정이 정상적으로 되었는지 다시 한 번 확인해주시기 바랍니다.

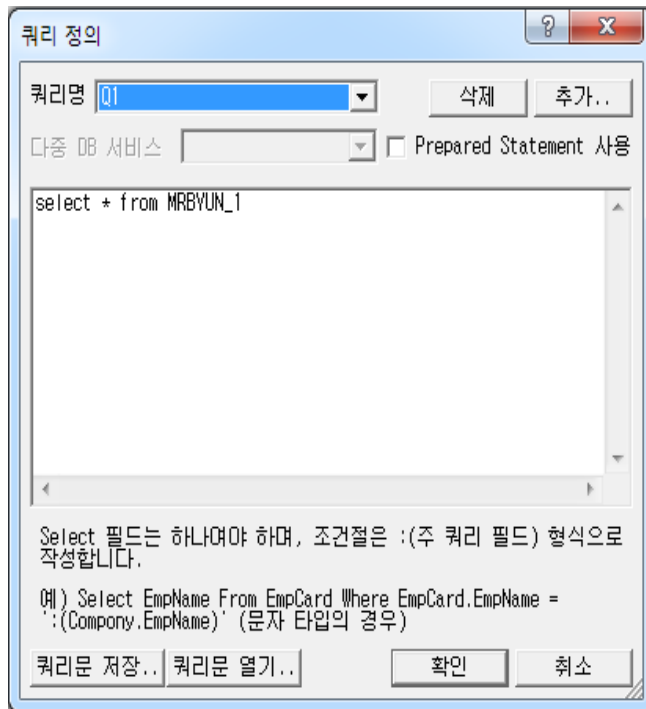



4.2. 쿼리 정의

데이터베이스 접속에 성공한 경우에 한해 SQL 문을 작성할 수 있습니다. [데이터 소스]-[쿼리 정의]를 선택하거나, 단축키 <Ctrl+Q>를 누르면 아래의 그림과 같은 쿼리 정의 대화상자가 나타납니다.



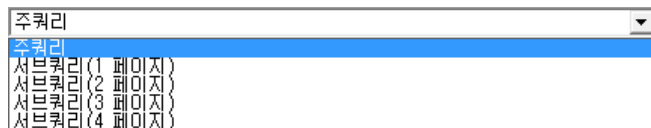
단, Data Server 접속 방식이고 Multi-Pass SQL 사용을 하는 경우에는 아래의 그림과 같은 쿼리 정의 대화상자가 나타납니다.




 Multi-Pass SQL 사용 쿼리 정의에 대한 자세한 설명은 4.3 필드쿼리 정의의 내용을 참조하시기 바랍니다.

4.2.1. 쿼리 종류 선택

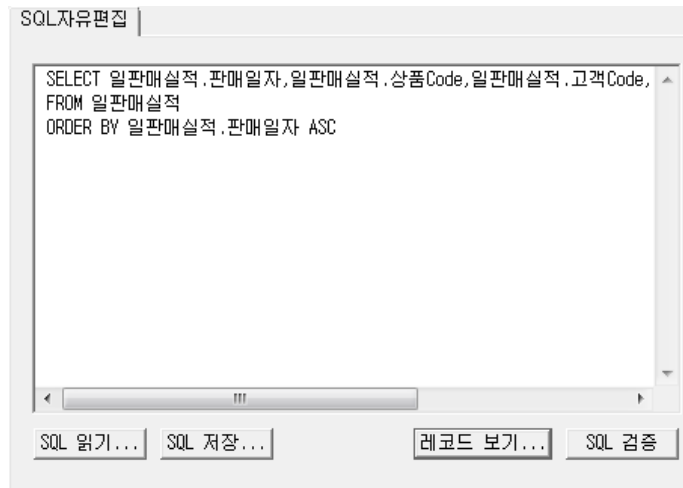
SQL 문을 작성하기 위해서는 먼저 쿼리 정의 대화상자의 상단에서 작성하고자 하는 쿼리의 종류를 선택합니다. 서브 페이지 문서가 아니라면 **주 쿼리**만을 작성할 수 있으며 서브 페이지 문서인 경우엔 **주 쿼리**와 **각 페이지의 서브 쿼리** 중에 작성하고자 하는 쿼리를 선택하여 작성할 수 있습니다.



 서브 페이지 문서에서 서브 쿼리는 페이지 수만큼 생성됩니다. 두 개 이상의 서브 쿼리를 작성하고자 하실 때는 [삽입]-[페이지]-[본문 추가/삽입]을 선택하여 원하는 서브 쿼리 개수만큼 본문 페이지를 추가하십시오.

4.2.2. SQL 자유편집

SQL 자유편집 체크 박스를 선택하면 쿼리 정의 대화상자의 하단 부분이 아래의 그림처럼 쿼리를 직접 입력할 수 있는 상태로 바뀝니다. 또한 **SQL** 자유편집을 통해 작성하는 **SQL** 문은 작성 중인 문서에서 사용하는 데이터베이스의 **SQL** 문법에만 맞는다면 모두 허용됩니다.



4.2.2.1. SQL 읽기/저장

SQL 읽기는 저장된 **SQL** 파일을 불러오는 기능이며 **SQL 저장**은 작성된 **SQL** 문을 외부 파일로 저장할 수 있는 기능입니다.

4.2.2.2. 레코드 보기

레코드 보기 버튼은 현재 작성된 쿼리로 가져올 수 있는 레코드를 보여주며 **외부 파일로 저장** 버튼을 통해 해당 레코드 데이터를 **csv** 파일이나 텍스트 파일로 저장할 수 있습니다.



단, 레코드 보기에 표시되는 결과 레코드 건수는 **최대 10,000 건**으로 제한하고 있으며 쿼리문

에 이미지 데이터나 메모데이터와 같은 **Blob 데이터(Binary large object Data)**를 가져오는 경우에는 사용할 수 없습니다.

또한, 현재 작성된 쿼리문장에 \$1, \$2 와 같이 보고서 실행 시에 전달받는 파라미터 변수를 사용한 경우에도 정상적으로 쿼리문이 수행될 수 없어서 레코드 보기 기능을 사용할 수 없습니다.

하지만, **[데이터 소스]-[파라미터 값입력]** 대화상자에 파라미터값을 입력하면 레코드 보기 실행 시 자동으로 해당 파라미터값으로 자동 치환된 쿼리문장을 수행하여 결과 레코드를 조회해 보실 수 있습니다.

4.2.2.3. SQL 검증

SQL 검증은 작성된 쿼리가 SQL 문법에 맞는지를 확인해줍니다. 단, SQL 검증 기능은 해당 데이터베이스의 SQL 문법을 기준으로 검증이 되므로 **쿼리 동적 생성 기능**을 사용하는 경우에는 SQL 검증이 되지 않습니다.

4.2.2.4. 쿼리 동적 생성

SQL 자유편집을 사용하는 경우, 쿼리의 특정 행이 파라미터 조건에 따라 빠지거나 더해지도록 설정하여 쿼리를 동적으로 생성해 줄 수 있습니다.

즉, 파라미터 옵션 중, **/rp** 를 사용하여 \$1, \$2 등과 같은 변수 값의 유무에 따라 쿼리 특정 행의 포함여부를 결정할 수 있다는 것입니다.



/rp 파라미터에 대한 자세한 설명은, **부록 3 Crownix Report 파라미터**에서 **3.1.159. /rp**의 내용을 참조하십시오.

파라미터의 조건에 따라 포함유무가 결정되는 행으로 지정하기 위해서는 행의 첫 문자는 반드시 **#** 또는 **^** 이 되어 합니다. 물론, 공백이 있어서도 안됩니다. **#** 과 **#** 으로 묶인 조건구문은 **and** 조건으로 묶이게 되며 **^** 과 **^** 으로 묶인 조건구문은 **or** 조건으로 묶이게 됩니다.

조건구문이란, 파라미터의 순서와 입력유무에 따라 조건을 표현하는 것으로 **1, 0, X** 세 가지를 사용하여 표현할 수 있습니다. **1** 은 해당 순서의 파라미터 값이 입력되었음을 나타냅니다. **0** 은 해당 순서의 파라미터 값이 입력되지 않았음을 나타냅니다. **X** 는 해당 순서의 파라미터 값은 입력 유무에 상관없음을 나타냅니다. 따라서, **#1X0#** 라는 조건구문은, '첫 번째 파라미터 값이 입력되고 세 번째 파라미터 값은 입력되지 않았을 때' 라는 조건이 되는 것입니다.

예를 들어, 특정 문서에서 최대값과 최소값에 대한 파라미터 값을 입력 받아 사용자가 지정한 최대값과 최소값의 사이에 있는 데이터들만 표시해야 한다고 가정해 봅시다. 이때, 사용자가 최대값만 입력했다면 최대값보다 작은 값들을 전부 표시해야 하고 최소값만 입력했다면 최소값보다 큰 값들을 전부 표시해야 하며 아무 값도 입력하지 않았다면 모든 데이터들을 전부 표시해야 하는 쿼리를 어떻게 작성해야 할까요? 이 경우, **쿼리 동적 생성 기능**을 사용하면 아래의 그림과 같이 간단하게 쿼리를 작성할 수 있습니다.

```
SELECT person.name, person.age
FROM person
#10# WHERE person.age >= $1
#01# WHERE person.age <= $2
#11# WHERE person.age >= $1 && person.age <= $2
```

위의 쿼리에서 파라미터 값이 '/rp [10] []' 과 같이 첫 번째 파라미터 값만 입력되었다면 실제로 생성되는 쿼리는 아래와 같이 생성됩니다.

```
SELECT person.name, person.age
FROM person
WHERE person.age >= 10
```

또는, 파라미터 값이 '/rp [10] [40]' 과 같이 첫 번째와 두 번째 파라미터 값이 모두 입력되었다면 실제로 생성되는 쿼리는 아래와 같이 생성됩니다.

```
SELECT person.name, person.age
FROM person
WHERE person.age >= 10 && person.age <= 40
```



참고로 파라미터 값은 문자열의 길이가 0 인 것도 파라미터 값이 없는 것으로 간주합니다.

또한, #...# 과 ^...^ 의 묶음을 같은 행에 조합하여 사용할 수도 있습니다. 이 경우에는 &(and), |(or) 와 같은 기호를 사용합니다. 예를 들어, #...#&^...^ 은 #...# 과 ^...^ 의 묶음을 and 로 처리한다는 것이고 #...#|^...^ 은 #...# 과 ^...^ 의 묶음을 or 로 처리한다는 뜻입니다. 물론, ^...^&#...#|^...^ 등의 조합도 사용 가능합니다.

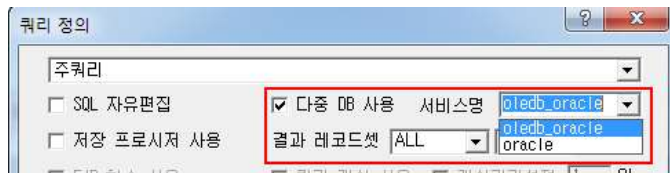
4.2.2.5. 주 쿼리 필드명 사용

서브 쿼리 정의에 주 쿼리의 필드명이나 파라미터 변수를 사용하여 동적 쿼리문을 작성할 수 있습니다. 쿼리문 안에서 주 쿼리의 필드명이나 파라미터 변수는 :(Customers.Name) 또는 :(\$1) 과 같이 :(...) 형태로 작성합니다. 만일 주 쿼리 필드의 데이터 형이 문자열 경우에는 :(Customers.Name)' 와 같이 사용합니다.

4.2.3. 다중 DB 사용

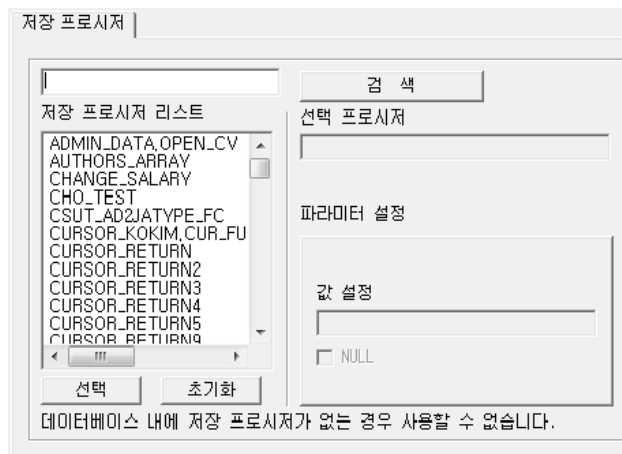
다중 DB 사용 기능이란 한 문서 안에서 여러 개의 서비스(Crownix ERS Data Server 에서 설정해 놓은 데이터베이스 서비스)에 접속하여 데이터를 가져올 수 있는 기능입니다. 데이터베이스 접속 방법 중 Data Server 접속 및 NET 접속 인 경우에만 다중 데이터베이스 기능을 사용할 수 있으며 이 기능을 사용하면 한 문서 안에서 전혀 다른 데이터베이스에 접속하여 데이터를 가져오는 것이 가능해집니다.

다중 데이터베이스 사용 기능을 사용하기 위해서는 데이터베이스 접속 대화상자에서 여러 개의 데이터베이스 서비스들을 선택한 후, 접속해야 합니다. 그런 다음 쿼리 정의 대화상자에서 다중 DB 사용 을 체크하고 서비스명 리스트에서 사용하고자 하는 데이터베이스 서비스명을 선택하여 사용합니다.



4.2.4. 저장 프로시저 사용


데이터베이스의 **저장 프로시저(Stored Procedure)**를 호출하여 데이터를 가져오기 위해 사용하며 호출할 저장 프로시저 이름과 넘겨줄 파라미터값을 정의합니다. **저장 프로시저 사용**을 체크하면 쿼리 정의 대화상자의 하단 부분이 아래의 그림처럼 변경됩니다.



저장 프로시저 목록상자에는 현재 접속한 데이터베이스에 있는 저장 프로시저 목록이 표시됩니다. 사용하고자 하는 저장 프로시저를 선택하려면 목록에서 마우스 왼쪽 버튼으로 더블 클릭하거나 목록상자 하단의 선택 버튼을 누르면 됩니다.

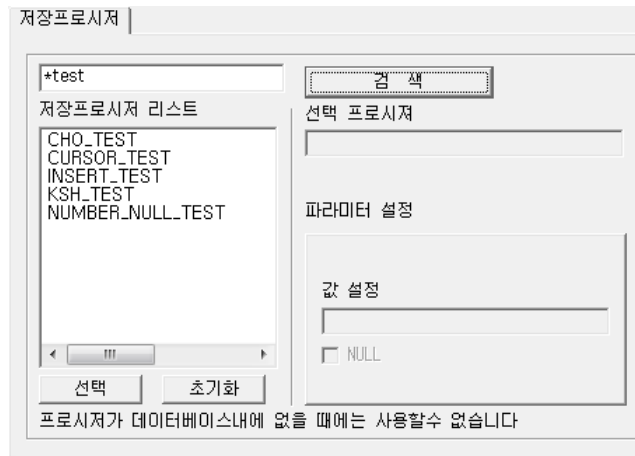
선택된 저장 프로시저에 저장 프로시저로 넘겨주는 파라미터가 있는 경우에는 파라미터 설정에서 파라미터를 선택한 후, 값 설정란에 넘겨줄 값을 입력하면 됩니다. 넘겨주는 값이 Null인 경우에는 하단의 **NULL** 체크 박스를 선택하면 됩니다.

파라미터 설정에 파라미터 옵션 중, **/rp** 나 **/rv** 를 사용하여 보고서 실행 시 파라미터 값을 넘겨줄 수도 있습니다.

 **/rp** 나 **/rv** 파라미터에 대한 자세한 설명은, **부록 3 Crownix Report 파라미터에서 3.1.159. /rp** 나 **3.1.317. /rv** 의 내용을 참조하십시오.

□ 검색

저장 프로시저 리스트 중에서 원하는 저장 프로시저를 이름으로 검색할 수 있습니다. 검색 시 대소문자 구분은 하지 않으며 부분 문자열을 통한 검색도 가능합니다. 또한, * 문자를 사용하여 특정 문자열로 시작하거나 끝나는 저장 프로시저 이름을 검색할 수도 있습니다.

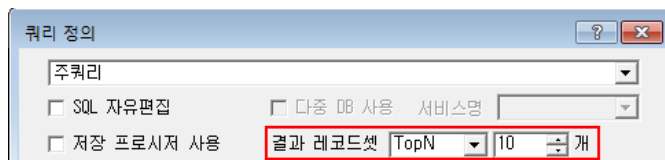


□ 초기화

선택되어 있던 저장 프로시저의 선택이 해제되고 파라미터 값 입력 부분도 초기화 됩니다.

4.2.5. 결과 레코드 셋

결과 레코드 셋 기능을 사용하면 쿼리를 통해 얻어진 결과 레코드를 위(**TopN**)나 아래(**BottomN**) 기준으로 지정한 개수만큼만 가져올 수 있습니다.



예를 들어, 위의 그림에서처럼 결과 레코드 셋을 TopN, 10 개로 지정하게 되면 쿼리를 통해 얻어진 결과 레코드 중, 상위 10 건만 가져오게 됩니다.

4.2.6. EJB 함수 사용

EJB 함수 사용은 데이터베이스 접속 방법 중 **Data Server** 접속 인 경우에만 사용할 수 있습니다. **EJB 함수 사용** 을 체크하면 쿼리 정의 대화상자의 하단 부분이 아래의 그림처럼 변경됩니다.



EJB 리스트 탭에서는 현재 접속한 데이터베이스에 존재하는 EJB 리스트를 보여주며 사용하고자 하는 EJB를 선택하여 마우스 버튼으로 더블 클릭하거나 추가 버튼을 누르면 선택 EJB 리스트에 추가됩니다. **EJB 메서드 리스트 탭**에서는 EJB 리스트 탭에서 선택한 EJB에 해당되는 메서드들이 나열되고 사용하고자 하는 메서드를 선택하면 됩니다.

선택된 메서드는 선택 메서드란에 보여지며 메서드로 넘겨주는 파라미터가 있는 경우에는 파라미터 설정에서 파라미터를 선택한 후, 값 설정란에 넘겨줄 값을 입력하면 됩니다. 또한, EJB 메서드 리스트 탭에 있는 여러 버튼의 이용방법은 저장 프로시저 탭과 유사하므로 **4.2.4 저장 프로시저 사용**의 내용을 참조하십시오.

4.2.7. Prepared Statement 사용

Prepared Statement 사용은 데이터베이스 접속 방법 중 **Data Server** 접속이면서 **Oracle DB**를 사용하는 경우에만 사용할 수 있습니다.

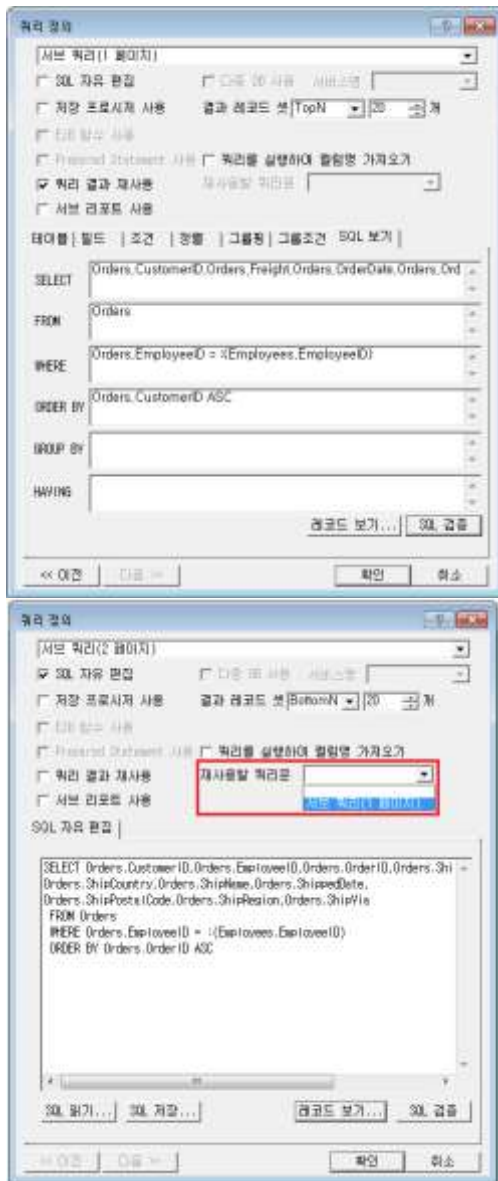
Prepare Statement를 사용하면 쿼리문에 대한 파싱을 처음 한 번만 실행하게 되어 보고서 작성을 위한 쿼리문 실행 속도가 향상됩니다.


4.2.8. 쿼리를 실행하여 컬럼명 가져오기

Select 절에 주석을 기입하였거나 **as**를 사용하지 않고 별칭(Aliasing)을 선언했을 경우에는 **[데이터 소스]-[데이터 셋 연결]** 대화상자의 **필드 리스트**를 정상적으로 가져올 수 없습니다. 이때는, 쿼리를 실행하여 **컬럼명 가져오기**를 사용하여 필드 리스트를 직접 가져오도록 합니다.

4.2.9. 쿼리결과 재사용

서브 페이지 보고서에서 특정 서브 페이지의 쿼리 결과데이터를 다른 서브 페이지에서 재사용하고 싶은 경우 사용하는 기능입니다. 현재 지정된 서브 페이지의 쿼리 결과를 다른 서브 페이지에서 재사용하고자 하는 경우, **'쿼리 결과 재사용'** 옵션을 체크합니다. 쿼리 결과를 재사용하려는 다른 페이지에서 **'재사용할 쿼리문'** 목록의 **'쿼리 결과 재사용'** 페이지 항목 중, 재사용하려는 페이지를 선택할 수 있습니다.

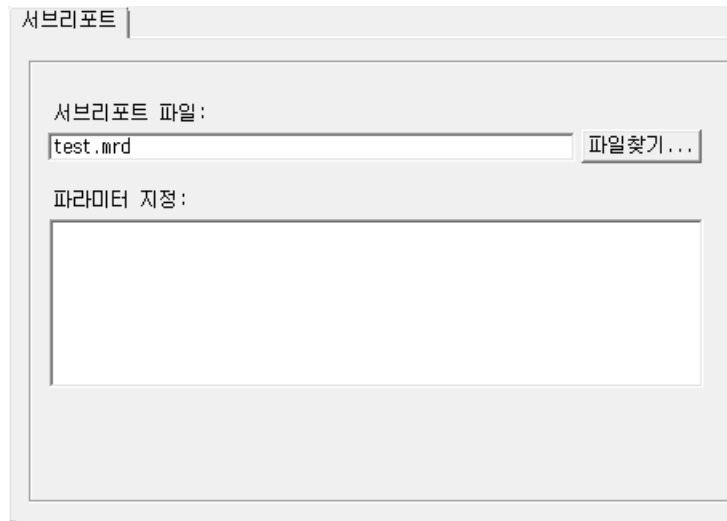


 단, 쿼리 결과데이터를 재사용하는 서브 페이지에서는 사용자가 별도로 쿼리를 편집할 수 없습니다.

4.2.10. 서브 리포트 사용

서브 리포트를 사용하면 별도의 파일로 저장된 문서를 다른 문서의 서브 페이지에 포함하여 마스터-디테일 구조뿐만 아니라 마스터-디테일-디테일...-디테일의 구조까지도 작성할 수 있습니다.

서브 리포트 사용을 체크하면 쿼리 정의 대화상자의 하단 부분이 아래의 그림처럼 변경됩니다.



서브 리포트 파일에는 서브 리포트로 포함될 양식 파일을 지정합니다. 파일의 경로는 절대경로뿐만 아니라, 서브 리포트파일을 포함하고 있는 파일의 경로를 기준으로 상대 경로도 지원됩니다. 또한, 파라미터 지정에는 서브 리포트수행을 위한 보고서 파라미터를 지정합니다. 이때, 파라미터로 주 쿼리의 필드 값이나 파라미터 변수 값을 사용하고 싶은 경우에는 해당 주 쿼리 필드명이나 파라미터 변수명을 **%RDS...%RDE**의 형태로 감싸주면 됩니다.

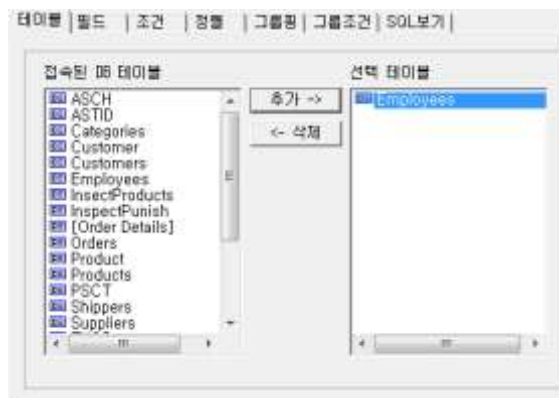
참고로, 서브 리포트 파일에 대한 설정은 [맞추기]-[페이지 옵션]-[서브 리포트 페이지 지정]에서도 동일하게 설정할 수 있습니다.

4.2.11. 쿼리 작성 탭 이용

쿼리 작성 탭을 이용하면 SQL 자유편집을 통해 쿼리문을 직접 입력하지 않아도 Wizard 방식을 통해 쿼리를 만들 수 있습니다.

□ 테이블

SQL의 **FROM 절**에 해당되며 출력할 데이터가 있는 테이블을 선택합니다.



접속된 데이터베이스 테이블 목록상자에서는 현재 접속한 데이터베이스에 있는 테이블을 보여줍니다. 이 중에서 SQL 문 작성에 필요한 테이블을 선택한 다음 **추가** 버튼을 누르면 선택한 테이블명이 **선택 테이블** 목록상자에 추가됩니다.

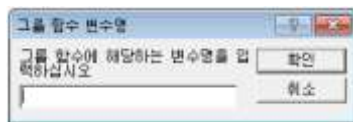
□ 필드

SQL의 **Select 절**에 해당되며 쿼리에 사용될 컬럼을 선택합니다.



테이블 탭에서 선택한 테이블과 그 테이블에 있는 컬럼이 **테이블 컬럼** 목록상자에 보여집니다. 출력할 데이터에 해당하는 컬럼을 선택하여 **선택 컬럼** 목록상자로 추가합니다.

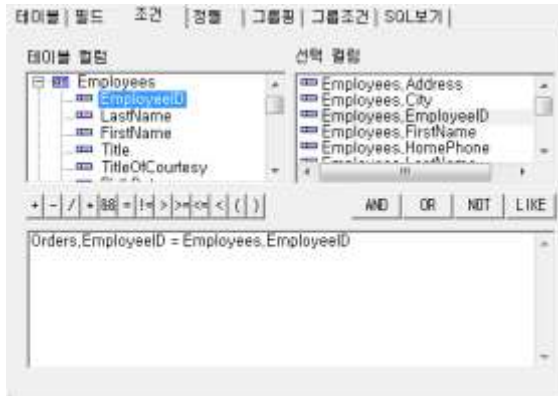
그룹 함수는 쿼리에 **GROUP BY 절**이 사용된 경우, **Select 절**의 컬럼값을 그룹핑해줄 때, 사용되는 기능입니다. 먼저 **테이블 컬럼** 목록상자에 있는 한 컬럼을 선택한 후, 그룹 함수 중 하나를 선택하여 추가 버튼을 누릅니다. 그러면 **그룹 함수 변수명** 대화상자가 나타나며 여기에 그룹 함수가 적용된 컬럼의 **Alias** 명을 입력하고 **확인** 버튼을 누르면 됩니다.



□ 조건

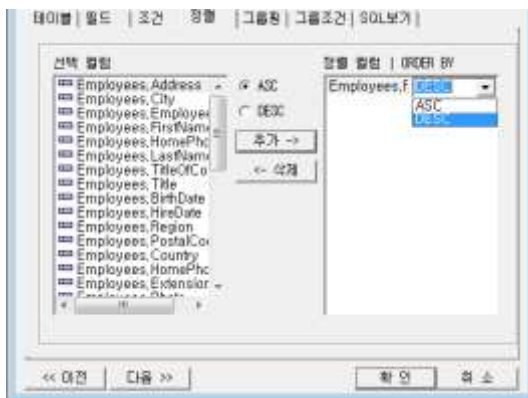
SQL의 **WHERE 절**에 해당되며 필드(컬럼)나 필드 사이의 조건을 작성합니다.

쿼리에 필요한 조건을 대화상자 하단의 입력란에 작성합니다. 선택 컬럼 목록상자의 컬럼명을 더블 클릭하면 입력란으로 입력되고 각 연산자도 마우스 클릭으로 입력할 수 있습니다. 입력란에 조건을 직접 작성하는 것도 가능합니다.



□ 정렬

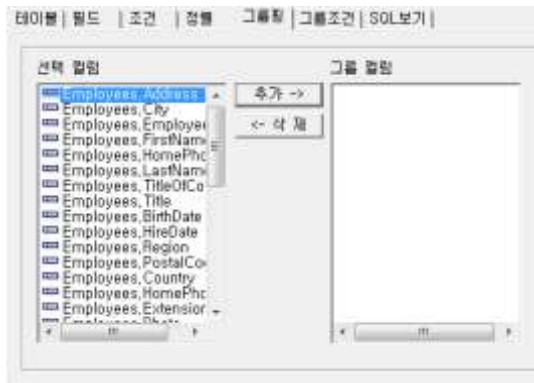
SQL의 **Order by 절**에 해당되며 출력될 레코드를 특정 컬럼 기준으로 오름차순(**ASC**) 또는 내림차순(**DESC**)으로 정렬하고자 할 때, 사용합니다.



선택 컬럼 목록상자에서 정렬에 사용할 컬럼을 선택하고 **ASC** 또는 **DESC**을 선택한 다음, 정렬 컬럼으로 추가하면 됩니다. 참고로 정렬 컬럼으로 추가한 후에도 정렬기준은 변경할 수 있습니다.

□ 그룹핑

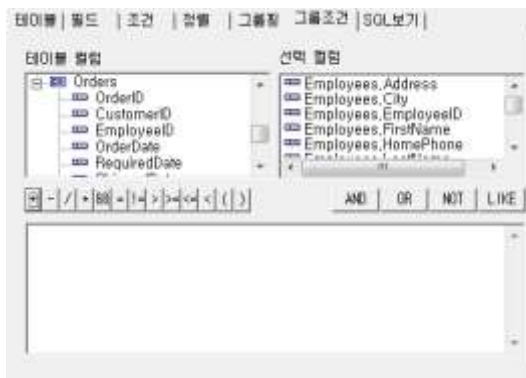
SQL의 **GROUP BY 절**에 해당되며 특정 컬럼을 기준으로 다른 컬럼의 값들을 그룹핑하고자 할 때, 기준 컬럼을 선택합니다.



선택 컬럼 목록상자에서 그룹핑에 사용될 컬럼을 선택해 추가 버튼을 누르면 그룹 컬럼 목록상자로 추가됩니다.

□ 그룹조건

SQL의 **HAVING** 절에 해당되며 그룹핑할 때의 조건을 입력합니다. 그룹핑 조건을 작성하며 작성법은 조건 탭과 동일합니다.



□ SQL 보기

최종적으로 작성된 쿼리를 보여주며 작성된 쿼리는 직접 편집할 수도 있습니다.



하단의 레코드 보기 버튼은 현재 작성된 쿼리로 가져온 레코드를 보여주며(4.2.2.2 레코드

보기 내용 참조), **SQL 검증** 버튼은 작성된 쿼리가 SQL 문법에 맞는지를 확인해줍니다.
(4.2.2.3 SQL 검증 내용 참조)

4.2.12. SAP 쿼리 정의

SAP 접속 문서에서 쿼리 정의 대화상자는 SAP 접속의 특성에 맞춰 설계되었으며 보고서 작성을 위해서 SAP R/3 시스템으로부터 가져올 데이터를 정의합니다.

4.2.12.1. 테이블 사용

테이블 레코드를 보고서 작성에 사용합니다.

□ 테이블



테이블명 일부를 입력하고 검색버튼을 누르면 검색된 테이블 리스트가 보여집니다. 검색된 테이블명 리스트에서 사용할 테이블명을 선택하여 추가합니다.

□ 필드



보고서 작성에 사용할 필드를 선택합니다.

□ 조건



레코드 검색 조건을 입력하고 ROW SKIP, ROW COUNTS 를 지정합니다.

4.2.12.2. RFC 함수 사용

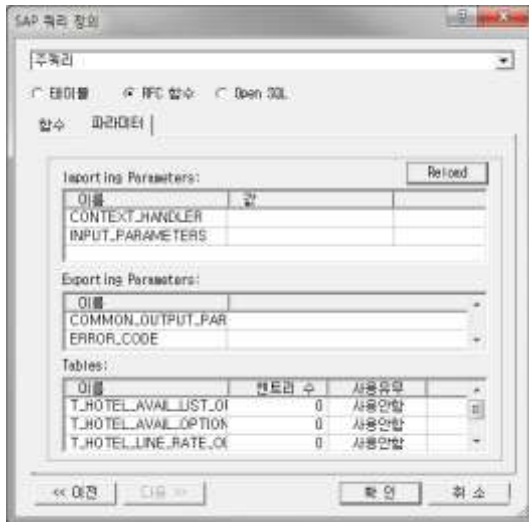
RFC 함수를 실행한 결과 데이터를 보고서 작성에 사용합니다.

□ 함수



함수명 일부를 입력하여 검색한 후, 검색된 RFC 함수명 리스트에서 사용할 RFC 함수명을 추가합니다.

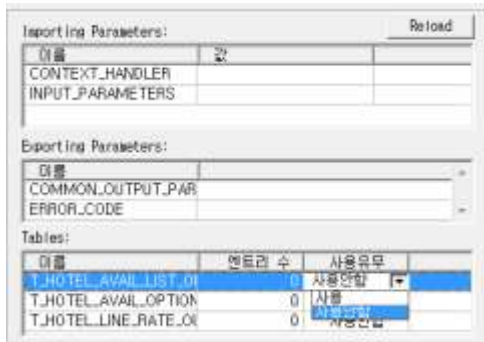
□ 파라미터



Importing Parameters - RFC 함수 호출에 사용할 Importing Parameters 의 값을 설정합니다.

Exporting Parameters - Exporting Parameters 입니다.

Tables - 사용 유무 컬럼을 편집해서 데이터로서 사용할 것인지를 결정합니다. 한 개의 테이블만 사용으로 설정해야 하며 설정한 테이블로부터 데이터를 받아서 보고서 작성에 사용 합니다.

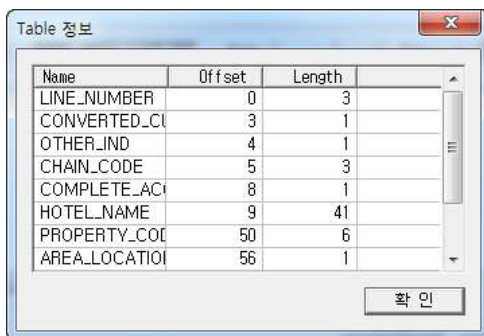


□ Table 편집

SAP 쿼리 정의 - 파라미터 탭에서 Tables 의 엔트리 수 컬럼을 더블 클릭하면 다음의 Table 편집 대화상자가 나타납니다. 대화상자를 이용하여 Table 의 엔트리를 지정합니다.



정보 버튼을 누르면 테이블의 컬럼 정보를 볼 수 있습니다.



+ 버튼을 누르면 엔트리 항목이 추가 되고 - 버튼을 누르면 현재 선택된 엔트리가 삭제됩니다.

엔트리 값을 입력하고 확인버튼을 누르면 지정한 Table 에 엔트리 추가가 완료됩니다.

4.2.13. FILEAGENT 쿼리 정의

TR Code 리스트에서 사용할 TR Code 를 선택하면 해당 TR 에 정의된 InBlock 과 OutBlock 리스트가 각각의 데이터 창에 나타납니다. InBlock 리스트에서는 InBlock Item 에 Input 데이

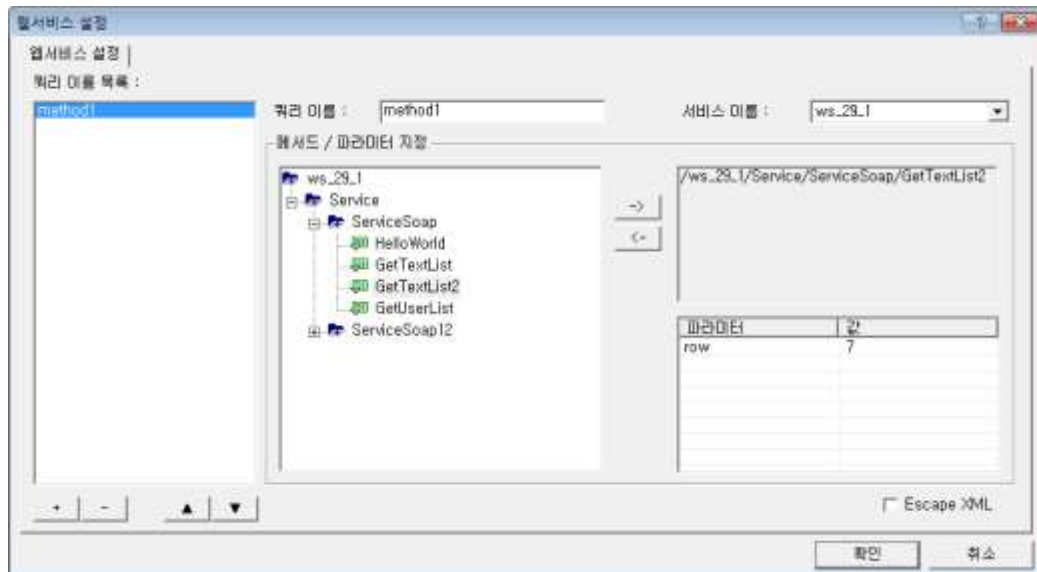
터를 입력하고, **OutBlock** 리스트에서는 각각의 **OutBlock** 마다 출력될 페이지를 지정합니다. 하나의 **OutBlock** 은 하나의 주 쿼리 혹은 서브 쿼리가 됩니다.

TR Code 검색 기능은, 입력한 문자열이 포함된 **TR Code** 들과 체크된 **TR Code** 들만 리스트에 표시되도록 하는 기능이며, '*' 문자를 지원합니다. 또한 **선택한 TR Code** 만 표시 체크박스를 선택하면 사용자가 선택한 **TR Code** 들만 리스트에 표시합니다. 그리고 +, - 버튼을 이용하여 선택된 하나 이상의 **TR Code** 들 중에서 우선순위를 지정할 수 있습니다.



4.2.14. 웹 서비스 설정

Data Server 접속방식 보고서 이고 데이터베이스 접속 대화상자에서 서비스 유형을 **webservice** 로 선택 한 경우, 웹 서비스 설정 대화상자에서 웹 서비스 호출을 위한 메서드, 파라미터 지정을 합니다.



+ 버튼 : 새 쿼리 추가

- 버튼 : 선택 한 쿼리 삭제

▲ 버튼 : 선택 한 쿼리를 위로 위치 이동

▼ 버튼 : 선택 한 쿼리를 아래로 위치 이동


쿼리 이름 : 선택 한 쿼리의 이름


서비스 이름 : 선택 한 쿼리가 사용하는 서비스 이름. 데이터베이스 접속 대화상자에서 선택한 서비스명 목록이 서비스 이름 목록에 표시 됩니다. 서비스 이름을 변경 하면 메서드 / 파라미터 지정을 위한 XML 트리가 갱신 됩니다.

메서드 / 파라미터 지정 : XML 트리에서 호출 할 메서드를 선택하고 마우스 더블클릭을 하거나 -> 버튼을 눌러서 메서드를 지정 합니다. 파라미터 값 지정시 /rp 또는 /rv 변수명을 이용할 수 있으며 /rp 또는 /rv 변수값은 파라미터 값 입력 대화상자에서 미리 지정 되어 있어야 합니다. ([홈]-[파라미터 값 입력])

Escape XML : 선택 한 쿼리의 Escape XML 값을 지정. Escape XML 을 체크 표시 하면, 웹 서비스 호출 결과 XML 에서 < > 엔터티를 데이터가 아닌 XML 태그로 처리 합니다.

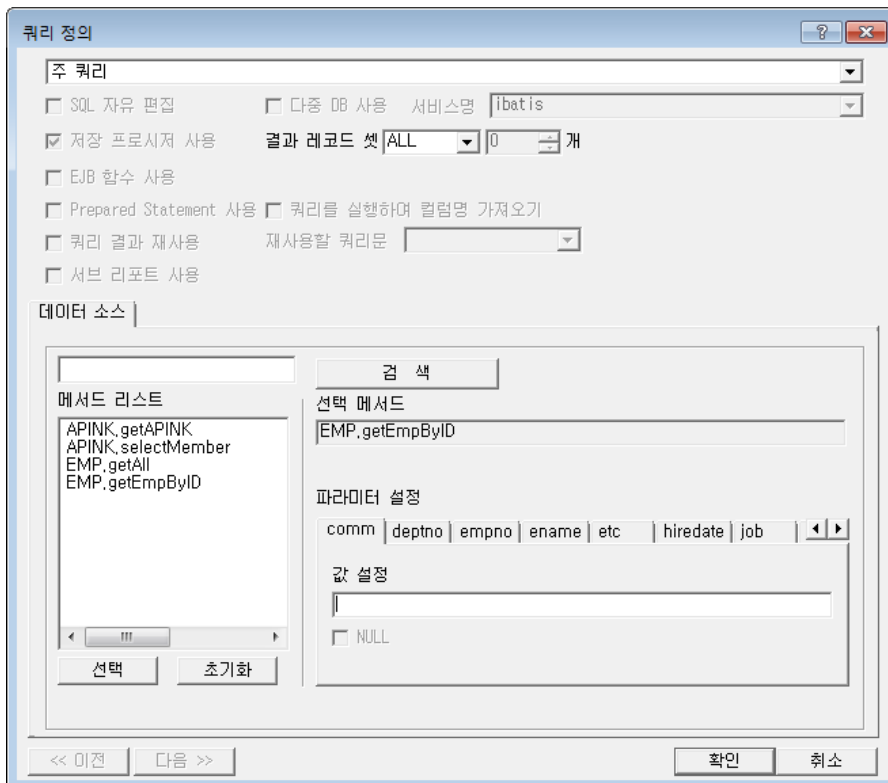
웹서비스 설정 대화상자에 정보 입력 후 확인 버튼을 누르면 Data Server 를 통하여 웹 서비스를 호출 하게 됩니다. 그리고 Data Server 에서 가져온 결과 XML 은 XML 쿼리정의 대화상자에서 이용 합니다.

 Data Server 에서 가져온 결과 XML 데이터는 보고서 양식 파일에 저장되지 않기 때문에 XML 쿼리정의 기능을 이용 하려면 웹서비스 설정을 먼저 해야 합니다.

 Report Viewer 는 Data Server 를 통하여 웹 서비스를 호출 하게 되고, 가져온 결과 XML 을 이용하여 XMLFILE 접속방식 보고서 리포팅을 하게 됩니다. 따라서 웹 서비스 연동 보고서 실행을 하려면 MSXML 파서 배포가 필요 합니다.

4.2.15. iBatis, plugin 연동 보고서 쿼리 정의

Data Server 접속방식 보고서 이고 데이터베이스 접속 대화상자에서 서비스 유형을 iBatis 또는 plugin 으로 선택 한 경우, 쿼리 정의 대화상자는 다음과 같습니다.



각 기능에 대한 설명은 '4.2.4 저장 프로시저 사용' 을 참고 합니다.

4.3. 필드 쿼리 정의

쿼리 정의 대화상자에서 지정한 쿼리문을 실행하여 그 결과 레코드 값을 가져올 때, 원하는 모든 데이터 값을 하나의 쿼리로 가져올 수 없는 경우가 많이 있습니다. 이런 경우 주 쿼리나 서브 페이지 쿼리에서 가져온 값을 가지고 다른 쿼리문장을 만들어서 원하는 출력 값을 가져올 수 있는데 이때, 사용하는 것이 **필드 쿼리**입니다.

[데이터 소스]-[필드 쿼리 정의]를 선택하면 **필드 쿼리** 대화상자가 나타납니다. 필드 쿼리 대화상자에서 상단의 추가버튼을 눌러 보고서에서 사용할 필드 쿼리명을 추가합니다.




보통의 경우, 필드 쿼리에서 **SELECT** 할 수 있는 필드는 오직 하나 뿐이며 **WHERE** 절에서 주 쿼리 필드나 파라미터 변수 값을 참조할 경우에는 **:(주 쿼리필드명)** 이나 **:\$1)** 의 형태로 작성합니다. 서브 쿼리 필드를 참조할 경우도 **:(서브 쿼리필드명)** 의 형태로 작성합니다. 만일 주 쿼리 필드의 데이터 형태가 문자일 경우에는 **:(주 쿼리필드명)** 과 같이 입력합니다.

만약, 필드 쿼리에서 **Select** 절에 여러 개의 필드를 사용하고자 할 때에는 필드 쿼리명에 **# 기호**를 붙이면 됩니다. 예를 들어, 필드 쿼리 명에 “인적사항#” 이라고 입력하고 쿼리문장을 “**Select A, B, C From T**”라고 작성하실 수 있습니다. 이런 경우에 **A** 컬럼은 인적사항 **1**, **B** 컬럼은 인적사항 **2**, **C** 컬럼은 인적사항 **3** 이 되며 데이터 셋 연결 대화상자의 필드명에 해당 필드 쿼리명을 직접 입력하여 원하는 객체에 적용하거나 요약 작성에서 사용하실 수 있습니다.

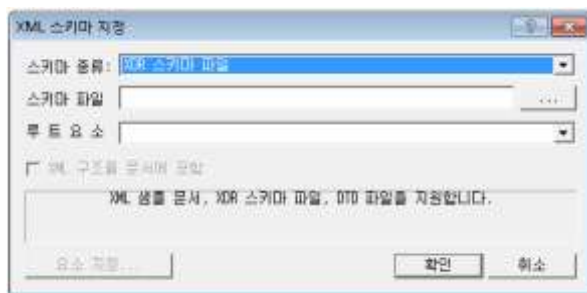
또한, 주 쿼리문이나 서브 페이지 쿼리문의 **Select** 절에서 일반 필드와 **Blob 필드**를 함께 사용할 수 없으므로 이런 경우에는 필드 쿼리를 사용해야 합니다. 주 쿼리문이나 서브 페이지 쿼리문의 **Select** 절에서는 일반 필드만 가져오도록 하고 **Blob 필드**는 필드 쿼리를 이용하여 별도로 가져오도록 합니다. 필드 쿼리명에 **# 기호**를 붙여 **Select** 절에서 여러 개의 필드를 가져오는 경우에는 반드시 일반 필드만 사용해야 하며 **Blob 필드**는 함께 사용할 수 없습니다.

다중 데이터베이스 서비스명은 **4.2.3 다중 데이터베이스 사용** 을 참조하시고 **Prepared Statement 사용**은 **4.2.7 Prepared Statement 사용** 을 참조하십시오.

 또한, 필드 쿼리는 Crownix ERS Data Server (JAVA or .NET) 에 접속하여 사용할 경우, 데이터베이스에 있는 Blob 필드(이미지나 메모 데이터)를 가져오는 경우에도 사용합니다.

4.4. XML 스키마 지정

XMLFILE 접속 문서에서 XML 문서구조를 정의한 스키마 파일을 지정하기 위해 [데이터 소스]-[XML 스키마 지정] 을 선택하면 아래의 그림과 같은 대화상자가 나타납니다.



스키마 파일은 보통 XML 문서의 유효성 검사에 사용합니다. **Crownix Report**에서는 XML 문서구조 정보를 얻기 위해서 스키마 파일을 이용합니다.

4.4.1. 스키마 종류 지정

사용할 스키마 파일의 종류를 지정합니다. XDR 스키마 파일, DTD 파일, XML 샘플 문서 중에서 선택할 수 있습니다.

4.4.1.1. XDR 스키마 파일

XML-Data Reduced (XDR) language 는 MSXML 파서가 지원하는 스키마이며 네임스페이스 선언은 다음과 같이 합니다.

```
<Schema name="fooSchema" xmlns="urn:schemas-microsoft-com:xml-data"
xmlns:dt="urn:schemas-microsoft-com:datatypes">
```

다음은 XDR 스키마 파일 예제 입니다. (authors-schema.xml)

```
<Schema name="authors" xmlns="urn:schemas-microsoft-com:xml-data
xmlns:dt="urn:schemas-microsoft-com:datatypes">
  <ElementType name="name"/>
  <ElementType name="nationality"/>
  <ElementType name="authors" content="eltOnly" model="closed">
    <element type="author" maxOccurs="*" />
  </ElementType>
```

```

<ElementType name="author" model="closed" content="eltOnly" >
  <element type="name"/>
  <element type="nationality"/>
</ElementType>
</Schema>

```

4.4.1.2. DTD 파일

XML Document Type Definition(DTD)도 XML 문서형 정의 파일입니다. **Crownix Report** 에서는 DTD의 일부만을 지원하고 있습니다. 지원하고 있는 DTD 스펙은 아래와 같습니다.

엘리먼트 선언 : 제한적으로 구현됨. (엔티티사용을 안 하는 경우)
속성 선언 : 제한적으로 구현됨. (엔티티사용을 안 하는 경우, @type 속성만 구현됨)

다음은 DTD 파일 예제입니다.(authors.dtd)

```

<!ELEMENT authors (author*)>
<!ELEMENT author (name , nationality)>
<!ELEMENT name (#PCDATA)>
<!ELEMENT nationality (#PCDATA)>

```



DTD 파일이 현재 지원하고 있지 않은 요소선언을 포함하고 있으면 정상적으로 스키마지정을 할 수 없습니다.

4.4.1.3. XML 샘플 문서

Crownix Report에서는 XDR 스키마 파일, DTD 파일, XSD 스키마 파일 뿐만 아니라 XML 샘플 문서를 통해서도 XML 문서의 구조를 가져올 수 있습니다.

4.4.1.4. XSD 스키마 파일

Crownix Report에서는 W3C 표준인 XSD 스키마 파일을 사용하여 보고서 작성에 사용할 XML 데이터 파일의 구조와 #PCDATA 요소, 반복되는 요소 등을 정의할 수 있습니다.

XSD 스키마의 네임스페이스는 <http://www.w3.org/2001/XMLSchema> 이고, XSD 스키마에 대한 자세한 정보는 <http://www.w3.org> 웹사이트를 참고할 수 있습니다.



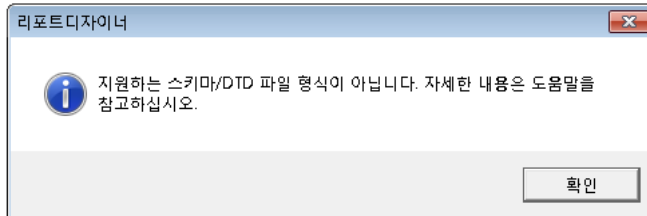
XSD 스키마 파일을 사용하여 보고서 디자인을 하려면 XSD 스키마 파일 지원하는 별도의 옵션 팩을 설치해야 하며, 사용하려는 로컬 PC에 .NET Framework 2.0 이 설치되어 있어야 합니다.

4.4.2. 스키마 파일 지정

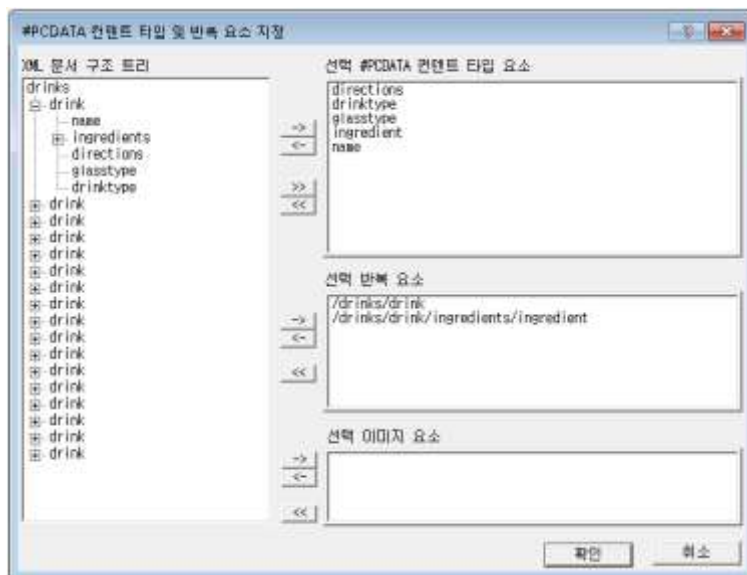
XDR 스키마 파일이나 DTD 파일, XML 샘플 문서 또는 XSD 스키마 파일의 전체경로를 지정합니다. 스키마 파일을 지정하면 루트요소 콤보 박스에 선택한 스키마의 모든 엘리먼트 요소

가 추가됩니다.

지원하는 스키마 파일이 아니면 다음과 같은 오류 메시지를 출력합니다.

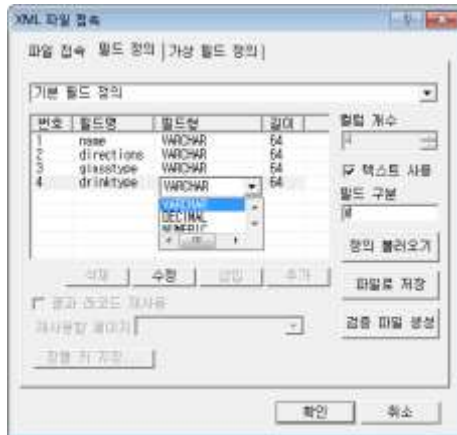


XML 샘플 문서는 XDR 스키마 파일, DTD 파일, XSD 스키마 파일을 사용하는 경우와 달리, **#PCDATA 컨텐트 타입 요소**와 **반복 요소**를 자동으로 인식하지 못합니다. 따라서 스키마 파일로 XML 샘플 문서를 지정한 경우에는 아래의 그림과 같은 대화상자를 통해서 **#PCDATA 컨텐트 타입 요소**와 **반복 요소**를 지정해 주어야 합니다.



□ 이미지 요소 지정

XML 샘플 문서를 사용하는 경우, **Base64** 타입으로 인코딩된 특정 요소를 이미지로 출력할 수 있습니다. 사용법은 해당 요소를 **이미지 요소**로 지정한 후, **[데이터 소스]-[파일 접속]**을 선택하여 **필드 정의**할 때, 이미지로 표현할 요소의 필드형을 **LONGVARBINARY**로 지정하면 됩니다.



4.4.3. 루트요소 지정

지정한 스키마 파일에 정의되어 있는 엘리먼트 요소의 리스트 중 최상위 루트요소를 선택합니다. 선택한 루트요소는 주 쿼리의 최상위 요소 값입니다. [데이터 소스]-[XML 쿼리 정의] 대화 상자를 열어보면 루트요소가 주 쿼리의 최상위 요소로 지정된 것을 확인할 수 있습니다.

XSD 스키마 파일을 사용하는 경우에는 사용할 루트 요소를 직접 입력해주어야 합니다.

4.4.4. XML 구조를 문서에 포함

XML 샘플 문서를 이용하여 XML 스키마 지정은 한 경우, XML 샘플 문서가 MRD 파일에 저장되지 않기 때문에 지정된 XML 스키마를 수정해야 할 때, XML 샘플 문서 파일이 지정된 위치에 없으면 “스키마/DTD 파일/XML 샘플 파일이 지정한 경로에 존재하지 않습니다.” 라는 알림 메시지가 뜨면서 파일을 다시 지정 하라고 알려주는 불편 사항이 있습니다.

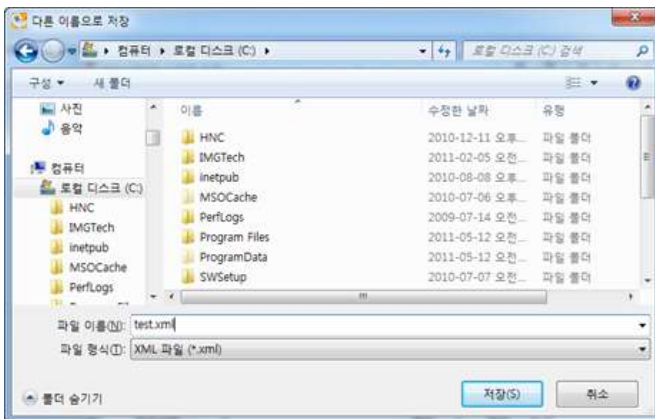
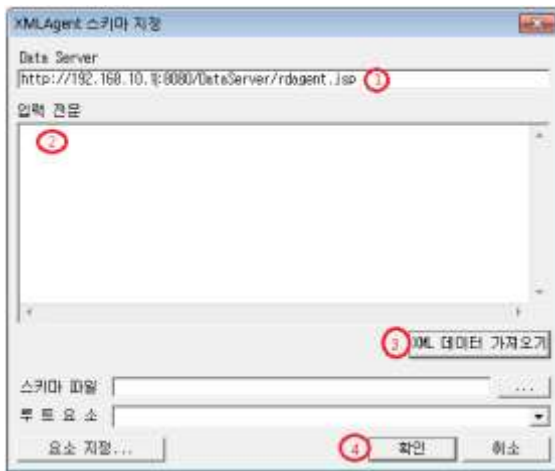
하지만, **XML 구조를 문서에 포함** 기능을 체크한 경우에는 XML 샘플 문서 파일이 없어도 문서에 저장된 XML 구조를 이용하여 지정된 스키마를 수정할 수 있습니다. 단, XML 구조가 아예 바뀌는 경우에는 **XML 구조를 문서에 포함** 기능을 해제한 후에 변경된 구조의 XML 파일 경로를 지정해야 합니다.

4.4.5. XMLAGENT 스키마 지정

XMLAGENT 접속 방식을 사용하고, AGENT 선택 옵션에서 **Data Server-PW** 나 **Data Server-AF** 를 선택한 경우, 서버로부터 XML 데이터를 내려받아 스키마 지정은 합니다. 단, 스키마 지정 방법이 다른, 나머지는 **XMLFILE** 접속 방식의 경우와 동일합니다.

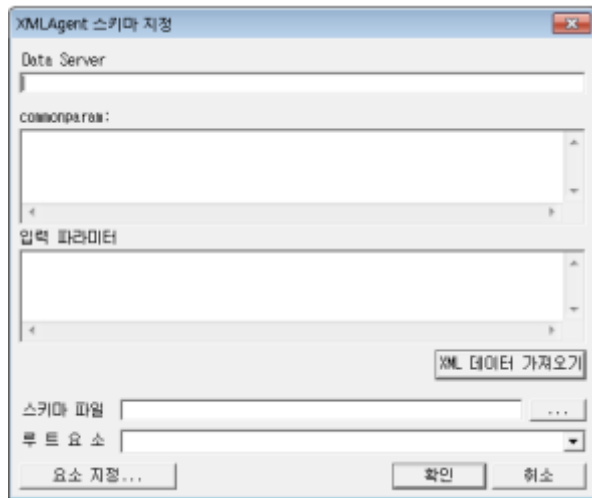
□ Data Server-PW

Data Server 경로에는 **Crownix ERS Data Server** 의 경로를 입력하고, **Input 전문**에는 원하는 XML 데이터를 얻어올 수 있는 **Input 전문**을 입력한 후, **XML 데이터 가져오기** 버튼을 누릅니다. 이때, 스키마 파일 경로가 비어있으면, “**먼저 내려받을 XML 파일의 경로를 지정하십시오.**” 라는 메시지가 나타납니다. 그러면 로컬에 XML 데이터를 저장할 파일 경로를 지정해 줍니다. 파일 경로를 지정하고 **저장** 버튼을 누르면, 서버에서 XML 데이터를 가져오게 되는데, 이때 XML 데이터를 가져오는 시간이 오래걸리면 화면이 멈춰있는 것처럼 보일 수 있습니다. 서버로부터 XML 데이터를 가져와서, **#PCDATA 콘텐츠 타입 요소 / 반복 요소 지정** 대화상자가 뜨면, 원하는 대로 스키마 지정이 가능합니다.



□ **Data Server-AF**

사용법은 **Data Server-PW** 의 경우와 동일하나, 입력해야 하는 창이 **commonparam** 과 **inputparam** 두 가지인 것만 다릅니다.



4.5. XML 쿼리 정의

XML 쿼리 정의는 XML 문서로부터 데이터를 추출해 오기 위한 **XSLT 스타일시트**를 정의하는 작업입니다. SQL 쿼리 정의와 유사한 방식으로 XML 쿼리 정의를 수행할 수 있도록 **필드 정의** 탭과 **조건정의** 탭, **정렬/기타정의** 탭, **XSLT 보기** 탭을 제공하므로 손쉽게 쿼리를 정의할 수 있습니다.

4.5.1. SQL 쿼리 정의와 XML 쿼리 정의 비교

□ 데이터 액세스 방법

관계형 데이터베이스의 데이터를 액세스 하기 위해서는 아래의 쿼리문 예제와 같이 **테이블명. 컬럼명**을 사용합니다.

```
SELECT Employees.EmployeeID, Employees.FirstName, Employees.HomePhone  
FROM Employees
```

하지만, XML 문서의 데이터를 액세스하기 위해서는 **XPath(XML Path Language)**를 사용합니다. XPath란, 아래의 XPath 예제에서와 같이 파일시스템의 디렉토리/파일 구조와 유사한 트리 구조입니다.

```
authors/author/name - authors 요소의 자식요소 author, 그 자식요소로의 name 요소에 액세스 하기 위한 XPath
```

```
authors/author[nationality='Russian']/name - author 의 nationality 값이 'Russian' 인 사람을 액세스 하기 위한 XPath
```

□ 쿼리 정의 결과

SQL 쿼리 정의를 완료하면 완성된 SQL 쿼리문이 생성됩니다. 하지만, XML 쿼리 정의를 완료하면 XSLT 스타일시트가 생성됩니다. 이들은 보고서 작성시에 데이터 소스로부터 데이터를

추출하기 위해서 사용됩니다.

4.5.2. 쿼리이름 과 최상위 요소

□ 쿼리이름

쿼리 정의를 위한 주 쿼리 또는 서브 쿼리를 선택합니다.

□ 최상위 요소

XML 문서에서 가져올 데이터를 XPath 로 지정하려면 선택 요소의 최상위 요소를 먼저 지정해야 합니다. 선택 요소는 최상위 요소의 상대 XPath 로 지정합니다. 이때, 주 쿼리의 최상위 요소는 루트요소이고 서브 쿼리의 최상위 요소는 반복요소입니다. 단, 이미 사용중인 최상위 요소는 다시 사용할 수 없습니다.

□ 최상위 요소 선택하기

주 쿼리의 최상위 요소는 **XML 스키마 지정**에서 지정한 루트요소가 선택되고 수정할 수 없습니다.

서브 쿼리의 최상위 요소는 최상위 요소 콤보 박스에 나열된 요소들 중 하나를 선택할 수 있습니다. 나열된 요소들은 XML 문서에서 반복 가능한 요소들의 XPath 입니다. 따라서 XML 문서 구조에 반복요소가 없다면 서브 쿼리를 사용할 수 없습니다.

□ 최상위 요소 XPath 와 선택 요소 XPath

최상위 요소는 해당쿼리 XML 문서구조 트리의 루트요소입니다. 선택 요소의 XPath 는 최상위 요소의 상대 경로입니다. 즉, 최상위 요소 XPath 가 **'/authors'** 이고 선택 요소 XPath 가 **'/author/name'** 이면 선택 요소의 XPath 는 **'/authors/author/name'** 입니다.

XSLT 스타일시트에는 `xsl:template` 요소가 각 쿼리(주 쿼리/서브 쿼리) 마다 1 개씩 생성됩니다. 이때, 최상위 요소는 `xsl:template` 요소의 `match` 속성의 값으로 사용됩니다.

XML 쿼리 정의를 하여 생성된 XSLT 문서를 보면 `xsl:template` 요소가 있습니다. 최상위 요소는 이 요소의 `match` 속성의 값이 됩니다. 예를 들어, 주 쿼리의 최상위 요소가 **'/authors'** 인 경우 XSLT 스타일 시트는 아래와 같이 만들어집니다.

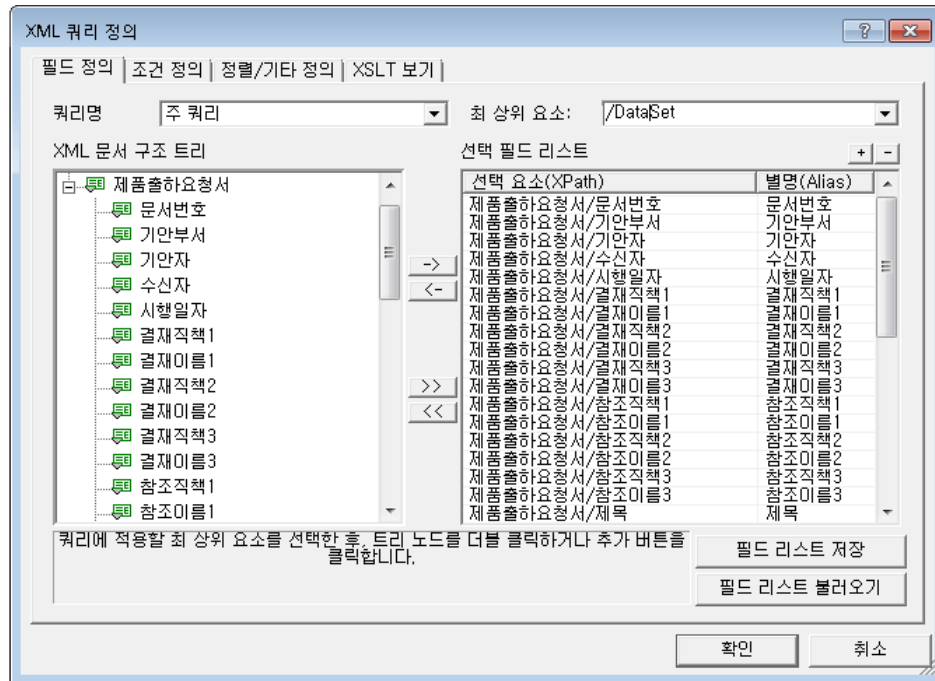
```
<xsl:template match="/authors">
  </xsl:template>
```

4.5.3. 필드 정의

XML 문서에서 가져올 데이터 요소의 XPath 를 지정합니다.

대화상자의 왼쪽 **XML 문서구조 트리**에서 선택할 요소를 선택하고 추가버튼을 누르거나 마우

스 더블클릭합니다. 선택 필드 리스트에는 요소의 최상위 요소에 대한 상대 XPath 와 별명이 입력 됩니다. 이때, 추가할 수 있는 요소는 Mixed 형태의 엘리먼트 요소(📄)와 속성 요소(📄)입니다.



□ XML 문서구조 트리

최상위 요소를 루트요소로 하는 **XML 문서구조 트리**를 표시합니다. XML 문서에서 해당 데이터를 추출해 오기 위해서 노드를 **선택 필드 리스트**로 추가합니다.

□ 선택 필드 리스트

XML 문서로부터 가져올 데이터를 XPath 로 지정합니다.

□ 선택 요소(XPath)

선택 요소는 XML 문서 내에서 추출할 데이터의 XPath 입니다. 최상위 요소 XPath 와 선택 요소 XPath 로 XML 문서내의 데이터를 액세스할 수 있습니다.

□ 별명(Alias)

별명란을 클릭하면 선택 요소의 별명을 수정할 수 있습니다. 별명은 데이터 셋 연결에 사용되는 선택 요소의 다른 이름입니다. 단, 별명은 같은 이름이 중복되면 안됩니다.

□ 선택 요소의 순서 바꾸기

선택 필드 리스트에 요소를 추가하면 마지막 요소로 추가됩니다. 요소의 위치를 변경하고 싶은 경우, **+** 버튼을 누르면 요소가 위로 이동하고 **-** 버튼을 누르면 요소가 아래로 이동합니다.

□ 요소 추가/삭제

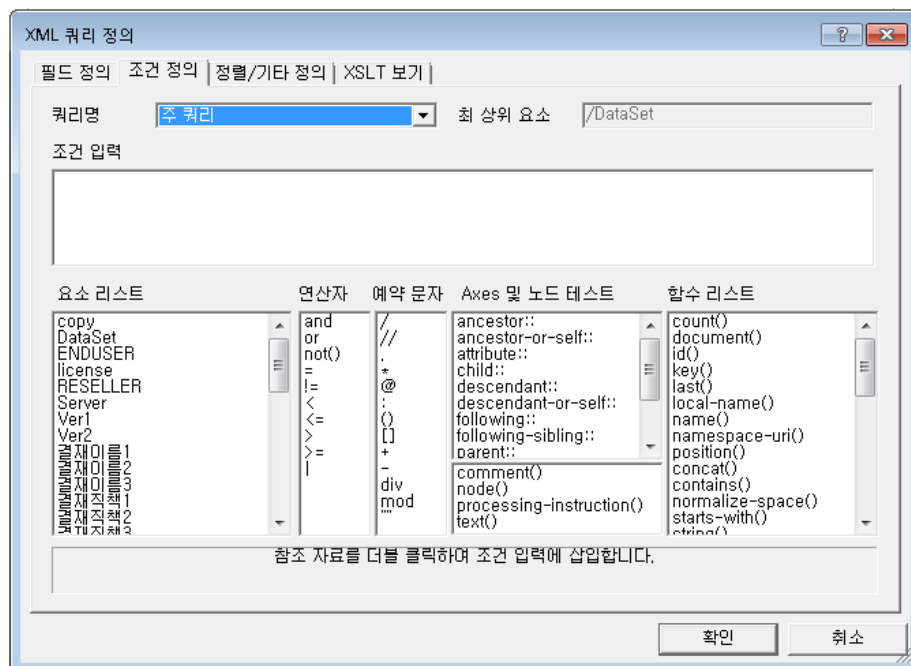
XML 문서구조 트리에 있는 요소를 선택 필드 리스트로 추가하고자 하는 경우에는 **->** 버튼을 누릅니다. 또한 선택 필드 리스트에 있는 요소를 삭제하고자 하는 경우에는 **<-** 버튼을 누릅니다. 그리고 **>>** 버튼을 누르면 XML 문서구조 트리에 있는 모든 요소를 선택 필드 리스트로 전체 추가할 수 있고 **<<** 버튼을 누르면 선택 필드 리스트에 있는 요소를 전체 삭제할 수 있습니다.

□ 필드 리스트 저장 / 필드 리스트 불러오기

지정되어 있는 필드 리스트를 txt 파일로 저장하고 다시 불러올 수 있는 기능입니다. 필드 개수가 많은 경우, txt 파일로 저장하여 편집하고자 할 때 사용 가능하며, 비슷한 필드 리스트 구조를 다른 페이지 혹은 다른 보고서에서 사용할 경우, 필드 리스트를 txt 파일로 저장하여 해당 페이지에서 다시 불러오기를 할 수 있습니다.

4.5.4. 조건정의

데이터 추출 조건을 입력합니다. 주 쿼리 또는 서브 쿼리에 조건절을 입력하면 조건에 맞는 데이터를 XML 문서에서 추출하여 리포팅합니다.



□ 조건 입력

쿼리에 적용할 조건을 입력합니다. 참고로 조건절에서 파라미터 변수 값을 이용할 수도 있으며 요소의 데이터형이 문자형인 경우에는 작은따옴표(')를 사용합니다.

name='홍길동'

../@유형='\$1'

□ 요소리스트

XML 문서에서 사용되는 엘리먼트이름, 속성이름이 나열됩니다. 요소를 더블 클릭하면 조건 입력란에 삽입됩니다.

□ 연산자, 예약문자, Axes 및 노드테스트, 함수리스트

조건절 입력에 이용할 수 있는 예약문자, 함수리스트입니다. 요소를 더블 클릭하면 조건 입력란에 삽입됩니다. 이에 대한 자세한 설명은 XSL 스펙을 참고하십시오. (관련사이트 - <http://www.w3c.org/>)

□ 주의점

엘리먼트이름, 속성이름을 조건절에 사용하는 경우, XPath 는 상대 경로로 입력해야 합니다. 상대 경로의 기준요소는 주 쿼리, 서브 쿼리에 따라서 다릅니다. 주 쿼리에 입력하는 경우는 반복 요소가 기준요소가 됩니다. 서브 쿼리인 경우는 최상위 요소가 기준요소가 됩니다. 상대 경로를 올바르게 입력해야 정상적으로 데이터를 추출할 수 있습니다.

주 쿼리의 반복요소가 **data/region** 이고 조건절에 **name** 요소를 추가 했다면 **data/region/name** 이 됩니다.

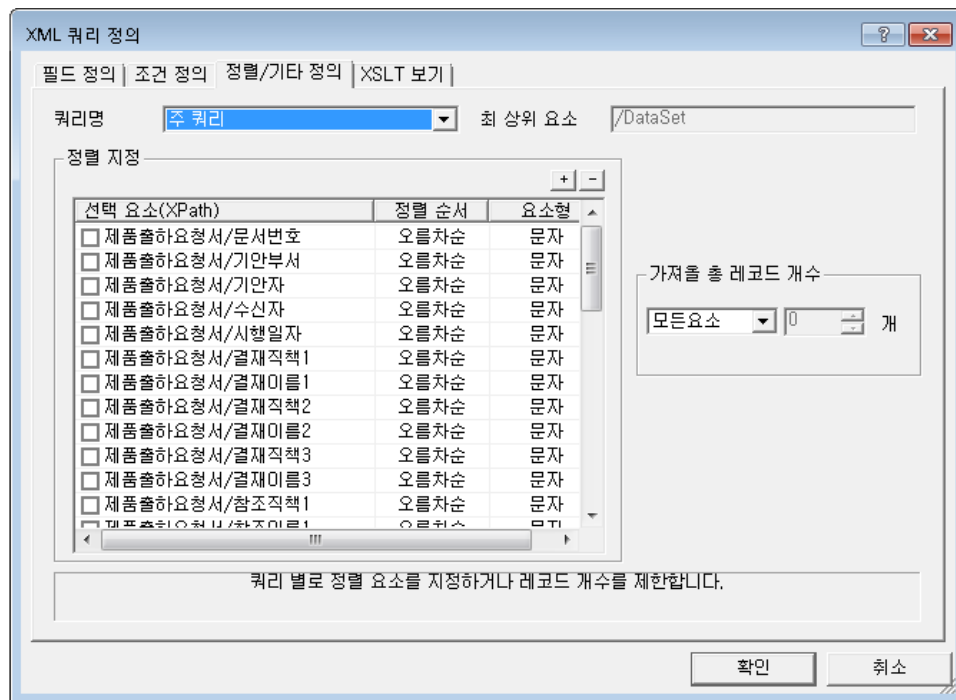
서브 쿼리의 최상위 요소가 **/sales/data/region/quarter** 이고 조건절에 **@number** 요소를 추가했다면 **@number** 는 **/sales/data/region/quarter** 의 상대패스 이므로 **/sales/data/region/quarter/@number** 가 됩니다.

조건절을 입력한 후에 생성한 XSLT 를 확인하면 다음과 같이 확인할 수 있습니다.

```
<xsl:template match="주 쿼리 최상위 요소">
  <xsl:for-each select="주 쿼리 반복요소">
    <xsl:if select="[조건절]">
      <xsl:if>
        <xsl:for-each>
          <xsl:template>
            <xsl:template match="서브 쿼리 최상위 요소[조건절]">
              <xsl:template>
```

4.5.5. 정렬/기타 정의

정렬필드를 지정하면 정렬된 데이터를 가져올 수 있습니다. 정렬 순서(오름차순/내림차순)와 요소형(문자/숫자)를 지정할 수 있습니다. 가져올 총 레코드 개수는 모든 요소, 상위요소 N 개, 하위요소 N 개로 지정할 수 있으며 지정한 개수만큼의 레코드를 가져 옵니다.



□ 선택 요소

체크박스를 체크하면 해당 필드 이름으로 정렬하여 데이터를 가져옵니다. 지정을 취소하려면 체크박스의 체크를 OFF 상태로 합니다.

□ 정렬 순서

오름차순, 내림차순을 지정합니다.

□ 요소형

문자, 숫자로 요소형을 지정합니다.

□ 위치이동

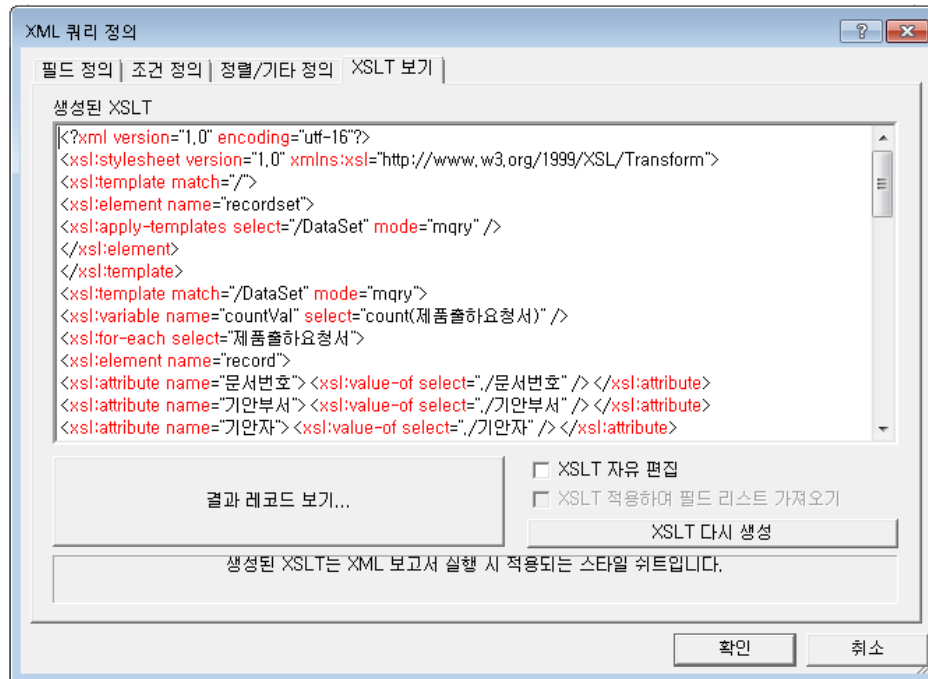
여러 개의 필드 이름에 정렬지정을 한 경우에는 위에 위치한 요소가 높은 우선순위를 갖게 됩니다. 우선순위를 변경하고 싶은 경우, **+** 버튼을 누르면 요소가 위로 이동하고 **-** 버튼을 누르면 요소가 아래로 이동합니다.

□ 가져올 총 레코드 개수 지정

가져올 레코드 개수를 제한합니다. 모든 요소를 지정하면 쿼리 정의한 조건에 맞는 모든 레코드를 가져옵니다. 예를 들어, 상위요소 N 개를 지정하면 상위의 N 개 레코드를 가져오고 하위 요소 N 개를 지정하면 하위의 N 개 레코드를 가져옵니다.

4.5.6. XSLT 보기

XML 쿼리 정의를 통해 최종적으로 생성된 XSLT 스타일시트를 보여 줍니다.



XSLT 스타일시트는 여러 개의 `xsl:template` 요소로 구성된 XML 파일입니다. XSLT에 대한 자세한 사항은 스펙을 확인하십시오. (관련사이트 - <http://www.w3c.org>)

□ XSLT 자유 편집

XSLT 자유 편집 속성을 적용하면 XML 쿼리 정의 대화상자의 탭을 이용하여 위자드 방식으로 생성된 XSLT 스타일 시트를 사용자가 자유롭게 재편집할 수 있습니다.

□ XSLT 적용하여 필드 리스트 가져오기

XSLT 자유 편집을 사용할 경우, 자동으로 생성된 필드 리스트가 아닌 수정된 XSLT를 적용하여 필드 리스트를 새로 작성하고 싶은 경우에 사용합니다.

□ XSLT 새로 생성

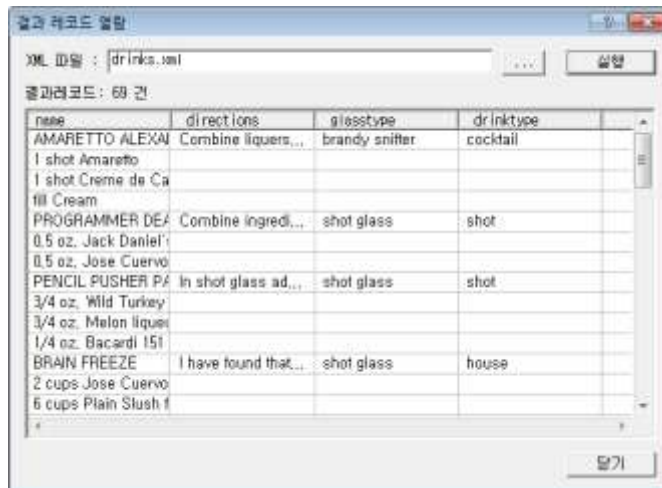
XML 쿼리 정의 대화상자에서 필드 정의, 조건정의 정렬, 기타정의 탭을 이용하여 Wizard 방식으로 XSLT 스타일 시트를 새로 생성할 수 있습니다.

□ 특징

주 쿼리와 각 서브 쿼리에 대한 XML 쿼리 정의는 결과적으로 1개의 `xsl:template` 요소로 만들어지며 최상위 요소는 스타일시트 템플릿의 매칭요소가 됩니다.

4.5.7. 결과 레코드 보기

생성한 결과 XSLT 를 XML 문서에 적용하여 레코드를 열람하는 기능입니다. XSLT 보기 탭의 결과 레코드 보기 버튼을 눌러서 결과 레코드 열람을 할 수 있습니다.



□ XML 파일


레코드열람을 위해서 XML 데이터파일을 선택합니다.

□ 실행

지정한 XML 데이터파일에 XSLT 스타일시트를 적용하여 레코드 가져오기를 실행합니다.

□ 결과 레코드

XML 문서에서 가져온 결과 레코드를 표 형식으로 표시합니다. 서브 쿼리를 사용하는 경우에 데이터는 주 쿼리, 서브 쿼리 1, 서브 쿼리 2 가 연속되어 표시되며 타이틀은 주 쿼리의 타이틀입니다.

 조건정의에서 파라미터 변수(\$)를 사용한 경우에는 파라미터 변수(\$)를 실제 값으로 치환한 후에 레코드 열람을 실행해야 합니다.

4.5.8. 생성된 XSLT 살펴보기

다음은 sales.xml 파일에서 데이터를 가져오기 위한 XSLT 스타일시트입니다.

```

1 <?xml version="1.0" encoding="UTF-16"?>
2 <xsl:stylesheet version="1.0" xmlns:xsl="http://www.w3.org/1999/XSL/Transform">
3 <xsl:template match="/">

```

```
4 <xsl:element name="recordset">
5 <xsl:apply-templates select="sales" />
6 </xsl:element>
7 </xsl:template>
8 <xsl:template match="/sales">
9 <xsl:variable name="heading" select="summary/heading" />
10 <xsl:variable name="subhead" select="summary/subhead" />
11 <xsl:variable name="description" select="summary/description" />
12 <xsl:variable name="countVal" select="count(data/region)" />
13 <xsl:for-each select="data/region">
14 <xsl:sort select="name" order="descending" data-type="text" />
15 <xsl:element name="mainrecord">
16 <xsl:if test="position() &lt;= 2">
17 <xsl:element name="record">
18 <xsl:attribute name="name">
19 <xsl:value-of select="name" />
20 </xsl:attribute>
21 <xsl:attribute name="heading">
22 <xsl:value-of select="$heading" />
23 </xsl:attribute>
24 <xsl:attribute name="subhead">
25 <xsl:value-of select="$subhead" />
26 </xsl:attribute>
27 <xsl:attribute name="description">
28 <xsl:value-of select="$description" />
29 </xsl:attribute>
30 </xsl:element>
31 <xsl:element name="subrecord">
32 <xsl:apply-templates select="quarter">
33 <xsl:sort select="@number" order="descending" data-type="text" />
34 <xsl:with-param name="vname1" select="count(quarter)" />
35 </xsl:apply-templates>
36 </xsl:element>
37 </xsl:if>
38 </xsl:element>
39 </xsl:for-each>
40 </xsl:template>
41 <xsl:template match="/sales/data/region/quarter">
42 <xsl:param name="vname1" />
```

```

43 <xsl:element name="record">
44 <xsl:attribute name="books_sold">
45 <xsl:value-of select="@books_sold" />
46 </xsl:attribute>
47 <xsl:attribute name="number">
48 <xsl:value-of select="@number" />
49 </xsl:attribute>
50 </xsl:element>
51 </xsl:template>
52 </xsl:stylesheet>

```

1 라인은 XML 문서 선언부입니다. XSLT 스타일시트 파일도 XML 문서의 예입니다.

2 라인은 스타일시트 네임스페이스 선언부입니다.

3-7 라인은 기본 템플릿입니다. 5 라인에 **sales** 는 루트요소입니다.

8-40 라인은 주 쿼리 템플릿입니다.

8 라인의 **/sales** 는 주 쿼리의 최상위 요소입니다.

13 라인의 **data/region** 은 주 쿼리의 반복요소입니다.

14 라인은 **name** 요소로 정렬함을 나타냅니다.(정렬정의 : 정렬요소의 전체 XPath 는 **/sales/data/region/name** 입니다.)

16 라인은 상위 2 개의 레코드만 가져옴을 나타냅니다.(가져올 총 레코드 수)

주 쿼리에서 가져올 XML 문서내의 요소는 'name', 'heading', 'subhead', 'description' 임을 알 수 있습니다. (필드 정의)

33 라인은 서브 쿼리를 **@number** 요소로 정렬함을 나타냅니다. (정렬요소의 전체 XPath 는 **/sales/data/region/quarter/name** 입니다.)

41-51 라인은 서브 쿼리(1 페이지) 템플릿입니다.

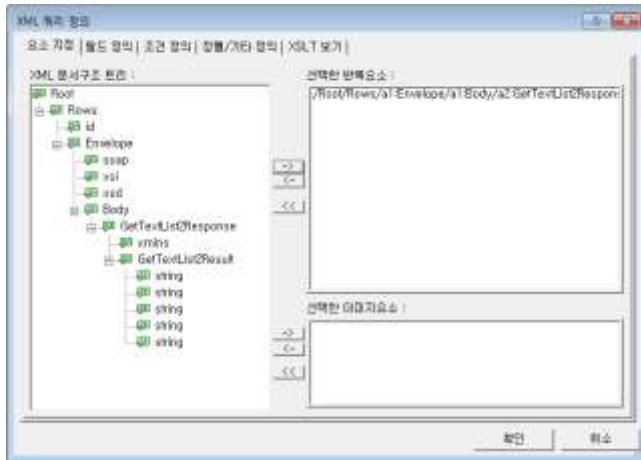
41 라인의 **/sales/data/region/quarter** 는 서브 쿼리의 최상위 요소입니다.

서브 쿼리에서 가져올 XML 문서내의 요소는 'books_sold', 'number' 입니다. (필드 정의)

4.5.9. 웹서비스 연동 보고서인 경우

Data Server 접속방식 보고서 이고 데이터베이스 접속 대화상자에서 서비스 유형을 **webservice** 로 선택 한 경우 XML 쿼리정의 대화상자는 다음 그림과 같습니다.

XML 쿼리정의 대화상자는 웹서비스 설정을 한 이후 사용할 수 있으며 요소 지정 탭이 추가로 제공 됩니다. 요소 지정 탭에서 반복요소, 이미징요소를 지정 한 이후 필드 정의를 합니다.



4.6. 파일 접속

문서에서 사용하는 데이터베이스 접속 방법이 **FILE 접속**이거나 **XMLFILE 접속**일 경우, **[홈]-[데이터소스]-[파일 접속]**을 선택하여, 문서에서 데이터로 사용할 파일의 위치나 파일 안에 들어있는 컬럼개수, 필드 구분자, 각 컬럼의 타입 등을 지정할 수 있습니다.

즉, FILE 접속 문서에서의 파일 접속 기능은 데이터베이스 접속 문서의 **데이터베이스 접속과 쿼리 정의** 기능을 합쳐놓은 것과 같다고 말할 수 있습니다.

4.6.1. 파일 접속 탭

[데이터 소스]-[파일 접속] 대화상자에서 **파일 접속** 탭을 선택하면 데이터로 사용할 파일의 위치와 접속방법을 정의할 수 있습니다. 파일 접속 방법에는 **서버 사용 안함**, **MI 서버 사용**, **FTP 서버 사용**, **HTTP 서버 사용**의 4 가지 방법이 있습니다.

4.6.1.1. 서버 사용 안함



서버를 사용하지 않고 클라이언트에 존재하는 파일에 접속할 경우, 사용하는 옵션입니다. **파일 찾기** 버튼을 통해 해당 데이터 파일을 찾아 선택하거나 직접 파일 경로를 입력할 수 있습니다. 또한, 파일 경로는 양식 파일의 경로를 기준으로 한 **상대 경로** 도 입력할 수도 있습니다.

4.6.1.2. MI 서버 사용



MI 서버 사용은 UNIX 용 Server Module 인 Middleware 를 통해 데이터 파일이 있는 서버와 접속하는 옵션입니다. MI 서버를 사용하려면 **서버 IP** 가 필요합니다.

MI 서버를 사용할 경우에는 MI 서버가 데이터 파일이 있는 서버에 설치가 되어야 하므로 MI 서버의 사용을 위해서는 별도로 구입하여야 합니다.

4.6.1.3. FTP 서버 사용



FTP 서버 사용은 FTP 서버에 접속하여 FTP 서버에 있는 데이터 파일에 접속하는 옵션입니다. FTP 서버를 사용하려면 **서버 IP**, **로그인 ID**, **패스워드** 가 필요합니다.

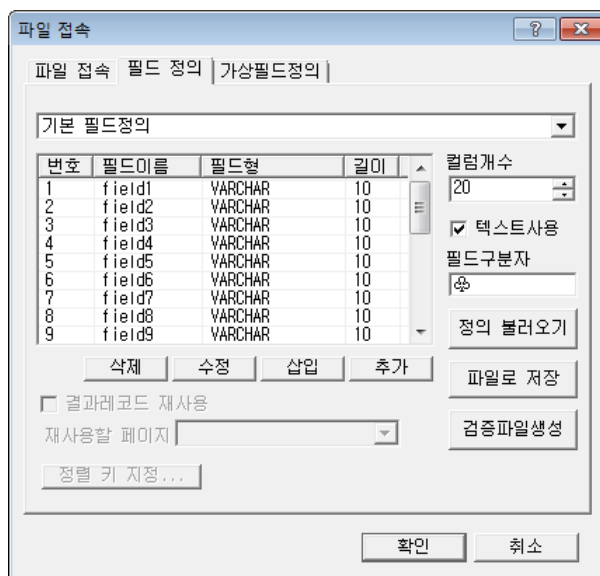
4.6.1.4. HTTP 서버 사용




HTTP 서버 사용은 데이터 파일의 URL 을 입력하여 접속하는 옵션입니다. 이 경우에는 일반적인 텍스트 파일 이외에도 데이터베이스에서 데이터를 추출하는 CGI, Servlet, ASP, JSP 등의 사용도 가능합니다. 단, Servlet 이나 JSP 등을 사용하는 경우에는 그 출력이 **Crownix Report**에서 지원하는 출력이어야 합니다.

4.6.2. 필드 정의 탭

[데이터 소스]-[파일 접속] 대화상자에서 **필드 정의** 탭을 선택하면 데이터 파일 내의 필드에 대한 정의를 할 수 있습니다.




먼저, **필드 정의** 대화상자 상단에서 작성하고자 하는 **필드 정의 및 종류**를 선택합니다. 서브 페이지 문서가 아니라면 **기본 필드 정의**의 만 작성할 수 있으며 서브 페이지 문서인 경우에는 **기본 필드 정의**와 각 페이지의 **서브 페이지 필드 정의**를 작성할 수 있습니다.

 필드명은 숫자로 시작할 수 없으며, 반드시 문자로 시작해야 합니다. 필드명에 대시(-) 문자나 사칙 연산자 등의 일부 특수 문자가 포함되어 있는 경우 요약 작성에서 오류가 발생할 수 있습니다.

4.6.2.1. 컬럼개수

필드 정의의 **컬럼 개수**는 접속한 데이터 파일의 필드 개수를 의미합니다. 데이터 파일의 필드 개수보다 컬럼 개수가 많으면 오류가 발생하고 적으면 나머지 필드는 무시됩니다.

필드 정의의 기본 필드형은 **CHAR** 이며 기본 길이는 **64** 입니다. 물론, 필드형과 길이는 직접 수정이 가능합니다.


 보통 데이터 파일에서는 한 행을 하나의 레코드로 인식합니다. 하지만, 데이터 안에 엔터값이 포함되어 있을 경우에는 필드형을 **VARCHAR** 로 설정하여 사용하면 됩니다.

4.6.2.2. 텍스트 사용 & 필드 구분자

텍스트 사용을 체크하면 텍스트 파일에서 데이터를 가져올 수 있으며 체크를 해제하면 이진 바이너리 파일에서 데이터들을 가져올 수 있습니다.

필드와 필드 사이를 구분하기 위해 텍스트 파일의 경우에는 필드 구분자를 사용할 수 있으며 이진 바이너리 파일의 경우는 필드에 지정된 길이를 가지고 구분합니다. 단, 이진 바이너리 파일의 구성은 **ODBC Data Type** 을 기준으로 합니다.

필드 구분자란 데이터 파일 내의 필드와 필드를 구분해주는 문자(열)로 **텍스트 사용**을 선택해야만 사용할 수 있습니다. 또한 사용자가 임의로 필드 구분자를 지정할 수 있으며 데이터 안에 포함될 수 있는 문자(열)은 사용하면 안됩니다. 만약, 필드 구분자가 지정되어 있지 않으면 필드에 지정된 길이를 가지고 필드를 구분합니다.

 파일 접속 문서에서 차트를 보여줄 때에는 차트에 그려질 값에 해당하는 필드의 필드형이 **숫자형** (NUMERIC, INTERGER 등)으로 되어 있어야 합니다.

4.6.2.3. 필드 정의

□ 삭제

필드 정의 리스트에서 선택된 필드 정의를 삭제합니다.

□ 수정

기본값으로 추가된 필드 정의를 편집하기 위해서는 수정하고자 하는 칸을 마우스 더블 클릭하거나 칸을 선택한 다음 하단의 수정 버튼을 누르면 선택된 칸의 값을 수정할 수 있는 상태로 만들어줍니다.

□ 삽입 / 추가

필드 정의 리스트에서 선택된 필드 정의 앞/뒤로 기본값을 갖는 필드 정의를 삽입/추가 합니다.

4.6.2.4. 파일로 저장

필드 정의 내용을 외부 파일로 저장하는 기능입니다. 텍스트 파일로 저장되며 한 번 정의한 필드 정의 내용을 다른 파일 또는 다른 페이지에서 재사용하고 싶은 경우에 유용합니다.

4.6.2.5. 정의 불러오기

텍스트 파일로 저장된 필드 정의 내용을 불러옵니다. 필드 정의의 **파일로 저장**을 통해 저장한 파일이 아니더라도 필드 정의 형식에 맞는 텍스트 파일이면 필드 정의로 불러 사용할 수 있습니다.

4.6.2.6. 결과 레코드 재사용

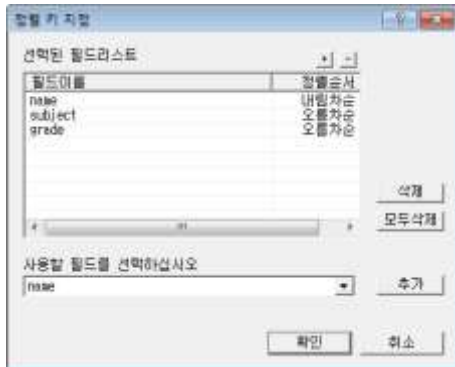
서브 페이지 보고서에서 특정 서브 페이지의 결과 레코드를 다른 서브 페이지에서 재사용하고 싶은 경우에 사용하는 기능입니다. 결과 레코드를 재사용하고자 하는 서브 페이지에서 **결과 레코드 재사용** 체크박스에 체크합니다. 그리고 결과 레코드를 재사용할 다른 서브 페이지 필드 정의에서는 **재사용할 페이지** 콤보 박스를 통해 **결과 레코드 재사용**을 체크했던 서브 페이지 필드 정의를 선택합니다.



4.6.2.7. 정렬키 지정

정렬키 지정 기능을 이용하면 데이터를 정렬 한 후 보고서를 실행 할 수 있습니다. 한 개 이상의 필드를 정렬키로 지정 할 수 있으며 `/rcsvsortex` 파라미터를 사용 하여 보고서 실행 해야 합니다. 첫 글자가 전각 공백, 반각공백, NULL 인 경우, 반각공백으로 비교하며 이후 문자열은 무시됩니다.

단, 서브 페이지 사용 표 문서에서는 사용 할 수 없습니다.



□ 추가

필드 이름, 가상 필드 이름 리스트에서 사용할 필드를 선택한 후 추가 버튼을 클릭 합니다.

□ 삭제

삭제 할 필드 이름을 선택한 후 삭제 버튼을 클릭 합니다.

□ 모두삭제

선택된 필드 리스트의 모든 필드를 삭제 하려면 모두삭제 버튼을 클릭 합니다.

□ 위치이동

선택된 필드 리스트에서 위치 이동할 필드를 선택후 (+/-)버튼을 클릭 합니다.

4.6.2.8. 검증파일생성

입력한 필드 정의 정보를 이용하여 미리 보기에 사용 할 샘플 데이터파일을 생성해 줍니다.

4.6.3. 가상 필드 정의 탭

[데이터 소스]-[파일 접속] 대화상자에서 가상 필드 정의 탭을 선택하면 데이터 파일 내의 가상 필드에 대한 정의를 할 수 있습니다. 단, 서브 페이지 사용 표 문서에서는 사용 할 수 없습니다.

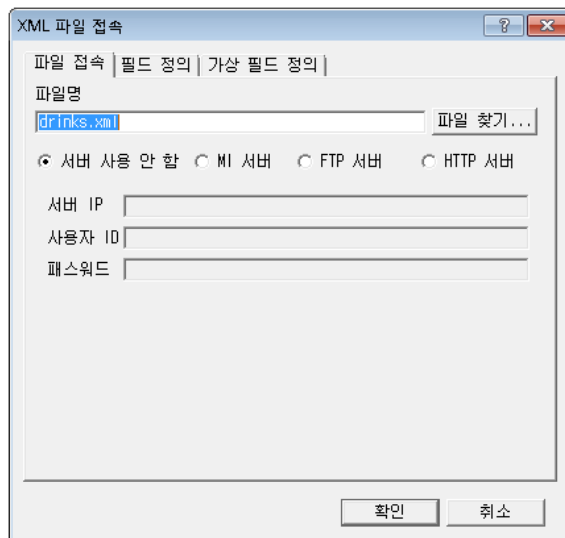
보고서 실행 시에 가상 필드값은 데이터 파일에서 가져 오는 것이 아니라 보고서 실행 과정에서 만들어진 가상 데이터를 이용 합니다. 요약식 수행 결과를 가상 필드의 값으로 이용하게 되며 사용가능한 요약 함수는 제한적 입니다.



대화상자 각 기능 설명은 필드 정의 탭 설명을 참고 합니다.

4.6.4. XML 파일 접속 대화상자 사용하기


XMLFILE 접속문서에서 [데이터 소스]-[파일 접속]을 선택하면 아래의 그림과 같은 XML 파일 접속 대화상자가 나타납니다. XML 파일 접속에서는 데이터로 사용할 XML 문서 파일을 지정하며 사용방법은 텍스트 파일 접속과 같습니다.



□ 텍스트 파일 접속과 XML 파일 접속 비교

텍스트 파일 접속에서는 데이터를 포함하고 있는 텍스트 파일의 위치를 지정하고 필드 정의를 하지만, XML 파일 접속은 데이터를 포함하고 있는 XML 파일의 위치를 지정하고 필드 정의를 합니다.

필드 정의 탭의 필드 리스트에는 XML 쿼리 정의에서 정의한 필드가 자동으로 추가됩니다. 필드 타입에 대한 수정 기능 이외에 필드의 삭제, 수정, 삽입, 추가 등은 XML 쿼리 정의 대화상자를 이용하셔야 합니다.

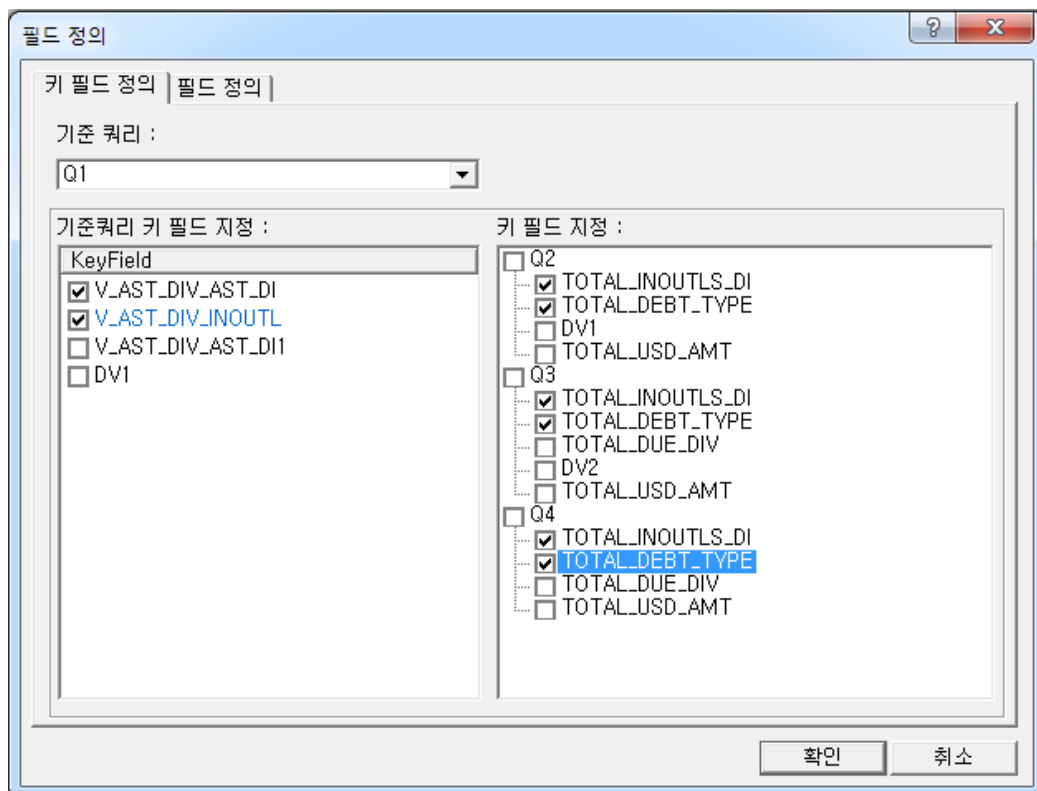
 Data Server 접속방식 보고서 이고 데이터베이스 접속 대화상자에서 서비스 유형을 webservice 로 선택한 경우에도 XML 파일접속 대화상자를 이용 하게 됩니다.

4.7. Multi-Pass SQL 필드 정의

문서에서 사용하는 데이터베이스 접속 방법이 **Data Server** 접속이고 **Multi-Pass SQL** 일 경우, **[홈]-[데이터소스]-[필드정의]**를 선택하여 키 필드 정의와 필드 정의를 할 수 있습니다.

4.7.1. 키 필드 정의 탭

[데이터 소스]-[필드정의] 대화상자에서 키 필드 정의 탭을 선택하면 기준 쿼리 키 필드를 지정할 수 있고 기준 쿼리 키 필드와 연관된 키 필드를 지정할 수 있습니다.




4.7.1.1. 기준 쿼리 선택

기준 쿼리로 사용할 쿼리를 선택합니다.

4.7.1.2. 기준 쿼리 키 필드 지정

기준 쿼리의 키 필드로 사용할 필드를 선택합니다.

 기준쿼리 키 필드는 1 개 이상 지정 해야 합니다.

4.7.1.3. 키 필드 지정

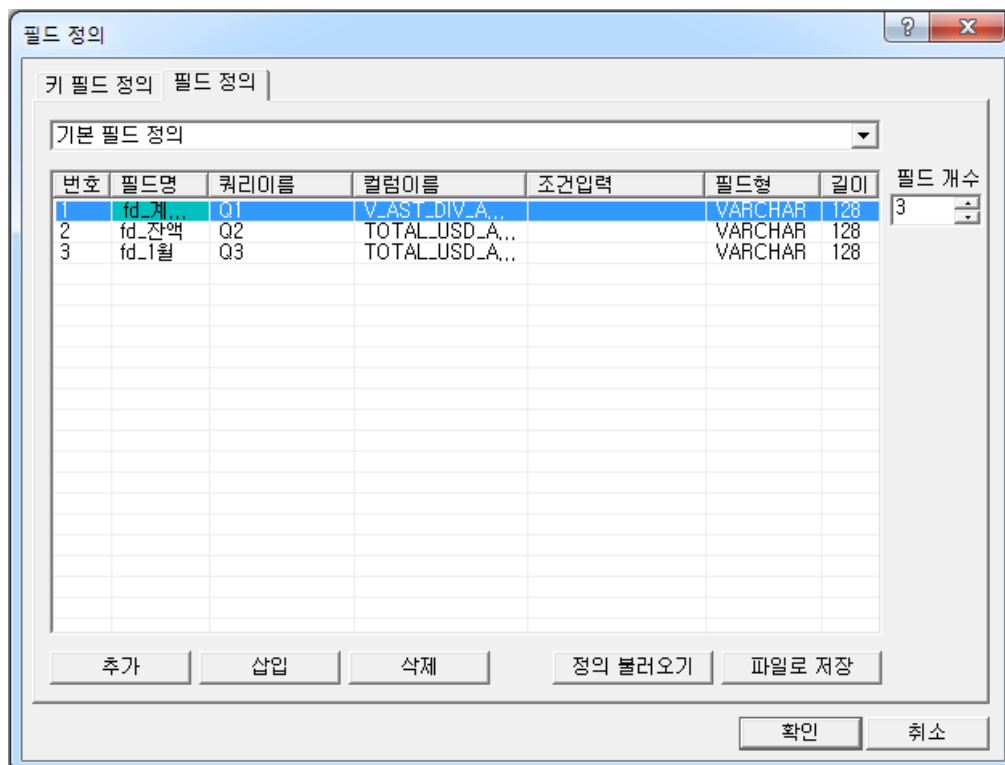
기준 쿼리 키 필드와 연관 된 키 필드를 지정 합니다.

 기준 쿼리 키 필드의 개수와 각 쿼리의 키 필드 개수는 동일 해야 합니다.

4.7.2. 필드 정의 탭

[데이터 소스]-[필드정의] 대화상자에서 필드정의 탭을 선택하면 필드 정의를 할 수 있습니다.

단, 키 필드 정의 탭에서 키 필드 정의를 한 이후에 필드 정의를 합니다.



4.7.2.1. 필드 개수

필드 정의로 사용할 **필드 개수**를 입력한 후 필드 정의 리스트를 선택 합니다. 필드 개수로 입력한 개수 만큼 기본 필드가 생성됩니다.

필드 정의의 기본 필드형은 **VARCHAR** 이며 기본 길이는 **128** 입니다.

4.7.2.2. 필드 정의

□ 추가

필드 정의 리스트의 마지막에 새 필드를 추가 합니다.

□ 삽입

필드 정의 리스트의 선택된 항목 앞에 새 필드를 삽입 합니다.

□ 삭제

필드 정의 리스트에서 선택된 항목을 삭제 합니다.

□ 수정

필드 정의를 편집하기 위해서는 수정하고자 하는 칸을 마우스 더블 클릭하거나 칸을 선택하면 선택된 칸의 값을 수정할 수 있는 상태로 만들어 줍니다.

필드명, 조건입력, 길이는 입력 수정이 가능합니다. 쿼리이름, 컬럼이름, 필드형은 **ComboBox**에서 선택을 변경 할 수 있습니다.

□ 복사

복사 할 항목을 선택 한 후에 마우스 오른쪽 클릭을 하면 복사/붙여넣기 팝업메뉴가 표시되며 복사를 선택 합니다.

□ 붙여넣기

복사한 항목이 있는 경우, 붙여넣기 하고자 하는 항목을 선택 한 후에 마우스 오른쪽 클릭을 하면 복사/붙여넣기 팝업메뉴가 표시되며 **붙여넣기**를 선택 합니다.

선택한 항목 이후에 복사한 항목이 붙여넣기 됩니다.

4.7.2.3. 파일로 저장

필드 정의 내용을 외부 파일로 저장하는 기능입니다. 텍스트 파일로 저장되며 한번 정의한 필드 정의 내용을 다른 파일 또는 다른 페이지에서 재사용하고 싶은 경우에 유용합니다.

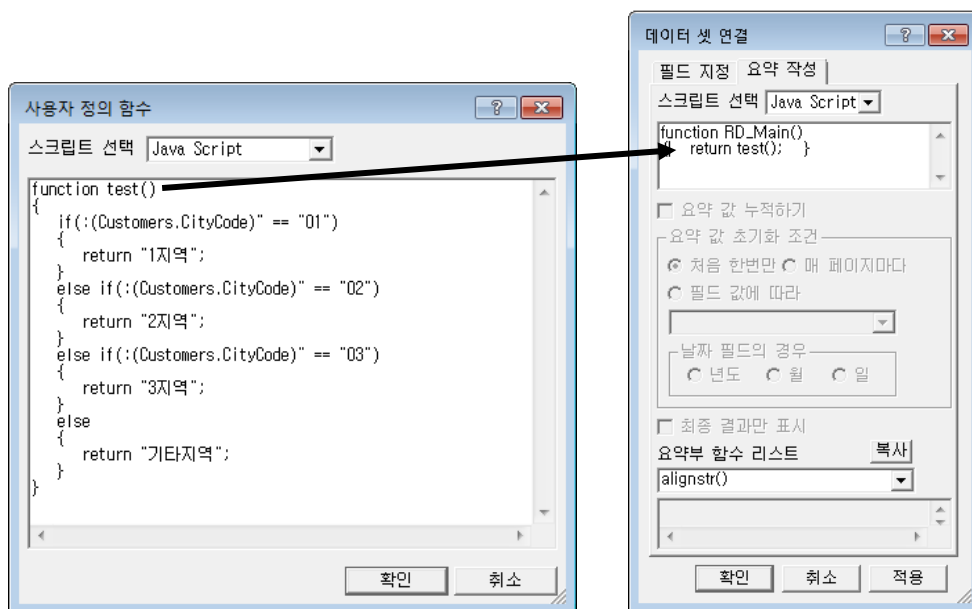
4.7.2.4. 정의 불러오기

텍스트 파일로 저장된 필드 정의 내용을 불러옵니다. 필드 정의의 **파일로 저장**을 통해 저장한

파일이 아니더라도 필드 정의 형식에 맞는 텍스트 파일이면 필드 정의로 불러 사용할 수 있습니다.

4.8. 사용자 정의 함수

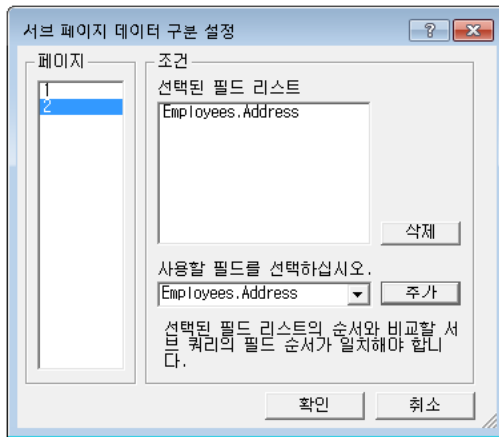
[데이터 소스]-[사용자 정의 함수] 기능을 사용하면 VB Script 나 Java Script 의 형태로 사용자가 직접 함수들을 정의할 수 있습니다. 정의한 함수들은 요약문에서 호출하여 사용할 수 있습니다.



4.9. 서브 페이지 데이터 설정

기존의 주 쿼리-서브 쿼리 구조에서는 주 쿼리의 건수만큼 서브 쿼리가 반복적으로 실행되기 때문에 전체 쿼리 수행시간이 길어지는 단점이 있었습니다. 하지만, 단순히 서브 쿼리의 조건절에 주 쿼리의 필드 값만이 들어가는 경우라면 **서브 페이지 데이터 설정** 기능을 이용하여, 실제로 서브 쿼리는 한 번만 실행한 후, 그 결과를 주 쿼리의 필드 값대로 분할하여 보고서를 작성할 수 있습니다.

[데이터 소스]-[서브 페이지 데이터 설정]을 선택하면 아래의 그림과 같은 대화상자가 나타납니다. 단, 표 문서에서 **SUBPAGE** 사용을 체크한 경우에만 메뉴가 표시됩니다.



대화상자를 통해 서브 페이지 데이터 설정 기능을 사용할 서브 페이지를 선택하고 조건절로 구분할 주 쿼리 필드 값을 선택한 후, 추가 버튼을 누릅니다.

단, 서브 페이지 데이터는 주 쿼리 필드값과 비교되는 서브 쿼리 필드명으로 정렬되어 있어야 합니다. 또한, 비교되는 필드명이 두 개 이상인 경우에는 대화상자에서 선택한 주 쿼리 필드명의 순서와 서브 쿼리의 조건절에 사용하는 필드명의 순서가 반드시 일치해야 합니다.

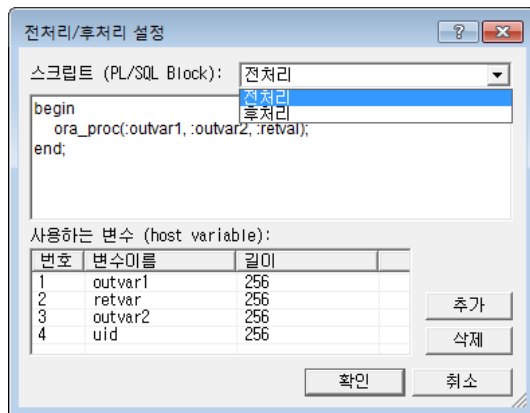
예를 들어, 서브 쿼리에서 주 쿼리 필드인 **Employees.EmployeeID** 값을 참조하여 보고서를 작성해야 한다면 **WHERE 절**에 주 쿼리 필드명을 포함시켜서 작성해야 합니다. 하지만, 서브 페이지 데이터 설정 기능을 사용하면 **WHERE 절**과 **Order by 절**을 아래의 그림과 같이 작성한 후, 서브 페이지 데이터 설정 대화상자에서 해당 페이지에 참조할 주 쿼리 필드명인 **Employees.EmployeeID** 값을 추가해주기만 하면 됩니다. 이렇게 함으로써, 서브 쿼리는 최초 한 번만 실행되게 되고 전체 쿼리 수행시간은 줄어들게 됩니다.



4.10. 전처리/후처리 지정

이 기능은 ORACLE 에 직접 접속하는 경우나 **Crownix ERS Data Server** 를 사용해서 ORACLE 에 접속하는 경우에 사용할 수 있습니다. ORACLE 의 PL/SQL Block 을 등록해서 리포팅을 시작하기 전과 리포팅이 끝난 후에 실행시킬 수 있습니다. 즉, **전처리**는 리포팅을 시작하기 전에 실행하는 PL/SQL Block 을 설정하는 것이고 **후처리**는 리포팅이 끝난 후, 실행하는 PL/SQL Block 을 설정하는 것입니다. 전처리를 실행해서 넘겨받은 결과값은 리포팅에 사용하는 쿼리문에서도 사용할 수 있습니다.

[데이터 소스]-[전처리/후처리 지정]을 선택하면 아래의 그림과 같은 대화상자가 나타납니다.

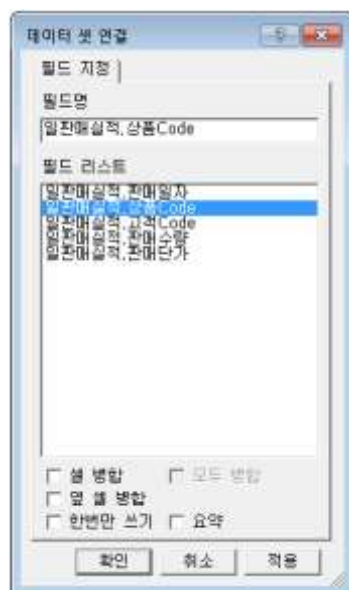


PL/SQL Block 을 작성하기 전에 전처리인지 후처리인지를 먼저 선택합니다. 그리고 추가 버튼을 눌러 PL/SQL Block 에서 사용하는 호스트 변수를 추가합니다. 추가한 변수를 스크립트에서 사용할 때는 반드시 콜론(:)을 앞에 적어주어야 합니다. 변수명 및 길이는 리스트의 해당 항목을 마우스 왼쪽버튼으로 더블클릭하여 직접 수정할 수 있습니다. 단, 변수의 형은 **CHAR** 형으로 제한합니다.

4.7. 데이터 셋 연결

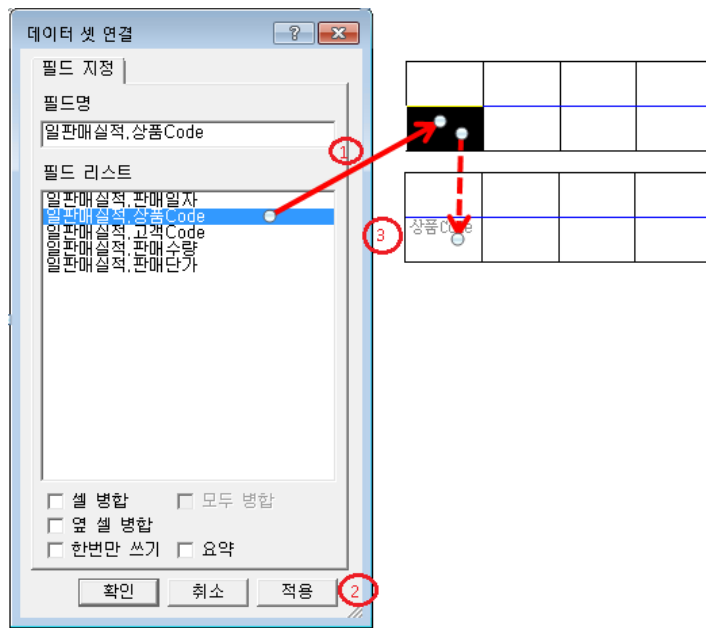
데이터베이스 접속 방법에 관계없이 데이터베이스 접속과 쿼리 정의 또는 파일 접속 등의 절차가 완료되면 데이터 셋 연결 기능을 이용하여 보고서에 그려진 각 표의 셀이나 텍스트 상자 등에 데이터값을 연결시킬 수 있습니다.

[데이터 소스]-[데이터 셋 연결]을 선택하거나 단축키 <Ctrl+W>를 누르면 아래의 그림과 같은 대화상자가 나타납니다.



대화상자를 띄운 상태에서 데이터를 출력할 표의 셀이나 텍스트 상자 등을 마우스로 선택하고 대화상자의 필드 리스트에서 연결할 필드를 지정한 다음, **적용** 버튼을 누르면 선택한 필드가 해당 객체에 연결됩니다. 연결된 내용은 화면상에 희미한 글씨로 표시됩니다.

참고로 연결된 내용을 수정하거나 삭제하고자 할 경우에는 마찬가지로 데이터 셋 연결 대화상자를 표시한 상태에서 해당 객체를 선택한 후, 필드명에 표시된 내용을 수정하거나 삭제하면 됩니다.



또는, 표의 셀을 여러 개 블록지정하고 필드 리스트에서 같은 수의 필드를 한꺼번에 선택한 다음, 적용 버튼을 누르면 각 필드의 순서에 따라 표의 선택한 셀들에 그대로 연결됩니다.

데이터 셋 연결 대화상자는 각 옵션에 따라 **필드 지정**, **요약 조건**, **요약 작성**, **참조 필드** 등 최대 4 개의 탭을 사용할 수 있습니다.



4.11.1. 필드 지정 탭

데이터 셋 연결의 가장 기본적인 기능으로 필드나 변수를 객체나 표와 연결시키는 기능이며 셀 병합, 모두 병합, 옆 셀 병합, 한 번만 쓰기, 요약과 같은 옵션을 설정할 수 있습니다.

□ 필드명

대화상자를 띄운 상태에서 필드가 연결되어 있는 표의 셀이나 텍스트 상자 등을 선택했을 경우, 해당 객체에 연결되어 있는 필드명을 보여줍니다. 경우에 따라 필드 리스트에 없는 임의의 변수명(/rp 또는 /rv 파라미터 변수명)을 객체에 적용하고 싶은 경우에는 해당 객체를 선택한 후, 임의의 변수명을 사용자가 직접 입력하여 적용할 수 있습니다.



/rp 나 /rv 파라미터에 대한 자세한 설명은, 부록 3 Crownix Report 파라미터에서 3.1.159. /rp 나 3.1.317. /rv 의 내용을 참조하십시오.

□ 필드 리스트

쿼리 정의와 필드 쿼리 정의 또는 파일 접속의 필드 정의 등을 통해 정의된 필드명들이 나열됩니다.

□ 한 번만 쓰기

동일한 값의 데이터가 연속해서 나올 때, 첫 번째 데이터만 출력하고 다음 레코드부터는 그 필드 값이 바뀔 때까지 빈 행으로 나타나도록 합니다. 반복절이나 요약문에서 모두 사용할 수 있고 참조 필드 탭을 사용할 수 있습니다.

□ 셀 병합

한 번만 쓰기 처럼 동일한 값의 데이터가 연속해서 나올 때, 표의 셀을 합치도록 하는 옵션입니다. 셀이 합쳐지므로 데이터는 자동으로 한 번만 써집니다. 반복절이나 요약문에서 모두 사용할 수 있고 참조 필드 탭을 사용할 수 있습니다.

□ 모두 병합

셀 병합 옵션에 의해 활성화되는 기능입니다. 모두 병합 옵션은 페이지 첫 행 데이터의 값으로 빈 행이 그려지는 부분까지 전부 셀 병합을 수행합니다.

□ 옆 셀 병합

동일한 데이터가 옆 셀에 연속하여 나올 때, 하나의 셀로 결합하는 기능입니다. 옆 셀 병합 옵션은 병합할 셀들에 모두 설정해 주어야 합니다.

‘셀 병합/옆 셀 병합/모두 병합’ 를 함께 설정한 경우, ‘셀 병합/옆 셀 병합’ 과 동일하게 동작합니다.

원래 필드	원래 필드	원래 필드	원래 필드	원래 필드	원래 필드	원래 필드
18000202	18000202	18000202	18000202	18000202	18000202	18000202
18000202				18000202		
18000202				18000202		
18000202				18000202		
18000202				18000202		
18000202				18000202		
18000202				18000202		

빈 행 삽입 한 번만 쓰기 셀 병합 셀 병합+모두 병합 옆 셀 병합 셀 병합+옆 셀 병합 셀 병합+옆 셀 병합
+모두 병합

원래 필드	원래 필드	원래 필드	원래 필드	원래 필드	원래 필드	원래 필드
18000202	18000202	18000202	18000202	18000202	18000202	18000202
18000202				18000202		
18000202				18000202		
18000202				18000202		
18000202				18000202		
18000202				18000202		
18000202				18000202		

빈칸 그리기가 적용된 표를 뷰어에서 본 결과 한 번만 쓰기 옵션들 준 경우 칸 합치기 칸 합치기 + 모두 합치기 옆칸 합치기 칸 합치기 + 옆칸 합치기 칸 합치기 + 옆칸 합치기 + 모두 합치기

□ 요약

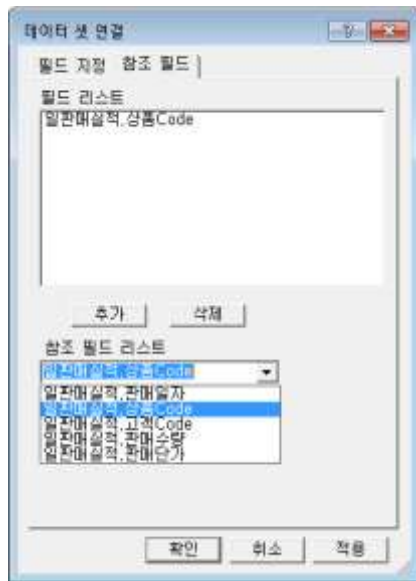
요약부가 아닌 텍스트 상자나 표의 셀에 요약문을 작성하여 적용하고자 할 때, 선택합니다. 요약 설정하면 데이터 셋 연결 대화상자에 **요약 작성 탭**이 추가됩니다.

가져온 레코드에 행 단위 계산식을 적용시키거나 데이터 값의 조건에 따라 데이터 출력 속성을 다르게 주고 싶을 때, 요약 작성에서 적절한 요약문을 작성합니다.

예를 들어, 단가와 수량을 곱해 매출액을 구하는 열을 하나 추가하거나 매출액수에 따라 글씨의 색상을 다르게 하는 등의 요약 작성을 해줄 수 있습니다.

4.11.2. 참조 필드 탭

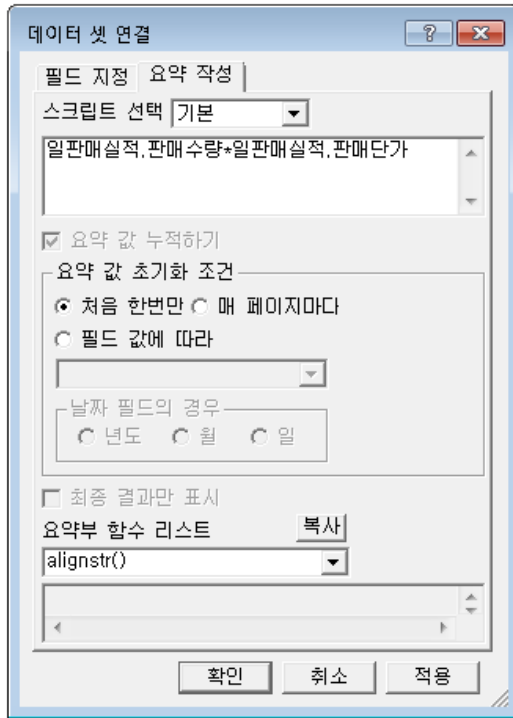
참조 필드 탭은 필드 지정 탭에서 한 번만 쓰거나 셀 병합 기능을 지정하면 나타납니다. 참조 필드 탭의 필드 리스트 목록상자에서 필드를 선택하고 추가 버튼을 누르면 참조 필드가 되며 아래의 그림과 같이 참조 필드 리스트에 추가됩니다.



참조 필드는 한 레코드에서 **한 번만 쓰거나 셀 병합** 옵션을 수행하기 전에 먼저 참조해야 하는 필드입니다. 따라서, **한 번만 쓰거나 셀 병합**이 지정된 필드의 데이터가 연속해서 동일하더라도 **참조 필드**의 데이터가 서로 다르다면 **한 번만 쓰거나 셀 병합**은 적용되지 않습니다. 그리고 참조 필드는 여러 개 선택할 수도 있습니다.

4.11.3. 요약 작성 탭

요약 작성 탭은 입력되는 값에 어떠한 계산을 적용시켜 출력하고자 할 때, +, -, *, / 를 이용한 간단한 사칙연산의 적용이 가능합니다. 또한, 요약 작성 시 제공되는 각종 요약 함수를 이용하여 원하는 계산식들을 입력할 수 있는 기능도 제공합니다.



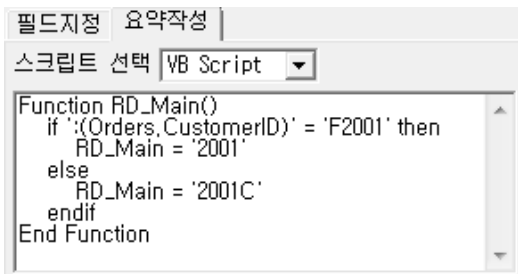
□ 스크립트 선택

Crownix Report 에서는 요약문을 작성할 때 사용하는 RD 요약함수와 요약구문(syntax)을 제공합니다. 하지만 RD 요약구문 외에도 VB Script, Java Script 를 이용하여 요약문을 작성할 수 있습니다. 단, RD 요약구문을 사용하지 않는 경우에는 RD 요약함수도 사용할 수 없습니다.

□ 요약 작성란

+, -, *, / 를 이용한 간단한 사칙연산이나 요약 작성 시 제공되는 각종 요약 함수를 이용하여 입력 받은 필드 값에 대한 연산식을 입력합니다.

VB Script 나 Java Script 를 사용하는 경우에는 해당 Script 의 구문대로 작성할 수 있습니다. 단, 반드시 요약 작성은 RD_Main() 함수 내에서 해야합니다.



□ 요약 값 누적하기


선택한 표의 셀이 반복부인 경우에는 무조건 요약 값이 누적되지만 요약부인 경우에는 **요약 값 누적하기** 체크박스를 선택하여 요약 값 누적 여부를 지정할 수 있습니다. 요약 값 누적하기

가 선택되면 **요약 값 초기화조건**들이 활성화됩니다.

□ 요약 값 초기화조건

요약 값 초기화조건은 요약 작성된 값을 조건에 따라 초기화시킬 수 있도록 합니다.

처음 한 번만, 매 페이지마다, 필드값에 따라 등의 조건을 제공하는데 필드값에 따라 조건은 사용 가능한 필드명을 선택할 수 있고 날짜 필드인 경우에는 년도, 월, 일까지 구분해서 지정할 수 있습니다.

 **요약 값 초기화조건의 필드값에 따라** 필드명을 선택할 때, 두 개 이상의 필드명을 지정할 수도 있습니다. 이때는, 각각의 필드명을 `Orders.CustomerID:Employees.EmployeeID` 와 같이 **콜론(:)**으로 구분해서 사용하면 됩니다. 단, 조건은 선택한 필드 중의 하나라도 바뀌면 초기화되는 **OR 조건**이 됩니다.

□ 최종 결과만 표시

선택한 표의 셀이 **요약부인** 경우에만 사용하는 옵션입니다. 요약부는 기본적으로 요약 작성에서 작성된 연산식의 값을 지정한 조건에 따라 출력하는 특징을 가지고 있습니다.

그러나 대차대조표와 같이 내용이 끝나지 않은 상태에서 합계를 출력하면 안 되는 보고서의 경우, **최종 결과만 표시** 옵션을 선택하면 내부적으로 계산을 하고 있다가 마지막 페이지에서 값을 출력하게 됩니다.

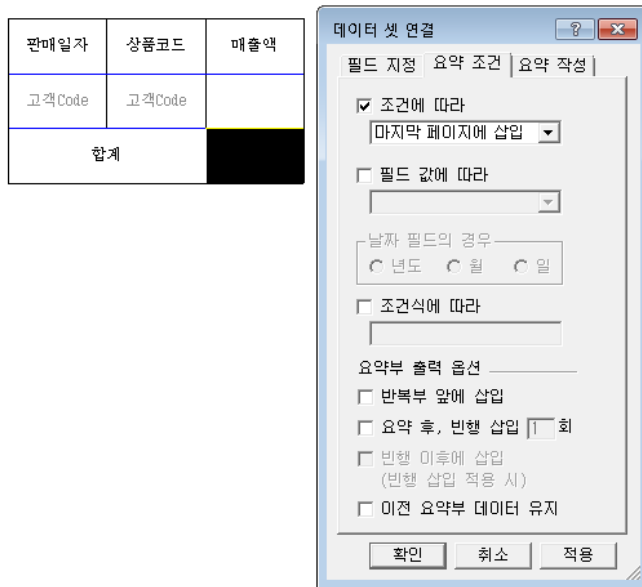
□ 요약문 함수 리스트 및 사용법

요약문 작성을 돕기 위해서 요약문에서 사용 가능한 함수를 보여주고 하단에 그 설명을 보여줍니다. **복사 버튼**은 요약문 함수리스트에서 선택한 함수를 요약 작성부분에 복사해 주는 역할을 합니다.

 **요약함수에 관한 내용은 부록 1. Crownix Report 요약의 내용을 참조하십시오.**

4.11.4. 요약 조건 탭

요약부의 출력 조건을 지정할 수 있는 기능을 제공하며 선택한 표의 셀이 **요약부인** 경우에만 나타나는 탭입니다.



□ 조건에 따라

마지막 페이지에 삽입, 매 페이지마다 삽입, 반복수마다 삽입, 빈 행 삽입 시, 첫번째 행으로 삽입 등의 요약부 출력 옵션을 제공합니다.

빈 행 삽입 시, 첫번째 행으로 삽입 요약부는 빈 행 삽입 수행할 때 나오는 요약부이고 빈 행 삽입 행수가 2 개 이상인 경우 나옵니다. 빈 행 삽입 시, 첫 행에 '이하여백' 문자열을 출력하기 위한 용도입니다.. 빈 행 삽입 시, 첫번째 행으로 삽입 요약부는 1 행만 추가 할 수 있으며 다른 요약 조건을 함께 사용할 수 없고 크로스탭 사용 보고서는 지원 하지 않습니다.

□ 필드 값에 따라

특정 필드의 값이 변할 때마다 요약부를 출력하는 조건입니다. 필드 지정의 필드 리스트가 나타나며 그 중 하나를 선택하면 됩니다.

선택된 필드가 날짜 필드인 경우에는 년도,월,일 까지 구분해서 지정할 수 있습니다.

두 개 이상의 필드를 지정하는 경우에는 콜론(:) 문자를 이용합니다.


□ 조건식에 따라

지정한 조건식이 TRUE 가 되면 요약부가 출력 됩니다. 요약부 출력 조건식은 그 결과가 항상 TRUE 또는 FALSE 이어야 하며 요약부가 나오고 나면 바로 초기화를 시킵니다. 단, CrossTab 표 보고서에는 사용할 수 없습니다.

□ 반복부 앞에 삽입

일반적으로 표의 요약부는 반복부의 다음 행에 추가됩니다. '반복부 앞에 삽입' 옵션을 사용하여 요약부를 반복부의 첫 행으로 삽입할 수 있습니다.

크로스탭 문서에서 ‘반복부 앞에 삽입’ 기능을 적용하고 싶은 경우, 요약 조건은 ‘매 페이지마다 삽입’으로 지정되어야 합니다.


 반복부 앞에 삽입되는 요약부는 텍스트 상자 속성의 “셀 크기 고정” 옵션이 체크되어 있어야 합니다. 크기 고정 되어 있지 않거나 “메모 객체” 옵션을 사용할 경우 보고서가 페이지 영역을 벗어날 수 있습니다.

□ 요약 후, 빈 행 삽입

보고서의 형태를 보기 좋게 하기 위해 요약부를 출력한 후, 지정한 횟수만큼 빈 행을 추가합니다.

□ 빈 행 이후에 삽입

표-표 속성지정에서 빈 행 삽입 옵션을 해당 표에 지정했을 때, 적용할 수 있습니다. 요약부 출력 조건에 의해 해당 요약부가 출력될 때, 빈 행 삽입 옵션이 적용되어 있으면 빈 행을 먼저 출력하고 다음에 요약부를 출력합니다.

 빈 행 삽입 기능에 대한 자세한 내용은 8.3 표 속성의 내용을 참조하십시오.

□ 이전 요약부 데이터 유지

표에 매 페이지마다 출력하는 요약부와 마지막 페이지에 출력하는 요약부가 있는 경우, 만약 페이지의 끝부분에서 반복부 처리가 끝나면 마지막 페이지에 출력하는 요약부는 다음 페이지에 출력되게 됩니다. 그리고 페이지가 새로 생겼으므로 매 페이지마다 출력하는 요약부도 나오게 되는데, 새로 생긴 페이지에는 반복부 행이 없기 때문에 매 페이지마다 출력하는 요약부에는 데이터 매핑이 되지 않습니다. 이런 경우에 **이전 요약부 데이터 유지** 기능을 사용하면 요약부가 이전 요약부 데이터를 유지하도록 해서 데이터를 표시할 수 있습니다.

참고로 이 기능은 이전 버전까지는 요약 조건 조건식에 따라 “KEEPDATA” 라는 예약어를 사용하여 제공이 되었던 기능이며, **이전 요약부 데이터 유지** 체크 대신 **KEEPDATA** 를 입력해도 자동으로 이전 요약부 데이터 유지에 체크가 되어 이전과 동일하게 동작합니다.

4.12. 스크립트

스크립트 작성은 문서의 형태가 스크립트 문서이거나 일반, 표, 라벨 문서인 경우라도 스크립트 사용을 선택한 경우에만 사용할 수 있습니다.

[데이터 소스]-[스크립트]를 선택하거나 단축키 <Ctrl+S>를 누르면 아래의 그림과 같은 대화상자가 나타납니다.



4.12.1. 변수명

변수명 입력란은 선택된 객체에 이름(변수명)을 부여하는 곳입니다. 반드시 객체를 먼저 선택해야만 변수명 입력란에 객체 변수명을 입력할 수 있고 그 객체 변수명은 파일 내에서 유일해야만 합니다. 만약 객체 변수명이 중복될 경우에는 오류 메시지가 나타납니다.

4.12.2. 변수형

변수형 선택란은 객체 변수를 통해 출력될 데이터의 변수형을 지정하는 곳으로 숫자, 문자, 날짜, 시간이 있습니다. 문서 내의 모든 객체는 변수명을 가질 수 있지만 변수형을 가질 수 있는 객체는 **텍스트 상자**와 **표의 셀**로 제한됩니다.

4.12.3. 변수 리스트

변수 리스트 목록상자는 현재 작업 중인 문서에서 사용하는 모든 객체의 변수명이 담겨 있는 목록상자입니다.

변수 리스트 목록상자에서 임의의 변수명을 선택하면 선택된 변수명이 부여된 객체가 있는 페이지의 해당 객체로 이동됩니다. 선택된 변수명을 갖는 객체는 네 모서리에 작은 사각형이 표시되거나 표의 셀인 경우는 반전된 상태로 표시됩니다. 이 때, 변수명 입력란에는 선택된 객체의 변수명이 보여지고 변수형 입력란에는 그 객체의 변수형이 표시됩니다.

4.12.4. 스크립트 편집 창

스크립트 편집 창은 문서의 출력을 위한 스크립트를 입력, 편집할 수 있는 곳입니다.

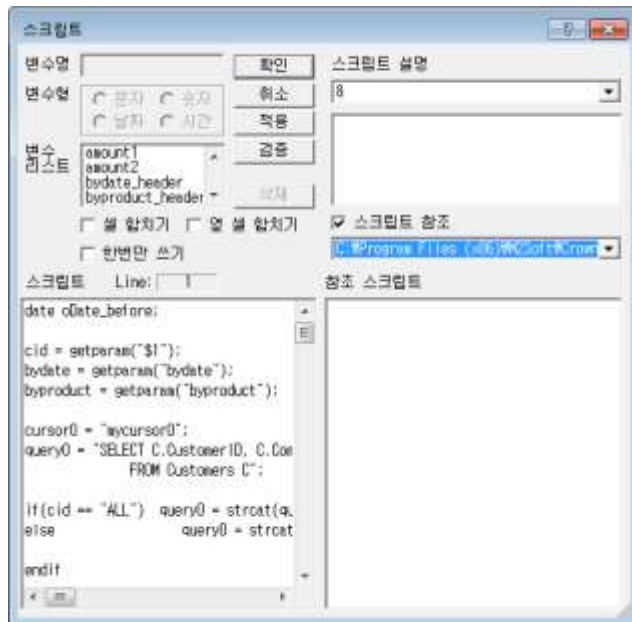
스크립트 편집 창에서는 스크립트를 편집하는데 편리함을 제공하기 위해 **블록 단위의 복사 (<Ctrl+Ins>, <Ctrl+C>), 잘라내기 (<Ctrl+Del>, <Ctrl+X>), 붙이기 (<Shift+Ins>, <Ctrl+V>)** 기능과 **문자열 찾기/바꾸기(<Ctrl+F>)** 기능 등을 제공합니다.

4.12.5. 스크립트 설명

Crownix Report 스크립트에서 제공하는 각종 스크립트 함수에 대한 설명을 확인할 수 있습니다. 전체 스크립트 함수들에 대한 설명은 **부록 2 Crownix Report 스크립트**의 내용을 참조하십시오.

4.12.6. 스크립트 참조

스크립트 참조 기능은 다른 파일에서 작성한 스크립트의 내용을 참조하고자 할 때, 사용하는 기능입니다. 스크립트 참조 체크 박스를 선택하면 아래의 그림과 같이 **참조 파일 목록**이 활성화되고 **참조 스크립트** 창이 생깁니다.



4.12.6.1. 참조파일

참조파일 목록상자는 현재 작업 중인 파일과 관련된 다른 파일의 스크립트를 참조하기 위해

살펴보는 것으로 **참조할 파일**을 선택하면 선택한 파일명이 목록상자에 표시됩니다.

4.12.6.2. 참조 스크립트

참조 스크립트 창은 **참조파일** 목록상자에서 선택된 파일 안에 기술된 **스크립트**를 보여줍니다. 참조 스크립트 창에 보여지는 스크립트는 참조만 가능하고 편집은 할 수 없습니다.

4.12.7. 버튼들의 기능

□ 확인

확인 버튼은 **검증** 버튼을 눌렀을 때와 같이 스크립트의 문법을 검사하면서 스크립트 입력 대화상자를 닫고 문서 편집 상태로 돌아갑니다.

□ 취소

취소 버튼은 스크립트를 편집 전의 상태로 되돌리는 기능을 수행합니다. 즉, 스크립트 대화상자를 연 후에 스크립트를 편집하였다면 수정된 내용을 스크립트 대화상자를 열었을 때의 처음 상태로 되돌리는 기능입니다. 그러나 스크립트를 수정한 후에 확인이나 적용, 검증 버튼을 눌렀거나 스크립트 대화상자를 닫았다가 다시 열었다면 취소 버튼을 이용한 되돌리기 기능을 사용할 수 없게 됩니다.

□ 적용

적용 버튼은 수정된 스크립트 내용을 작업 중인 문서에 적용시키는 기능을 수행합니다. 적용 버튼을 누르게 되면 스크립트의 수정된 내용이 저장되어 취소버튼을 이용한 되돌리기 기능을 수행할 수 없게 되므로 주의해야 합니다.

□ 검증

검증 버튼은 현재의 파일에 기술되어 있는 스크립트의 문법을 검사합니다. 기술된 스크립트에 문법 오류가 발생하면 발생한 위치를 가르쳐 줍니다.

□ 삭제

삭제 버튼은 현재 선택된 객체에 부여된 변수명을 삭제하는 기능입니다. 또한 변수선택 목록상자에 있는 변수명도 같이 삭제됩니다.



스크립트 대화상자에 있는 변수명이 삭제되었다고 편집 화면에 있는 객체까지 삭제 되는 것은 아닙니다.

□ 셀 병합

셀 병합은 반복지정된 행의 셀에 동일한 내용이 반복해서 나올 때, 반복해서 여러 번 출력하지 않고 셀을 합친 뒤 한 번만 출력되게 해주는 기능입니다. 셀 병합을 적용시키기 위해선 문서 내에서 해당 셀을 선택하거나 셀에 부여된 객체 변수명을 선택한 다음 셀 병합을 선택하면 됩

니다.

□ 옆 셀 병합

옆 셀 병합은 반복 지정된 행의 옆 셀에 동일한 내용이 나올 때, 셀을 옆으로 합친 뒤 한 번만 출력되게 해주는 기능입니다. 옆 셀 병합은 셀을 합칠 두 셀에 모두 설정해주어야 합니다.

□ 한 번만 쓰기

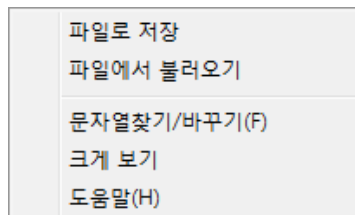
한 번만 쓰기는 같은 내용을 반복적으로 출력하지 않고 제일 위에 한 번만 출력하게 하는 기능입니다. 적용시키는 방법은 셀 병합 기능과 동일합니다.



셀 병합과 옆 셀 병합, 한 번만 쓰기는 표에 반복지정이 되어있을 때만 수행이 가능합니다.

4.12.8. 스크립트 편집 창의 팝업메뉴

스크립트 편집 창 내에서 마우스 오른쪽 버튼을 누르면 다음과 같은 팝업메뉴가 보입니다.



□ 파일로 저장

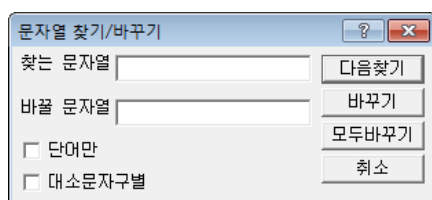
파일로 저장 메뉴는 기술되어 있는 스크립트를 스크립트 파일이나 텍스트 파일로 저장하는 기능을 수행합니다.

□ 파일에서 불러오기

파일에서 불러오기 메뉴는 저장된 스크립트 파일이나 외부의 텍스트 편집기에서 편집한 텍스트 파일을 스크립트 대화상자로 불러오는 기능을 수행합니다.

□ 문자열 찾기/바꾸기

문자열 찾기/바꾸기 메뉴는 저장된 스크립트 파일이나 외부의 텍스트 편집기에서 편집한 텍스트 파일에서 특정문자열을 찾거나 바꾸는 작업을 수행합니다.



스크립트를 편집하는 중에 <Ctrl+F>을 눌러도 동일하게 **문자열 찾기/바꾸기** 대화상자가 나타납니다.

단어만 체크박스를 선택하면 입력한 문자열로만 구성된 단어를 찾고 **대소문자구별** 체크박스를 선택하면 대소문자가 모두 일치하는 문자열을 찾습니다

문자열을 찾는 위치는 **현재 편집 창에서 깜빡 거리고 있는 입력 커서 위치에서부터** 찾게 되며 맨 끝까지 커서가 다 이동했을 경우 처음부터 다시 찾습니다.

□ 크게 보기

크게 보기 메뉴는 스크립트 대화상자에서 스크립트 편집창의 크기를 가능한 한 크게 합니다. 이 기능을 사용하게 되면 대화상자에 변수명이 보이는 부분과 스크립트 편집 창만이 보이게 됩니다. 다시 원래의 상태로 돌아가고 싶은 경우에는 팝업 메뉴를 띄워 크게 보기 메뉴를 다시 한 번 누르면 됩니다.

4.13. XPlatform DataSet 접속

데이터베이스 접속 방법이 **XPLATFORM_DS** 이면 다음과 같이 동작 합니다.

보고서 디자인 타임 : XPlatform DataSet XML 파일을 이용 하여, 보고서 실행에 필요한 쿼리 정의를 합니다. 또한 DataSet XML 파일을 이용 한 미리 보기 기능을 제공 합니다.

보고서 실행 타임 : XPlatform Runtime 이 RDWrapper 를 이용 하여 넘겨 준 DataSet 객체를 이용 하여, 보고서를 생성 합니다. 이때, 보고서 디자인 타임에 지정한 쿼리 정의 정보 - 필드 정보, 조건 정보, 정렬 정보 - 를 이용하여 DataSet 객체에서 데이터를 가져 오게 됩니다.

4.13.1. XML 스키마 지정

[데이터 소스]-[XML 스키마 지정] 메뉴 항목을 선택하여 XML 스키마 지정 대화상자를 열기 합니다. 로컬 경로의 스키마 파일을 지정 한 후 확인 버튼을 클릭 합니다. XML 스키마 지정을 하면 쿼리 정의 메뉴 항목이 활성화 됩니다.



XPlatform DataSet 스키마 파일 구조는 다음과 같습니다. 최상위 요소는 **root** 이고, **Dataset** 요소는 여러 개 일 수 있습니다. 주 쿼리 또는 서브 쿼리 정의시 **DataSet** 요소의 **id** 속성을 테이블 이름으로 지정 하게 됩니다.

```

<?xml version="1.0" encoding="utf-8"?>
<root>
  <Dataset id="Dataset00">
    <ColumnInfo>
      <ConstColumn id="ConstColumn0" type="STRING" size="30" value=""/>
      <Column id="a1" type="STRING" size="256"/>
      <Column id="a2" type="STRING" size="256"/>
      <Column id="a3" type="STRING" size="256"/>
    </ColumnInfo>
    <Rows>
      <Row>
        <Col id="a1">사람 </Col>
        <Col id="a2">나무 </Col>
        <Col id="a3">111 </Col>
      </Row>
      <Row>
        <Col id="a1">해 </Col>
        <Col id="a2">달 </Col>
        <Col id="a3">222 </Col>
      </Row>
      <Row>
        <Col id="a1">나무 </Col>
        <Col id="a2">바람 </Col>
        <Col id="a3">333 </Col>
      </Row>
      <Row>
        <Col id="a1">aaa </Col>
        <Col id="a2">bbb </Col>
        <Col id="a3">444 </Col>
      </Row>
    </Rows>
  </Dataset>
</root>

```

4.13.2. 쿼리 정의

[데이터 소스]-[쿼리 정의] 메뉴항목을 선택 하여 쿼리 정의 대화상자를 열기 합니다. 주 쿼리 및 각 서브 쿼리에 대한 테이블 이름, 필드 정의, 조건정의, 정렬정의 정보를 쿼리 정의 대화상자에서 지정 합니다. 쿼리 정의 이후에는 데이터 셋 연결 대화상자를 이용 하여 텍스트 상자 또는 표의 셀에 데이터를 지정 할 수 있습니다.

쿼리 정의

쿼리 이름 : 주쿼리

필드 정의 :

테이블 이름 : Dataset00

선택 필드 리스트

필드이름	필드형	필드길이
<input checked="" type="checkbox"/> Semester	VARCHAR	256
<input checked="" type="checkbox"/> Etc	VARCHAR	256
<input checked="" type="checkbox"/> Subject	VARCHAR	256
<input checked="" type="checkbox"/> Point	VARCHAR	256
<input checked="" type="checkbox"/> Grade	VARCHAR	256

조건 정의 :

Semester == '97 /1 Semester'

정렬 정의 :

확인 취소

쿼리 이름 : 쿼리 정의 할 쿼리 이름을 선택 합니다. 서브 페이지 사용하는 표 문서인 경우 서브 쿼리 이름이 표시 됩니다.

필드 정의 : 테이블 이름 리스트에서 사용 할 테이블을 선택 하면 선택한 테이블의 필드 리스트가 선택 필드 리스트에 표시 됩니다. 선택 필드 리스트 에서 사용할 필드 이름은 체크 표시 하고, 사용 안 할 필드 이름은 체크 표시를 해제 합니다. 필드 이름, 필드형, 필드 길이 정보는 수정 할 수 없습니다.

조건 정의 : 데이터 **Filtering** 조건을 입력 합니다. 조건 정의시 파라미터 변수명, 주 쿼리 변수명 사용 가능 합니다.

예) Semester == '97 /1 Semester'
Semester == ':(myvar)'

정렬 정의 : 데이터 **Sorting** 조건을 입력 합니다. 정렬 정의시 파라미터 변수명 사용 가능 합니다. 오름차순 정렬인 경우 **ASC**, 내림차순 정렬인 경우 **DESC** 키워드를 이용 합니다.

예) Semester DESC

4.13.3. 미리 보기기

XML 스키마 지정을 하면 미리 보기시 필요한 보고서 실행 파라미터를 자동 생성 합니다. 파라미터 값 입력 대화상자를 열기 하여 생성 된 파라미터를 확인 할 수 있습니다.

/rcontype [FILE] /rfn [<데이터파일이름>]

[홈]-[미리 보기 or 웹 미리 보기] 메뉴항목을 선택 하여 보고서 실행을 테스트 합니다. 지정 한 XML 스키마 파일에서 미리 보기에 필요한 데이터를 자동 추출 하여 파일로 저장 한 후 미리 보기를 실행 합니다. 단, 쿼리 정의시 입력한 조건 정의, 정렬 정의 정보는 미리 보기에 적용 되지 않습니다.

4.13.4. XPlatform Runtime 과 데이터 연동 (보고서 실행 타임)

XPLATFORM_DS 접속방식 보고서 실행은 XPlatform Runtime 이 RDWrapper 컴포넌트를 이용하여 넘겨 준 DataSet 객체를 이용 합니다. XPlatform Runtime 과 데이터 연동은 ActiveX Viewer 에서 제공 되는 기능 입니다.

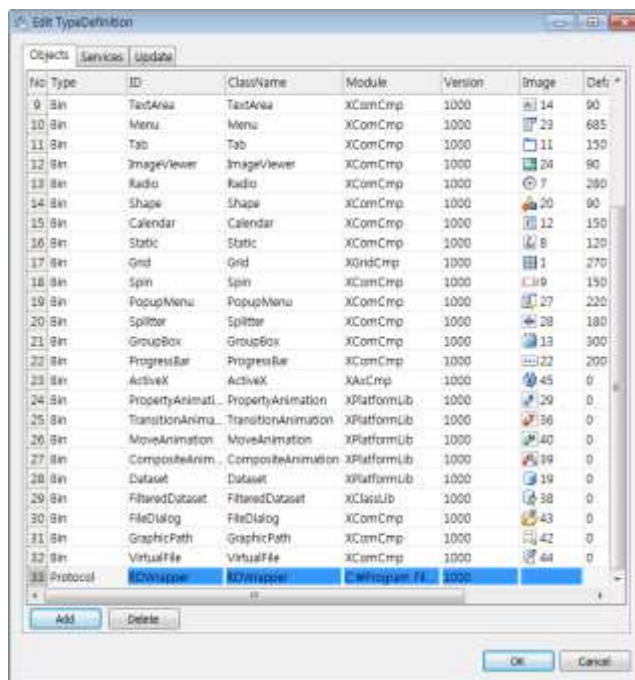
■ RDWrapper 개요

XPlatform UX Studio 에 등록 가능 한 컴포넌트 입니다. RDWrapper 는 CxReport ActiveX Viewer 를 포함 하고 있으며 6.0, 6.0u, 6.0j 버전을 옵션으로 제공 합니다. XPlatform Runtime 이 사용하는 DataSet 을 ActiveX Viewer 로 전달 하려면 RDWrapper 의 setDSArray() 메소드를 호출 합니다.

RDWrapper 는 XPlatform Runtime DataSet 을 전달 받아 ADO RecordSet 인터페이스 포인터 배열로 변환하여 ActiveX Viewer 의 SetAdoRsData() 메소드를 호출 합니다.

■ XPlatform UX Studio 에서 RDWrapper 컴포넌트를 등록 하고 사용 하는 방법

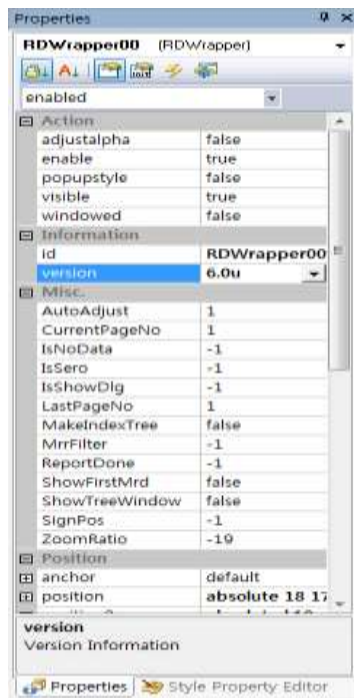
① RDWrapper 컴포넌트 등록 (Add Object)



XPlatform UX Studio, Project Explorer 창에서 TypeDefinition 을 더블 클릭하여 RDWrapper 컴포넌트를 등록 합니다.

② RDWrapper 객체 생성 및 Version 속성 값 지정

XPlatform UX Studio, 디자인 화면에 RDWrapper 객체를 그리기 한 후, RDWrapper 객체 속성 창에서 Version 속성 값을 지정 합니다.



③ 보고서 실행 Script 작성

XPlatform Runtime 이 RDWrapper 컴포넌트를 이용 하여 DataSet 객체를 ActiveX Viewer 로 전달 하기 위해, 다음과 같은 Script 를 작성 합니다.

```
function Button00_onclick(obj:Button, e:ClickEventInfo)
{
    var datasetArray = Array();
    datasetArray[0] = Dataset00;
    datasetArray[1] = Dataset01;
    datasetArray[2] = Dataset02;

    RDWrapper00.ApplyLicense ("http://192.168.1.39:8282/RDServer/rdagent.jsp");
    RDWrapper00.AutoAdjust = 2;
    RDWrapper00.setDSArray(datasetArray, "", ""); // DataSet Array 전달
    RDWrapper00.FileOpen("http://m2soft.co.kr/report/mrd/sample_ds.mrd", "/rp [보고서이름]");
}
}
```

④ XPlatform Runtime 과 데이터 연동 결과 화면

다음은 XPlatform Runtime 이 RDWrapper 컴포넌트를 이용 하여 DataSet 객체를 ActiveX

Viewer 로 전달 하여 보고서 실행 한 결과 화면 샘플 입니다.

Name Jim Martin **National ID** _____
Department Electrical and Computer Engineering **major** Computer Science **Second major** _____ **Student ID** 970501

97 /1 Semester			97 /2 Semester			98 /1 Semester			98 /2 Semester			
Subject	Point	Grade	Subject	Point	Grade	Subject	Point	Grade	Subject	Point	Grade	
1 Cybernetics	2	C	1 Cybernetics	2	C	1 Cybernetics	2	C	4 Computability Application/20 (Acquisition)/20	3	B	
1 Programming Projects In C++	3	D+	1 Programming Projects In C++	3	D+	1 Programming Projects In C++	3	D+	Average: /2.33			
2 Data Structures	3	B	2 Data Structures	3	B	2 Data Structures	3	B				
2 Theory of Computation and Formal Languages	3	C	2 Theory of Computation and Formal Languages	3	C	2 Theory of Computation and Formal Languages	3	C				
2 Admissions Requirements	2	B	2 Admissions Requirements	2	B	2 Admissions Requirements	2	B				
3 Error-Correcting Code	3	B+	3 Error-Correcting Code	3	B+	3 Error-Correcting Code	3	B+				
3 Artificial Intelligence Programming	3	D+	3 Artificial Intelligence Programming	3	D+	3 Artificial Intelligence Programming	3	D+				
Application/20 (Acquisition)/20 Average: /2.33			Application/20 (Acquisition)/20 Average: /2.33			Application/20 (Acquisition)/20 Average: /2.33			Application/20 (Acquisition)/20 Average: /2.33			
97 /2 Semester			98 /1 Semester			98 /2 Semester			Teaching Profession			
1 Engineering economy	3	B+	1 Engineering economy	3	B+	1 Engineering economy	3	B+	0.0	Thesis		0.0
2 Analysis of Algorithms	3	A	2 Analysis of Algorithms	3	A	2 Analysis of Algorithms	3	A	Mandatory Liberal Arts			
2 Engineering economic analysis	3	C	2 Engineering economic analysis	3	C	2 Engineering economic analysis	3	C	9.0	Basic		8.0
2 Introduction to Theory of Computation	3	B	2 Introduction to Theory of Computation	3	B	2 Introduction to Theory of Computation	3	B	Optional Liberal Arts			
2 Interactive Multimedia	3	B+	2 Interactive Multimedia	3	B+	2 Interactive Multimedia	3	B+	28.0	Optional Study		0.0
3 Fundamentals of Software Engineering	3	D	3 Fundamentals of Software Engineering	3	D	3 Fundamentals of Software Engineering	3	D	Mandatory Major Study			
4 Computability Application/20 (Acquisition)/20	3	B	4 Computability Application/20 (Acquisition)/20	3	B	4 Computability Application/20 (Acquisition)/20	3	B	8.0	Application		41
									Optional Major Study			
									0.0	Acquisition		41
									Mandatory Secondary Major Study			
									0.0	Examination Marks		108.6
									Optional Secondary Major Study			
									Average			2.55

24-Jul-2012
XXXXX University

페이지 1/1