

IBM® DB2® Life Sciences Data Connect



# 계획, 설치 및 구성 안내서

버전 7



IBM® DB2® Life Sciences Data Connect



# 계획, 설치 및 구성 안내서

버전 7

**주!**

이 책의 정보와 지원하는 제품을 사용하기 전에 반드시 75 페이지의 『주의사항』을 읽으십시오.

이 문서에는 IBM의 특허 정보가 들어 있습니다. 이 정보는 사용권 계약하에서 제공되며 저작권법의 보호를 받습니다. 이 책에 있는 정보는 어떠한 제품도 보증하지 않으며, 이 책에 제공된 어떤 내용도 이와 같이 해석되어서는 안됩니다.

책에 대한 주문은 한국 IBM 담당자 또는 해당 지역의 IBM 지방 사무소로 문의하십시오.

IBM은 귀하에게 책임이 발생하지 않는다고 판단되는 적절한 방식으로 귀하가 제공한 정보를 사용하거나 배포할 수 있습니다.

© Copyright International Business Machines Corporation 2001. All rights reserved.

# 목차

이 책에 대하여 . . . . .	v	파일 제한사항 및 고려사항 . . . . .	17
이 책의 사용자 . . . . .	v	파일 액세스 제어 모델 . . . . .	18
온라인 정보 . . . . .	v	최적화 추가 정보 및 고려사항 . . . . .	18
규약 . . . . .	vi	메시지 . . . . .	19
구문 도표를 읽는 방법 . . . . .	vi		
의견을 보내는 방법 . . . . .	ix		
<b>제1장 DB2 Life Sciences Data Connect의</b>		<b>제4장 데이터 소스로서 Documentum 사용</b>	23
정의 . . . . .	1	Documentum의 개념 . . . . .	23
DB2 Life Sciences Data Connect . . . . .	1	Documentum을 연함 시스템에 추가 . . . . .	25
IBM Life Sciences DiscoveryLink . . . . .	3	1단계: 환경 변수 설정 . . . . .	26
생명 과학 데이터 조회 . . . . .	4	2단계: Documentum 클라이언트 라이브러리에 대한 링크 . . . . .	26
		3단계: DB2 인스턴스 재순환 . . . . .	27
<b>제2장 DB2 Life Sciences Data Connect 설치.</b>	5	4단계: 랩퍼 등록 . . . . .	27
설치하기 전에 . . . . .	5	5단계: 선택적: DB2_DJ_COMM 환경 변수 설정 . . . . .	27
AIX, HP-UX 서버, Linux 및 Solaris 운영 환경에서 DB2 Life Sciences Data Connect 설치 . . . . .	6	6단계: 서버 등록 . . . . .	27
Windows NT 및 Windows 2000 서버에서 DB2 Life Sciences Data Connect 설치 . . . . .	7	7단계: 사용자 맵핑 . . . . .	30
설치 이후 . . . . .	8	8단계: 별칭 등록 . . . . .	30
		9단계: 사용자 정의 함수 등록 . . . . .	35
<b>제3장 데이터 소스로서 테이블 구조 파일 사용</b>	9	조회 실행 . . . . .	45
테이블 구조 파일의 정의 . . . . .	9	CreateNicknameFile 유틸리티 . . . . .	46
테이블 구조 파일의 유형 . . . . .	9	CreateNicknameFile 유틸리티 설치 . . . . .	47
테이블 구조 파일에서의 DB2 Life Sciences Data Connect 작업 방식 . . . . .	10	CreateNicknameFile 유틸리티 구성 . . . . .	47
테이블 구조 파일을 연함 시스템에 추가 . . . . .	11	Documentum 등록 테이블에서 DM_ID 오브젝트 유형 맵핑 . . . . .	48
1단계: 랩퍼 등록 . . . . .	11	반복 속성 정의 이중화 . . . . .	48
2단계: 선택적: DB2_DJ_COMM 환경 변수 설정 . . . . .	12	제한 및 고려사항 . . . . .	49
3단계: 서버 등록 . . . . .	12	액세스 제어 . . . . .	51
4단계: 별칭 등록 . . . . .	13	메시지 . . . . .	51
랩퍼 제한사항 및 고려사항 . . . . .	17		
		<b>제5장 데이터 소스로서 Excel 사용</b>	57
		Excel의 개념 . . . . .	57
		전제조건 . . . . .	59
		연함 시스템에 Excel 추가 . . . . .	59

1단계: 랩퍼 등록 . . . . .	59	컬럼 이름 변경 . . . . .	73
2단계: 서버 등록 . . . . .	60	데이터 유형 변경 . . . . .	73
3단계: 별칭 등록 . . . . .	61	파일 경로 변경 . . . . .	74
조회 실행 . . . . .	62	주의사항 . . . . .	75
샘플 시나리오 . . . . .	63	상표 . . . . .	78
제한 및 고려사항 . . . . .	65	관련 서적 . . . . .	81
랩퍼 고려사항 . . . . .	65	색인 . . . . .	83
랩퍼 제한사항 . . . . .	65	<b>IBM에 문의</b> . . . . .	85
Excel 파일 제한사항 . . . . .	65	제품 정보 . . . . .	85
파일 액세스 제어 모델 . . . . .	66		
메시지 . . . . .	66		
제6장 별칭 대체 . . . . .	73		

---

## 이 책에 대하여

이 책에는 다음과 같은 내용이 들어 있습니다.

- DB2 Life Sciences Data Connect에 대한 소개, IBM Life Sciences DiscoveryLink 오픈링에 적용되는 방식, 생명 과학에 맞게 조정된 종합적인 소프트웨어 및 서비스 세트.
- DB2 Life Sciences Data Connect의 설치 명령어.
- 랩퍼를 등록하여 연합 시스템에 데이터 소스를 추가하는 명령어. SQL을 사용하여 사용자나 응용프로그램이 데이터 소스와 통신할 수 있게 하는 모듈인 랩퍼.

텍스트에 대한 기술적 변경사항은 변경사항의 왼쪽에 있는 수직선에 의해 표시됩니다.

---

## 이 책의 사용자

이 책은 관리자가 생명 과학 연구 및 개발 데이터에 대한 연합 데이터베이스 환경을 설정하고, 응용프로그램 프로그래머가 이러한 환경에 대한 응용프로그램을 개발할 수 있도록 하기 위해 작성되었습니다.

---

## 온라인 정보

이 절에서는 이 제품과 관련된 웹 주소 및 전자우편 주소를 제공합니다.

**<http://www.ibm.com/software/data/db2/lifesciencesdataconnect/>**

DB2 Life Sciences Data Connect 제품 웹 사이트

**<http://www.ibm.com/solutions/lifesciences/discoverylink.html>**

DiscoveryLink 웹 사이트

**<http://www.ibm.com/solutions/lifesciences/>**

IBM Life Sciences 웹 사이트

---

## 규약

이 책에서는 다음과 같은 강조표시 규칙을 사용합니다.

### 굵은체

명령 및 그래픽 사용자 인터페이스(GUI) 제어(예: 필드 이름, 폴더 이름, 메뉴 선택사항)를 나타냅니다.

### 모노스페이스

입력한 텍스트나 코딩의 예를 나타냅니다.

### 이탤릭체

값으로 대체해야 하는 변수를 나타냅니다. 이탤릭체는 책 제목을 나타내거나 단어를 강조하기도 합니다.

대문자 오브젝트의 SQL 키워드 및 이름(예: 테이블, 뷰 및 서버)을 나타냅니다.

---

## 구문 도표를 읽는 방법

이 책에서는 다음과 같이 정의된 구조를 사용하여 구문을 설명합니다.

왼쪽에서 오른쪽으로, 위에서 아래로 행을 따라 구문 도표를 읽으십시오.

▶▶ 기호는 명령문의 시작을 나타냅니다.

▶▶ 기호는 명령문 구문이 다음 행에서 계속됨을 나타냅니다.

▶▶ 기호는 명령문이 이전 행에서 계속됨을 나타냅니다.

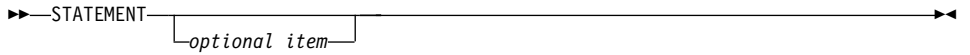
▶▶ 기호는 명령문의 끝을 나타냅니다.

필수 항목이 수평선(주 경로)에 표시됩니다.

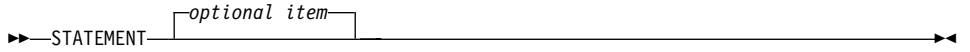
▶▶ STATEMENT—required item ▶▶

선택적 항목은 주 경로 아래에 표시됩니다.



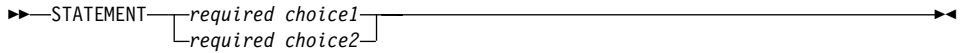


선택적 항목이 주 경로 위에 표시되는 경우 해당 항목은 명령문의 실행에는 영향을 미치지 않으며 판독에만 사용됩니다.



둘 이상의 항목을 선택할 수 있는 경우 스택으로 표시됩니다.

항목 중 하나를 반드시 선택해야 하는 경우 스택 중 한 항목이 주 경로에 표시됩니다.



항목 중 어느 것도 선택하지 않아도 되는 경우 주 경로 아래에 스택 전체가 표시됩니다.



항목 중 하나가 기본값이면, 하나는 주 경로 위에 표시되고 나머지 선택사항은 아래에 표시됩니다.



주 행 위에 있는 왼쪽으로 돌아가는 화살표는 반복될 수 있는 항목을 표시합니다. 이 경우 반복되는 항목은 하나 이상의 공백으로 구분되어야 합니다.



반복 화살표에 쉼표가 들어 있는 경우 반복되는 항목은 쉼표로 구분해야 합니다.



스택 위의 반복 화살표는 스택 항목에서 하나 이상의 선택사항을 작성하거나 하나의 선택사항을 반복할 수 있음을 표시합니다.

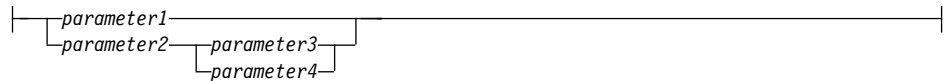
키워드는 대문자로 표시(예: FROM)되며 표시된 것과 똑같이 입력해야 합니다. 변수는 소문자로 표시됩니다(예: column-name). 이는 구문에서 사용자 제공 이름이나 값을 나타냅니다.

구두점, 괄호, 산술 연산자 또는 다른 기호들이 표시된 경우 이를 구문의 일부로서 입력해야 합니다.

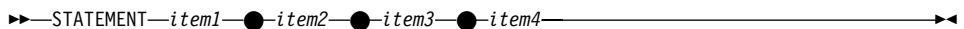
때때로 하나의 변수가 여러 개의 매개변수 세트를 표시하기도 합니다. 예를 들어, 다음 도표에서 변수 parameter-block은 **parameter-block**이라는 표제를 갖는 도표의 해석으로 대체될 수 있습니다.



**parameter-block:**



『큰 점』(●)은 임의의 순서로 지정될 수 있습니다.



위의 도표는 item2 및 item3이 어떤 순서로든 지정될 수 있음을 표시합니다. 다음 둘다 유효합니다.

```
STATEMENT item1 item2 item3 item4
STATEMENT item1 item3 item2 item4
```

---

## 의견을 보내는 방법

고객의 피드백은 IBM이 보다 좋은 품질의 정보를 제공하는 데 도움이 됩니다. 이 책이나 다른 DB2 문서에 대한 의견을 보내 주십시오. 다음 방법 중 하나를 사용하여 의견을 보내실 수 있습니다.

- 웹을 통해 의견을 보내 주십시오. <http://www.ibm.com/software/data/rcf>에서 IBM Data Management 온라인 고객 의견서 양식에 액세스할 수 있습니다.
- 전자우편으로 의견을 보내실 때는 [comments@vnet.ibm.com](mailto:comments@vnet.ibm.com)으로 보내 주십시오. 제품의 이름, 제품의 버전 번호, 책의 이름 및 부품 번호(적용 가능한 경우)도 함께 보내 주십시오. 특정 텍스트에 대한 의견을 보내는 경우 텍스트의 위치(예: 장 및 절 제목, 테이블 번호, 페이지 번호 또는 도움말 항목 제목)도 함께 보내 주십시오.



---

## 제1장 DB2 Life Sciences Data Connect의 정의

이 장에서는 DB2 Life Sciences Data Connect 제품, IBM Life Sciences DiscoveryLink 오픈링 및 생명 과학 데이터를 조회하기 위한 시스템 설정에 관련된 일반 단계를 소개합니다.

---

### DB2 Life Sciences Data Connect

IBM DB2 Life Sciences Data Connect는 DB2 연합 시스템이 분산된 소스에 있는 유전적, 화학적, 생물학적 및 기타 연구 데이터를 통합할 수 있게 합니다. DB2 연합 시스템은 DB2 Universal Database 서버와 DB2 Universal Database 서버가 데이터를 검색하는 여러 데이터 소스로 구성된 분산 컴퓨팅 시스템입니다.

연합 시스템에서 사용자나 응용프로그램은 SQL문을 사용하여 IBM, Oracle, Sybase 및 Microsoft의 관계형 데이터베이스 그리고 테이블 구조 파일과 같은 비관계형 데이터 소스와 같은 여러 이기종 데이터 소스에 위치한 데이터를 조회, 검색 및 조인할 수 있습니다. 2 페이지의 그림1에서는 연구 데이터의 여러 소스를 액세스하기 위해 DB2 Life Sciences Data Connect를 사용하는 연합 시스템에 대해 설명합니다.

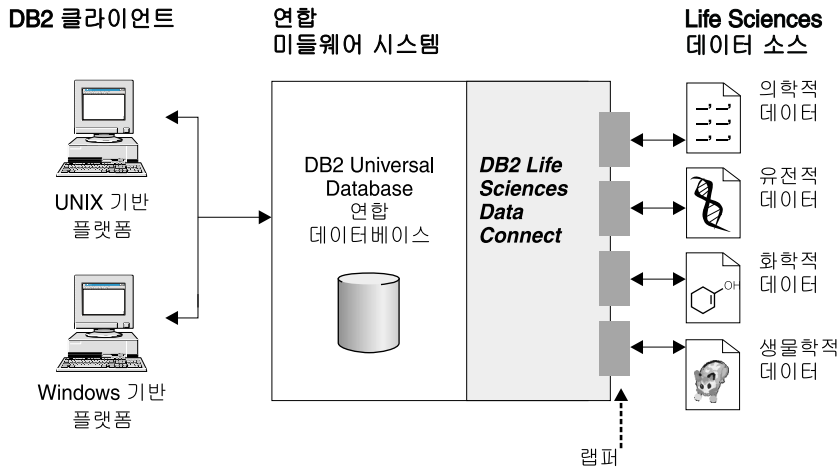


그림 1. DB2 Life Sciences Data Connect를 사용하여 생명 과학 데이터 액세스

DB2 연합 시스템은 클라이언트, 클라이언트가 조회를 제출하는 데이터베이스(연합 데이터베이스라고 함), 연합 데이터베이스가 데이터 소스와 통신하는 인터페이스 및 데이터 소스 자체를 포함합니다.

연합 서버가 데이터 소스와 통신하는 메커니즘을 랩퍼라고 합니다. 랩퍼를 구현하기 위해 서버는 랩퍼 모듈이라고 하는 라이브러리에 저장된 루틴을 사용합니다. 이 루틴은 서버가 데이터 소스에 연결하고 반복적으로 데이터 소스에서 데이터를 검색하는 조작을 수행하게 합니다.

연합 시스템이 설정되면 데이터 소스에 있는 정보는 하나의 큰 데이터베이스에 있는 것처럼 액세스될 수 있습니다. 사용자 및 응용프로그램은 여러 데이터 소스에서 데이터를 검색하는 조회를 하나의 연합 데이터베이스로 보냅니다. 응용프로그램은 연합 데이터베이스에 대해서도 다른 DB2 데이터베이스에서와 마찬가지로 작업합니다.

연합 시스템에 대한 자세한 정보는 *DB2 SQL 참조서*를 참조하십시오.

# IBM Life Sciences DiscoveryLink

DiscoveryLink 오픈링은 여러 이기종 데이터 소스의 데이터를 통합하기 위한 생명 과학 연구 및 개발 요구사항에 맞게 조정된 미들웨어 소프트웨어 및 서비스 세트입니다.

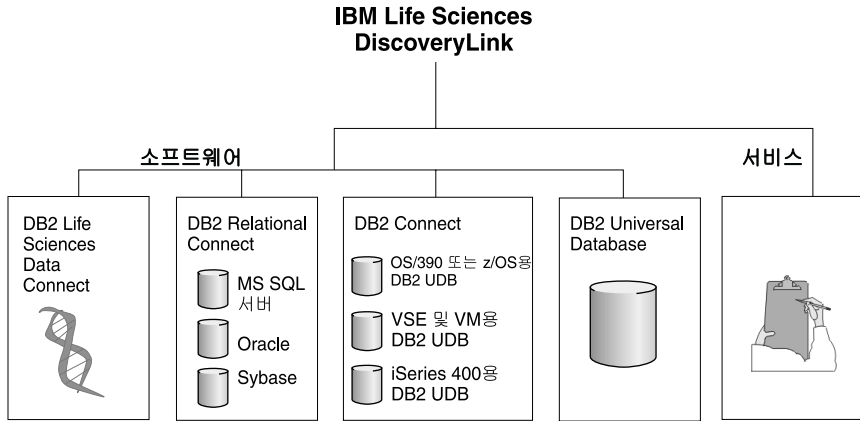


그림 2. IBM Life Sciences DiscoveryLink

예를 들어, DiscoveryLink에서 단일 SQL문을 사용하여 스위스에 있는 Oracle 데이터베이스의 단백질 순서 데이터, 일본에 있는 Sybase 데이터베이스의 화학적 구조 데이터 및 사용자 LAN에 있는 테이블 구조 플랫폼 파일에 저장된 분광기 데이터를 통합할 수 있습니다. 데이터는 마치 하나의 가상 데이터베이스에 있는 것처럼 표시됩니다.

소프트웨어 구성요소는 다음을 포함합니다.

## DB2 Life Sciences Data Connect

생명 과학 데이터에 액세스합니다.

## DB2 Relational Connect

Oracle, Sybase 및 Microsoft 관계형 데이터베이스에 액세스합니다. DB2 Relational Connect에 대한 자세한 내용은 *IBM DB2 Universal Database 릴리스 정보 버전 7.2/7.1 FixPak 4*를 참조하십시오.

## **DB2 Connect**

호스트 시스템에서 DB2 데이터베이스 서버에 액세스합니다. DB2 Connect에 대한 자세한 정보는 *DB2 Connect 사용자 안내서*를 참조하십시오.

## **DB2 Universal Database**

여러 이기종 데이터 소스에서 조회를 최적화하고 결과를 통합합니다. DB2 Universal Database에 대한 자세한 정보는 *DB2 관리 안내서*를 참조하십시오.

DiscoveryLink 소프트웨어 및 서비스에 대한 자세한 정보는 v 페이지의 『온라인 정보』를 참조하십시오.

---

## **생명 과학 데이터 조회**

생명 과학 데이터 소스에 들어 있는 데이터를 조회하고 검색하려면 우선 DB2 Life Sciences Data Connect를 설치해야 합니다.

DB2 Life Sciences Data Connect를 설치한 후 랩퍼를 데이터 소스에 구성하십시오. 이러한 프로세스를 랩퍼 등록이라고 합니다.



## 제2장 DB2 Life Sciences Data Connect 설치

이 장에서는 Windows NT, Windows 2000, AIX, HP-UX, Linux 및 Solaris 운영 환경에서 생명 과학 데이터를 조회하고 검색하기 위해 DB2 Life Sciences Data Connect를 설치하는 방법에 대해 설명합니다. 표1에서는 각 플랫폼에서 DB2 Life Sciences Data Connect 랩퍼를 표시합니다.

표 1. 플랫폼별 DB2 Life Sciences Data Connect 랩퍼

랩퍼	Windows NT / Windows 2000	AIX	HP-UX	Linux	Solaris 운영 환경
테이블 구조 파일	X	X	X	X	X
Documentum		X			
Excel	X				

DB2 Life Sciences Data Connect의 설치 이후에, 다양한 데이터 소스에 대한 랩퍼를 등록하여 연합 시스템에 추가해야 합니다. 각 생명 과학 랩퍼에 대한 명령어가 표2에 나열된 장에서 제공됩니다.

표 2. 각 생명 과학 랩퍼에 대한 정보를 찾는 장소

랩퍼	장
테이블 구조 파일	9 페이지의 『제3장 데이터 소스로서 테이블 구조 파일 사용』
Documentum	23 페이지의 『제4장 데이터 소스로서 Documentum 사용』
Excel	57 페이지의 『제5장 데이터 소스로서 Excel 사용』

### 설치하기 전에

연합 서버에 DB2 Life Sciences Data Connect를 설치하기 전에 다음을 수행하십시오.

- 연합 서버에 다음 제품 중 하나가 설치되어 있는지 확인하십시오.
  - DB2 Universal Database Enterprise Edition
  - DB2 Universal Database Enterprise - Extended Edition

- 데이터베이스에 연합 데이터베이스 시스템 지원이 설정되어 있는지 확인하십시오. 이 설정을 확인하려면 DB2 명령행에서 다음 명령을 수행하십시오.

GET DATABASE MANAGER CONFIGURATION

이 명령은 모든 데이터베이스 매개변수 및 현재 설정을 표시합니다. FEDERATED 매개변수가 예로 설정되었는지 확인하십시오.

FEDERATED 매개변수가 아니므로 설정된 경우 DB2 명령행 처리기에서 다음 명령을 수행하십시오.

UPDATE DATABASE MANAGER CONFIGURATION USING FEDERATED YES

## AIX, HP-UX 서버, Linux 및 Solaris 운영 환경에서 DB2 Life Sciences Data Connect 설치

AIX, HP-UX, Linux 및 Solaris 운영 환경에서 DB2 Life Sciences Data Connect를 설치하기 위해, db2setup 유틸리티를 사용하십시오.

주: db2setup 유틸리티 사용시 표시되는 화면은 연합 서버에 설치된 소프트웨어 제품에 의해 좌우됩니다. 이 단계에서는 DB2 Life Sciences Data Connect가 설치되어 있지 않다고 가정합니다.

1. 루트 권한을 가진 사용자로서 로그인하십시오.
2. DB2 Life Sciences Data Connect CD-ROM을 넣고 마운트하십시오. CD-ROM 마운트에 대한 정보는 *UNIX용 DB2 빠른 시작 매뉴얼*을 참조하십시오.
3. `cd /cdrom` 명령을 입력하여 CD-ROM이 마운트된 디렉토리로 변경하십시오. 여기서, *cdrom*은 사용자 제품 CD-ROM의 마운트 지점입니다.
4. 다음 명령을 입력하십시오.  

```
./db2setup
```

 DB2 설치 유틸리티 창이 열립니다.
5. 스페이스바를 눌러 생명 과학 데이터 소스에 대한 분산 액세스를 선택하십시오. 분산 액세스를 선택하면 옵션 옆에 별표가 표시됩니다.
6. 확인을 선택하고 Enter키를 누르십시오. DB2 서비스 창이 표시됩니다.

7. DB2 서비스 창에서 DB2 Life Sciences Data Connect에 대해 기존의 DB2 인스턴스를 설정하도록 선택할 수 있습니다. db2setup 설치 도움말에 설정하는 방법과 나머지 설치 단계가 나와 있습니다.

설치가 완료되면, DB2 Life Sciences Data Connect가 다른 DB2 제품과 함께 디렉토리에 설치됩니다.

- AIX 서버용 DB2에서 디렉토리는 /usr/lpp/db2\_07\_01입니다.
- Solaris 운영 환경 서버용 DB2에서 디렉토리는 /opt/IBMdb2/V7.1입니다.
- HP-UX 서버용 DB2에서 디렉토리는 /opt/IBMdb2/V7.1입니다.
- Linux 서버용 DB2에서 디렉토리는 /usr/IBMdb2/V7.1입니다.

---

## Windows NT 및 Windows 2000 서버에서 DB2 Life Sciences Data Connect 설치

1. DB2 Universal Database 설치를 수행하기 위해 작성한 사용자 계정으로 연합 서버에 로그인하십시오.
2. 실행 중인 모든 프로그램을 종료하여 설치 프로그램이 필요시 파일을 갱신할 수 있게 하십시오.
3. 설치 프로그램을 호출하십시오. 자동 또는 수동으로 설치 프로그램을 호출할 수 있습니다. 설치 프로그램이 자동으로 시작하는 데 실패하거나, 다른 언어로 설치를 수행하려는 경우, 수동으로 설치 프로그램을 호출하십시오.
  - 자동으로 설치 프로그램을 호출하려면, DB2 Life Sciences Data Connect CD를 드라이브에 넣으십시오. 자동 실행 기능이 자동으로 설치 프로그램을 시작합니다. 시스템 언어가 결정되며, 해당 언어의 설치 프로그램이 시작됩니다.
  - 수동으로 설치 프로그램을 호출하려면 다음을 수행하십시오.
    - a. 시작을 누른 다음 수행을 누르십시오.
    - b. 열기 필드에 다음 명령을 입력하십시오.

```
x:\setup /i language
```

여기서,

x:      CD-ROM 드라이브를 나타냅니다.

*language*

사용 언어를 나타냅니다(예를 들어, 영어의 경우 EN).

c. 확인을 누르십시오.

설치 런치패드를 여십시오.

4. 설치를 눌러 설치 프로세스를 시작하십시오.

5. 설치 프로그램에서 프롬프트에 따르십시오.

설치가 완료되면, DB2 Life Sciences Data Connect가 다른 DB2 제품과 함께 디렉토리에 설치됩니다. 기본 설치 디렉토리는 \sqllib입니다.

---

## 설치 이후

설치 이후에, 랩퍼 라이브러리 파일은 사용자 시스템에 위치합니다. 이 라이브러리는 랩퍼 등록 프로세스 중 사용됩니다. 지원되는 플랫폼별로 각 라이브러리에 대한 기본 파일 이름이 표3에 나열됩니다.

표3. 플랫폼별 기본 랩퍼 라이브러리 이름

랩퍼	Windows NT / Windows 2000	AIX	HP-UX	Linux	Solaris 운영 환경
테이블 구조 파일	liblsfile.dll	liblsfile.a	liblsfile.sl	liblsfile.so	liblsfile.so
Documentum		liblsdctm.a			
Excel97 및 Excel98	liblsexcel97. dll				
Excel2000	liblsexcel2k. dll				

---

## 제3장 데이터 소스로서 테이블 구조 파일 사용

이 장에서는 다음에 대해 설명합니다.

- 테이블 구조 파일
- 연합 시스템에 테이블 구조 파일 데이터 소스를 추가하는 방법
- 제한 및 고려사항
- 사용된 파일 액세스 제어 모델
- 최적화 추가 정보
- 테이블 구조 파일에서 작업시 발생할 수 있는 메시지

---

### 테이블 구조 파일의 정의

테이블 구조 파일은 일련의 레코드로 구성된 일반 구조이며 각 레코드에는 임의의 분리문자로 구분되는 동일한 수의 필드가 들어 있습니다. 널(NULL) 값은 양 옆에 2개의 분리문자로 표시됩니다.

다음 예는 DRUGDATA1.TXT 파일의 내용을 표시한 것입니다. 여기에는 세 개의 레코드가 들어 있으며 각 레코드는 세 개의 필드를 가지며 쉼표로 구분됩니다.

```
234,DrugnameA,Manufacturer1  
332,DrugnameB,Manufacturer2  
333,DrugnameC,Manufacturer2
```

첫 번째 필드는 약의 고유한 ID 번호입니다. 두 번째 필드는 약의 이름입니다. 세 번째 필드는 약을 생산하는 제조업체의 이름입니다.

### 테이블 구조 파일의 유형

테이블 구조 파일은 정렬되거나 정렬되지 않을 수 있습니다.

### 정렬된 파일

DRUGDATA1.TXT에는 정렬된 레코드가 들어 있습니다. 파일은 약의 고유한 ID 번호인 첫 번째 필드로 정렬됩니다. 이 필드는 각 약에 대해 고유하므로 기본 키입니다. 정렬된 파일은 오름차순으로 정렬되어야 합니다.

```
234,DrugnameA,Manufacturer1
332,DrugnameB,Manufacturer2
333,DrugnameC,Manufacturer2
```

### 정렬되지 않은 파일

DRUGDATA2.TXT에는 정렬되지 않은 레코드가 들어 있습니다. 파일에서 레코드가 나열되는 방식에는 순서가 없습니다.

```
332,DrugnameB,Manufacturer2
234,DrugnameA,Manufacturer1
333,DrugnameC,Manufacturer2
```

랩퍼는 정렬되지 않은 파일보다 정렬된 데이터 파일에서 훨씬 더 효과적으로 검색을 수행할 수 있습니다.

## 테이블 구조 파일에서의 DB2 Life Sciences Data Connect 작업 방식

랩퍼라는 모듈 사용을 사용하면 DB2 Life Sciences Data Connect는 테이블 구조 파일에 있는 데이터를 조회하는 SQL문이 일반 관계 테이블 또는 뷰에 포함되어 있는 것처럼 처리할 수 있습니다. 이는 테이블 구조 파일에 있는 데이터를 관계 데이터나 다른 테이블 구조 파일에 있는 데이터에 조인시킵니다.

예를 들어, 테이블 구조 파일 DRUGDATA1.TXT가 사용자 실험실에 있는 컴퓨터에 있다고 가정하십시오. 이 데이터를 조회하여 다른 데이터 소스에 있는 다른 테이블과 일치시키려는 시도는 지루할 수 있습니다.

DRUGDATA1.TXT를 DB2 Life Sciences Data Connect에 등록하면 파일은 관계 데이터 소스인 것처럼 작동합니다. 이제 다른 관계 및 비관계 데이터 소스와 함께 파일을 조회하고 함께 데이터를 분석할 수 있습니다.

예를 들어, 다음 조회를 수행할 수 있습니다.

```
SELECT * FROM DRUGDATA1 ORDER BY DCODE
```

이 조회는 다음과 같은 결과를 생성합니다.

코드	약	제조업체
234	DrugnameA	Manufacturer1
332	DrugnameB	Manufacturer2
333	DrugnameC	Manufacturer2

## 테이블 구조 파일을 연합 시스템에 추가

테이블 구조 파일의 데이터 소스를 연합 서버에 추가하려면 다음을 수행하십시오.

1. CREATE WRAPPER 명령을 사용하여 래퍼를 등록하십시오.
2. 선택적: 조회 성능을 개선하려면 DB2\_DJ\_COMM 환경 변수를 설정하십시오.
3. CREATE SERVER 명령을 사용하여 서버를 등록하십시오.
4. 모든 테이블 구조 파일에 대한 CREATE NICKNAME 명령을 사용하여 별칭을 등록하십시오.

이 단계는 이 절에서 자세히 설명됩니다. 명령은 DB2 명령행 처리기에서 수행될 수 있습니다.

### 1단계: 래퍼 등록

테이블 구조 파일에 액세스하는 데 사용할 래퍼를 지정하려면 CREATE WRAPPER문을 사용하십시오. 래퍼는 연합 서버가 통신을 하고 데이터 소스에서 데이터를 검색하는 데 사용하는 메커니즘입니다.

래퍼는 라이브러리 파일로 시스템에 설치됩니다. 8 페이지의 표3에서는 테이블 구조 파일 래퍼에 대한 플랫폼별 기본값 이름을 나열합니다. 이 라이브러리 파일은 래퍼가 사용되기 전에 CREATE WRAPPER문을 사용하여 등록되어야 합니다.

예를 들어, AIX에서 래퍼를 등록하려면, 다음 명령문을 실행하십시오.

```
CREATE WRAPPER laboratory_flat_files LIBRARY 'liblsfile.a'
```

이 예에서 laboratory\_flat\_files는 래퍼용으로 선택된 이름입니다. 이 이름은 등록 중인 래퍼의 데이터베이스 내에서 고유해야 합니다. AIX에서 테이블 구조 파일 래퍼에 대한 필수 라이브러리 이름은 liblsfile.a입니다.

라이브러리 이름은 기본값으로 `liblsfile.a`로 설치되지만 설치 중에 사용자 정의할 수 있습니다. 올바른 이름은 시스템 관리자에게 문의하십시오.

CREATE WRAPPER문에 대한 자세한 정보는 *DB2 SQL* 참조서를 참조하십시오.

## 2단계: 선택적: DB2\_DJ\_COMM 환경 변수 설정

테이블 구조 파일에 액세스할 때 성능을 개선하려면 `DB2_DJ_COMM` 환경 변수를 설정하십시오. 이 변수는 연합 서버가 초기화 중에 래퍼를 로드하는지 여부를 결정합니다. 11 페이지의 『1단계: 래퍼 등록』에 지정된 래퍼에 해당하는 래퍼 라이브러리가 포함되도록 `DB2_DJ_COMM` 환경 변수를 설정하십시오. 예를 들어, 다음과 같습니다.

```
export DB2_DJ_COMM='liblsfile.a'
```

등호(=)의 양쪽에 공백이 없게 하십시오.

`DB2_DJ_COMM` 환경 변수에 대한 자세한 정보는 *DB2 관리 안내서*를 참조하십시오.

## 3단계: 서버 등록

테이블 구조 파일에 액세스할 서버를 정의하려면 `CREATE SERVER`문을 사용하십시오. 정렬되거나 정렬되지 않은 테이블 구조 파일에 액세스하도록 서버를 구성할 수 있지만, 하나의 서버가 둘다에 사용될 수는 없습니다. 그러나 한 서버는 정렬된 테이블 구조 파일을 관리하고 다른 서버는 정렬되지 않은 테이블 구조 파일을 관리하도록 정의할 수 있습니다. 예를 들어, 다음과 같습니다.

```
CREATE SERVER biochem_lab TYPE SORTED VERSION 1.0 WRAPPER laboratory_flat_files  
OPTIONS (NODE 'biochem_node1')
```

이 예에서 `biochem_lab`은 테이블 구조 파일 서버에 지정된 이름입니다. 이름은 등록 중인 서버의 데이터베이스에 고유해야 합니다.

`TYPE` 키워드가 필요합니다. 이 키워드는 사용하는 검색 알고리즘을 결정하는 데 사용됩니다. 데이터 소스에 따라 `SORTED` 또는 `UNSORTED`를 지정하십시오. 정렬 및 정렬되지 않은 파일 유형에 대한 자세한 정보는 9 페이지의 『테이블 구조 파일의 유형』을 참조하십시오.



VERSION이 필요합니다. 이는 1.0으로 설정되어야 합니다.

이 예에서 랩퍼 이름은 `laboratory_flat_files`입니다. 이는 이전에 CREATE WRAPPER문에 지정된 이름입니다.

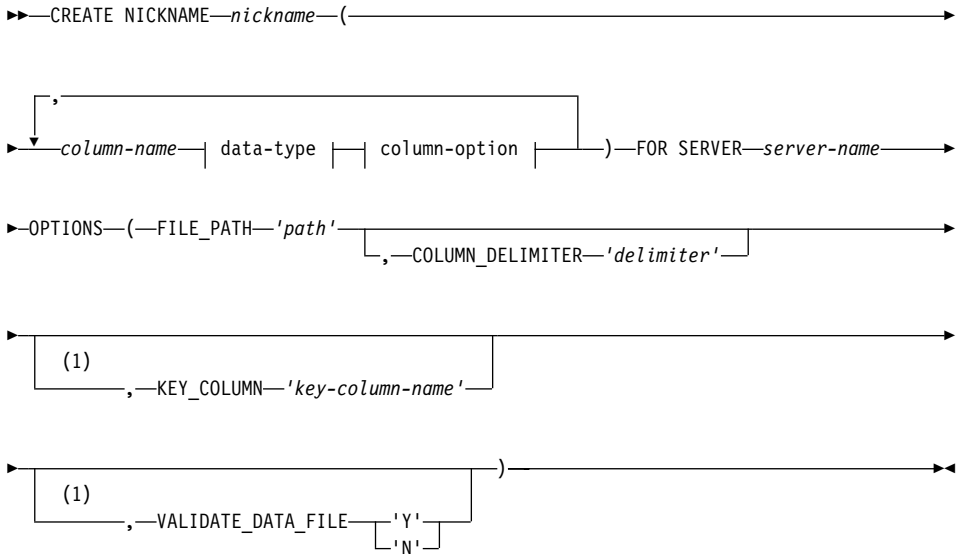
NODE 옵션이 필요합니다. 이는 지역 노드에 주어진 이름입니다. 원하는 텍스트 문자열을 포함할 수 있습니다.

CREATE SERVER문에 대한 자세한 정보는 *DB2 SQL 참조서*를 참조하십시오.

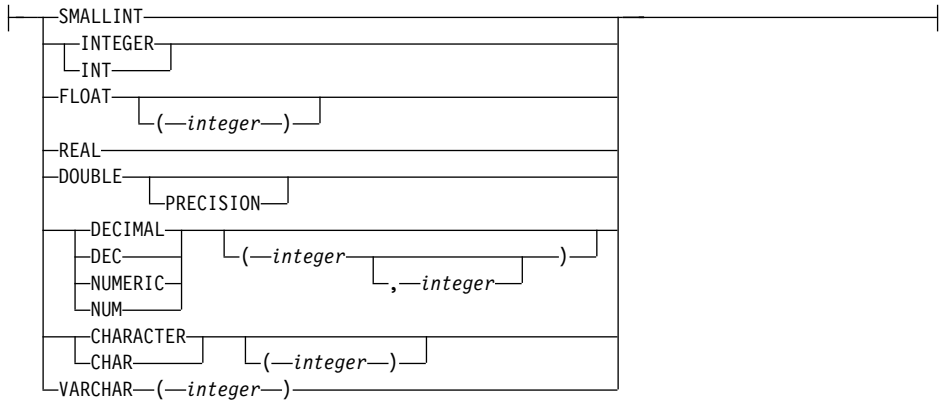
#### 4단계: 별칭 등록

12 페이지의 『3단계: 서버 등록』에 등록된 서버를 사용하여 액세스하려는 각 테이블 구조 파일의 별칭을 등록하려면 CREATE NICKNAME문을 사용하십시오. 별칭은 조회에서 테이블 구조 파일을 참조할 때 사용됩니다.

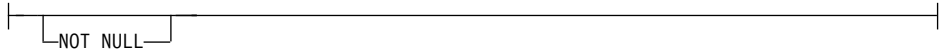
CREATE NICKNAME문의 구문은 다음과 같습니다.



**data-type:**



**column-option:**



**주:**

- 1 정렬된 파일 전용 선택사항.

*nickname*

액세스되는 테이블 구조 파일의 고유한 별칭. 등록 중인 스키마의 다른 모든 별칭, 테이블 및 뷰와 구별되어야 합니다.

*column-name*

테이블 구조 파일의 각 필드에 제공된 고유한 이름. 각 컬럼 이름 다음에는 데이터 유형이 옵니다. 유형이 CHAR, VARCHAR, SMALLINT, INTEGER, FLOAT, DOUBLE, REAL 및 DECIMAL인 컬럼만이 지원됩니다.

**SMALLINT**

작은 정수에 사용됩니다.

**INTEGER 또는 INT**

큰 정수에 사용됩니다.

**FLOAT(*integer*)**

*integer*의 값에 따라 단일 또는 배정밀도 부동 소수점 숫자에 사용됩니다.

*integer*의 값은 1 - 53에 있어야 합니다. 1 - 24 값은 단정밀도를 나타내며 25 - 53은 배정밀도를 나타냅니다.

### **REAL**

단정밀도 부동 소수점에 사용됩니다.

### **DOUBLE 또는 DOUBLE PRECISION**

배정밀도 부동 소수점에 사용됩니다.

### **FLOAT**

배정밀도 부동 소수점에 사용됩니다.

**DECIMAL**(*precision-integer, scale-integer*) 또는 **DEC**(*precision-integer, scale-integer*)

십진수에 사용됩니다.

첫 번째 정수는 숫자의 정밀도로 총 자리수를 나타냅니다. 이 값의 범위는 1 - 31입니다.

두 번째 정수는 숫자의 스케일입니다. 즉, 소수점의 오른쪽에 있는 자리수입니다. 이 값의 범위는 0부터 숫자의 정밀도까지입니다.

정밀도 및 스케일이 지정되지 않은 경우 기본값인 5, 0이 사용됩니다.

단어 **NUMERIC** 및 **NUM**은 **DECIMAL** 및 **DEC**와 동의어로서 사용될 수 있습니다.

**CHARACTER**(*integer*) 또는 **CHAR**(*integer*) 또는 **CHARACTER** 또는 **CHAR**

고정 길이를 가진 *integer*에 사용되며 범위는 1 - 254입니다. 길이 스펙이 누락되면 길이는 1로 간주됩니다.

**VARCHAR**(*integer*)

가변 길이를 가진 *integer*에 사용되며 범위는 1 - 32767입니다.

### **NOT NULL**

컬럼에 널(Null) 값이 들어 있지 않게 합니다.

### *server-name*

12 페이지의 『3단계: 서버 등록』에 등록된 서버를 식별합니다. 이 서버는

테이블 구조 파일을 액세스하는 데 사용됩니다. 파일이 정렬된 경우 지정된 서버의 유형은 SORTED이어야 하며, 그렇지 않으면 UNSORTED 유형을 지정해야 합니다.

*'path'* 작은따옴표로 묶여 액세스되는 테이블 구조 파일로의 완전한 경로. 데이터 파일은 파이프나 다른 비표준 파일 유형보다는 표준 파일이나 기호 링크이어야 합니다. 데이터 파일은 DB2 인스턴스 소유자가 읽을 수 있어야 합니다. 인스턴스 소유자에 대한 자세한 정보는 *DB2 관리 안내서*를 참조하십시오.

*'delimiter'*

테이블 구조 파일의 컬럼을 구분하기 위해 사용되는 분리문자로 작은따옴표로 묶입니다. 컬럼 분리문자가 정의되지 않으면 컬럼 분리문자의 기본값으로 쉼표가 사용됩니다. 컬럼 분리문자는 컬럼에 대해 유효한 데이터로서 존재할 수 없습니다. 예를 들어, 컬럼 중 하나에 쉼표가 삽입된 데이터가 들어 있는 경우 컬럼 분리문자로 쉼표를 사용할 수 없습니다.

*'key-column-name'*

정렬된 파일에서 키를 형성하는 파일에서의 컬럼의 이름으로, 작은따옴표로 묶입니다. 정렬된 파일에만 이 옵션을 사용하십시오. 대소문자가 구분되지 않습니다.

단일 컬럼 키만이 지원됩니다. 값은 CREATE NICKNAME문에 정의된 컬럼의 이름이어야 합니다. 컬럼은 오름차순으로 정렬되어야 합니다. 정렬된 서버에 대한 값이 지정되지 않은 경우 별칭을 갖는 파일의 첫 번째 컬럼이 기본값이 됩니다.

## **VALIDATE\_DATA\_FILE**

정렬된 파일에 사용됩니다. 이 옵션은 랩퍼가 키 컬럼이 오름차순으로 정렬되었는지를 검증하는지 여부를 지정합니다. 이 옵션에 유효한 값은 'Y' 또는 'N'으로, 작은따옴표로 묶입니다. 등록시에만 값이 점검됩니다. 이 옵션이 지정되지 않은 경우 유효성 검증은 발생하지 않습니다.

다음 예에서는 9 페이지의 『테이블 구조 파일의 정의』에서 설명하는 테이블 구조 파일 DRUGDATA1.TXT에 대한 CREATE NICKNAME문을 보여줍니다.

```
CREATE NICKNAME DRUGDATA1(Dcode Integer, Drug CHAR(20), Manufacturer CHAR(20))
FOR SERVER biochem lab OPTIONS(FILE_PATH '/usr/pat/DRUGDATA1.TXT',
COLUMN_DELIMITER ',', KEY_COLUMN 'Dcode', VALIDATE_DATA_FILE 'Y')
```

CREATE NICKNAME문에 대한 자세한 정보는 *DB2 SQL 참조서*를 참조하십시오. 별칭에 대한 자세한 정보는 *DB2 관리 안내서*를 참조하십시오.

---

## 래퍼 제한사항 및 고려사항

- Passthru 세션은 래퍼를 사용할 때 허용되지 않습니다.
- 여러 컬럼으로 구성된 키는 허용되지 않습니다.
- 파일은 오름차순으로만 정렬되어야 합니다. 내림차순에서 정렬은 지원되지 않습니다.
- 래퍼는 NOT NULL 제한조건을 강제로 적용하지 않지만 DB2는 강제로 적용합니다. 별칭을 작성하고 컬럼에 NOT NULL 제한조건에 첨부하여 해당 컬럼에 대해 널(NULL) 값을 갖는 행을 선택한 경우 DB2는 널(NULL) 값을 NOT NULL 컬럼에 지정할 수 없음을 나타내는 SQL0407N 오류를 발행합니다.  
이 규칙에 대한 예외는 정렬된 서버에 적용됩니다. 정렬된 서버에 정의된 별칭의 키 컬럼은 널(NULL)일 수 없습니다. 정렬된 서버를 사용하여 별칭에 대한 NULL 키 컬럼을 찾은 경우 키 컬럼이 누락되었음을 나타내는 SQL1822N 오류가 발행됩니다.
- DB2 Universal Database Enterprise-Extended Edition에서, 별칭이 작성된 테이블 구조 파일은 각 노드의 동일한 경로 이름으로 액세스할 수 있어야 합니다. 일반 경로를 갖는 모든 노드로부터 액세스할 수 있는 파일은 DB2 Universal Database 노드에 있을 필요가 없습니다.

---

## 파일 제한사항 및 고려사항

- 파일은 행 당 하나의 레코드로 제한됩니다.
- 각 레코드는 동일한 수의 구분된 컬럼을 가져야 합니다.
- 각 레코드는 래퍼가 설치된 플랫폼에 대한 표준 행 종료 문자로 종료되어야 합니다.
- 컬럼 분리문자는 파일에서 일관성이 있어야 합니다.

- 널(Null) 값은 양 옆에 있는 두 개의 분리문자로 표시되거나, 널(Null)필드가 행의 마지막 필드인 경우 행 종료자가 뒤에 오는 분리문자로 표시됩니다.
- 기수(radix) 문자는 LC\_NUMERIC 자국어 지원 범주의 RADIXCHAR 항목에 의해 결정됩니다.
- 정렬된 데이터 소스는 LC\_COLLATE 자국어 지원 범주에 있는 설정에 의해 정의된 현재 로케일의 대조 순서에 따라 오름차순으로 정렬되어야 합니다.
- 데이터베이스 코드 페이지는 파일의 문자 세트와 일치해야 하며, 그렇지 않으면 예상하지 못한 결과를 가져올 수 있습니다.
- 다중바이트 문자가 들어 있는 파일은 지원되지 않습니다.
- 컬럼 유형에 비해 숫자 이외의 필드가 너무 긴 경우 초과 데이터는 절단됩니다.
- 파일의 십진수 필드에 컬럼 유형의 스케일 매개변수에 의해 허용되는 것보다 더 많은 자리수가 기수(radix) 문자 다음에 오는 경우 초과 데이터는 절단됩니다.
- 최대 행 길이는 32768입니다.

---

## 파일 액세스 제어 모델

데이터베이스 관리 시스템은 DB2 인스턴스 소유자의 권한을 사용하여 테이블 구조 파일에 액세스합니다. 랩퍼는 이 사용자 ID(또는 그룹 ID)로 읽을 수 있는 파일에만 액세스할 수 있습니다. 응용프로그램의 권한 부여 ID(연합 데이터베이스에 대한 연결을 설정하는 ID)는 관련되어 있지 않습니다.

---

## 최적화 추가 정보 및 고려사항

- 시스템은 정렬되지 않은 파일보다 정렬된 데이터 파일에서 훨씬 더 효과적으로 검색을 수행할 수 있습니다.
- 정렬된 파일의 경우 키 컬럼에 대한 값이나 범위를 지정하여 성능을 개선시킬 수 있습니다.
- 테이블 구조 파일의 별칭에 대한 통계는 SYSTAT 뷰를 갱신하여 수동으로 갱신되어야 합니다. 수동으로 STSTAT 뷰를 갱신하는 방법에 대해서는 *DB2 관리 안내서*를 참조하십시오.

## 메시지

이 절에서는 테이블 구조 파일의 랩퍼로 작업하는 중에 발생할 수 있는 메시지 목록을 나열하고 설명합니다. 메시지에 대한 자세한 정보는 *DB2 메시지 참조서*를 참조하십시오.

표 4. 테이블 구조 파일의 랩퍼가 발행한 메시지

오류 코드	메시지	설명
SQL0405N	숫자 리터럴 "<literal>"은 범위를 벗어나므로 유효하지 않습니다.	데이터 파일에 있는 컬럼 또는 SQL문에 있는 술어 값에 해당 데이터 유형에 대해 가능한 범위를 벗어난 값이 들어 있습니다. 데이터 파일을 수정하거나 좀 더 적합한 유형으로 컬럼을 재정의하십시오.
SQL0408N	값이 관련 지정 목표의 데이터 유형과 호환 가능하지 않습니다. 목표 이름은 "<column_name>"입니다.	데이터 파일의 컬럼에 해당 데이터 유형에 유효하지 않은 문자가 들어 있습니다. 데이터 파일을 수정하거나 좀 더 적합한 유형으로 컬럼을 재정의하십시오.
SQL0901N	심각하지 않은 시스템 오류로 인해 SQL문이 실패했습니다. 후속 SQL문은 처리될 수 있습니다(이유: "데이터 소스 경로가 널(NULL)임").	IBM 고객만족센터에 문의하십시오.
SQL0901N	심각하지 않은 시스템 오류로 인해 SQL문이 실패했습니다. 후속 SQL문은 처리될 수 있습니다(이유: "키 컬럼 검색 실패").	IBM 고객만족센터에 문의하십시오.
SQL0901N	심각하지 않은 시스템 오류로 인해 SQL문이 실패했습니다. 후속 SQL문은 처리될 수 있습니다(이유: "STAT가 데이터 소스에서 실패했음. 오류 번호 = <error_number>").	IBM 고객만족센터에 문의하십시오.
SQL0901N	심각하지 않은 시스템 오류로 인해 SQL문이 실패했습니다. 후속 SQL문은 처리될 수 있습니다(이유: "컬럼 정보가 없음").	IBM 고객만족센터에 문의하십시오.

표 4. 테이블 구조 파일의 랩퍼가 발행한 메시지 (계속)

오류 코드	메시지	설명
SQL0901N	심각하지 않은 시스템 오류로 인해 SQL문이 실패했습니다. 후속 SQL문은 처리될 수 있습니다(이유: "서버 구문분석기 실패, RC = <parser_return_code>").	IBM 고객만족센터에 문의하십시오.
SQL0901N	심각하지 않은 시스템 오류로 인해 SQL문이 실패했습니다. 후속 SQL문은 처리될 수 있습니다(이유: "지원되지 않는 연산자").	IBM 고객만족센터에 문의하십시오.
SQL0901N	심각하지 않은 시스템 오류로 인해 SQL문이 실패했습니다. 후속 SQL문은 처리될 수 있습니다(이유: "바인드 변수를 식별할 수 없음").	IBM 고객만족센터에 문의하십시오.
SQL0901N	심각하지 않은 시스템 오류로 인해 SQL문이 실패했습니다. 후속 SQL문은 처리될 수 있습니다(이유: "조회 구성요소를 식별할 수 없음").	IBM 고객만족센터에 문의하십시오.
SQL0901N	심각하지 않은 시스템 오류로 인해 SQL문이 실패했습니다. 후속 명령문은 처리될 수 있습니다(이유: "값을 변환할 때 데이터에 액세스할 수 없음").	IBM 고객만족센터에 문의하십시오.
SQL1816N	랩퍼 "<wrapper_name>"은 연합 데이터베이스에 정의하려고 시도 중인 데이터 소스의 "유형"("<type>" "")에 액세스하는 데 사용될 수 없습니다.	서버 유형이 유효하지 않습니다. 허용되는 서버 유형은 SORTED 또는 UNSORTED뿐입니다. SQL 문을 변경하고 리터하십시오.
SQL1822N	데이터 소스 "<server_name>"에 서 예상치 못한 오류 코드 "ERRNO = <error_number>"를 받았습니. 연관된 텍스트 및 토큰은 "파일을 읽을 수 없습니다"입니다.	오류 번호의 값을 확인하십시오. DB2 인스턴스 소 사용자가 파일을 읽을 수 있는지 확인하십시오. 그런 다음 SQL문을 재실행하십시오.



표 4. 테이블 구조 파일의 랩퍼가 발행한 메시지 (계속)

오류 코드	메시지	설명
SQL1822N	데이터 소스 "<server_name>"에 서 예상치 못한 오류 코드 "데이터 오류"를 받았습니 다. 연관된 텍스트 및 토큰은 "데이터 소스가 비표준 파일입니다"입니다.	데이터 소스 파일은 디렉토리, 소켓 또는 FIFO입니다. 표준 파일만이 데이터 소스로서 액세스될 수 있습니다. 유효한 파일을 지정하도록 FILE_PATH 옵션을 변경한 후 SQL 명령을 재발행하십시오.
SQL1822N	데이터 소스 "<server_name>"에 서 예상치 못한 오류 코드 "ERRNO = <error_number>"를 받았습니 다. 연관된 텍스트 및 토큰은 "파일 열기 오류"입니다.	랩퍼가 파일을 열 수 없습니다. 오류가 발생한 이유를 판별하려면 오류 번호를 확인하십시오. 데이터 소스에서 문제점을 지정하고 SQL 명령을 재발행하십시오.
SQL1822N	데이터 소스 "<server_name>"에 서 예상치 못한 오류 코드 "데이터 오류"를 받았습니 다. 연관된 텍스트 및 토큰은 "키 컬럼 누락"입니다.	데이터 소스에서 검색된 레코드에 키 필드가 누락되었습니다. 키 컬럼은 널(NULL)일 수 없습니다. 데이터를 지정하거나 정렬되지 않은 서버에 파일을 등록하십시오.
SQL1822N	데이터 소스 "<server_name>"에 서 예상치 못한 오류 코드 "데이터 오류"를 받았습니 다. 연관된 텍스트 및 토큰은 "파일이 정렬되어 있지 않습니다"입니다.	파일이 키 컬럼으로 정렬되지 않았습니다. 다음 중 하나를 수행하십시오. 올바른 컬럼을 지정하도록 KEY_COLUMN 옵션을 변경하거나 데이터 파일을 재정렬하거나 정렬되지 않은 서버에 별칭을 등록하십시오.
SQL1822N	데이터 소스 "<server_name>"에 서 예상치 못한 오류 코드 "데이터 오류"를 받았습니 다. 연관된 텍스트 및 토큰은 "키가 정의 크기를 초과합니다"입니다.	데이터 소스에서 읽은 키 컬럼 필드가 DB2 컬럼 정의보다 크면 랩퍼 검색 루틴이 제대로 작동하지 못할 수 있습니다. 데이터를 지정하거나 별칭 정의를 지정하고 별칭을 다시 등록하십시오.
SQL1822N	데이터 소스 "<server_name>"에 서 예상치 못한 오류 코드 "데이터 오류"를 받았습니 다. 연관된 텍스트 및 토큰은 "데이터 파일의 행이 32K를 초과합니다"입니다.	데이터 파일의 행이 랩퍼가 허용하는 최대 행 길이를 초과했습니다. 행 길이는 32768보다 클 수 없습니다. 데이터 파일에서 행의 길이를 줄이십시오.
SQL1823N	서버 "<server_name>"의 데이터 유형 "<data_type>"에 대한 데이터 유형 맵핑이 존재하지 않습니다.	지원되지 않는 데이터 유형을 가진 별칭이 정의되었습니다. 지원되는 데이터 유형만을 사용하여 별칭을 재정의하십시오.
SQL1881N	"<option_name>"은 "<object_name>"에 대해 유효한 옵션이 아닙니다.	나열된 값은 나열된 오브젝트의 유효한 옵션이 아닙니다. 유효하지 않은 옵션을 제거하거나 변경한 다음 SQL문을 다시 제출하십시오.

표 4. 테이블 구조 파일의 랩퍼가 발행한 메시지 (계속)

오류 코드	메시지	설명
SQL1882N	"Nickname" 옵션 "COLUMN_DELIMITER"는 "<nickname_name>"의 "<delimiter>"로 설정될 수 없습 니다.	컬럼 분리문자의 길이가 한 자를 초과합니다. 옵션 을 한 자로 재정의하십시오. 그런 다음 SQL 명령 을 재실행하십시오.
SQL1882N	"Nickname" 옵션 "KEY_COLUMN"은 "<nickname_name>"의 "<column_name>"으로 설정될 수 없습니다.	키 컬럼으로 선택된 컬럼이 이 별칭에 대해 정의되 어 있지 않습니다. KEY_COLUMN 옵션을 이 별 칭의 정렬된 컬럼 중 하나로 정정한 다음 SQL 명 령을 재발행하십시오.
SQL1882N	"Nickname" 옵션 "VALIDATE_DATA_FILE"은 "<nickname_name>"의 "<option_value>"로 설정될 수 없 습니다.	옵션 값이 유효하지 않습니다. 유효한 값은 "Y" 또 는 "N"입니다. 옵션을 정정하고 별칭을 다시 등록 하십시오.
SQL1883N	"<option_name>"은 "<object_name>"의 필수 "<component>" 옵션입니다.	SQL문에 랩퍼에 대한 필수 옵션이 누락되었습니 다. 필수 옵션을 추가하고 SQL문을 다시 제출하 십시오.
SQL30090N	응용프로그램 실행 환경에 대해 유효하지 않은 조작입니다. 이유 코드 = "21".	Passthru 세션을 시도했습니다. 테이블 구조 파일 랩퍼가 passthru 세션을 지원하지 않습니다.

---

## 제4장 데이터 소스로서 Documentum 사용

이 장에서는 다음에 대해 설명합니다.

- Documentum 데이터 소스
- 연합 시스템에 Documentum 데이터 소스를 추가하는 방법
- Documentum 데이터 소스에 대해 조회를 수행하는 방법
- 반복 속성 정의를 이중화하는 방법
- CreateNicknameFile 유틸리티를 사용하여 별칭을 작성하는 방법
- 제한 및 고려사항
- 사용된 액세스 제어 모델
- Documentum 랩퍼에서 작업시 발생할 수 있는 메시지

---

### Documentum의 개념

Documentum은 체크인, 체크아웃, 작업흐름 및 버전 관리와 같은 문서 내용 및 속성의 관리를 제공하는 문서 관리 소프트웨어입니다. Documentum 제품은 관계형 데이터베이스의 맨 위에 빌드된 3-tier 클라이언트-서버 시스템입니다.

Docbase는 문서 내용, 속성, 관계, 버전, 해석, 형식, 작업흐름 및 보안을 저장하는 Documentum 저장소입니다. DQL(Documentum Query Language) 확장 SQL 다이얼렉트(dialect)가 Documentum 데이터를 조회하는 데 사용됩니다. Docbase는 Oracle 인스턴스 또는 DB2 데이터베이스에 문서 내용 파일을 더한 것과 같습니다. 메타데이터는 기본 관계형 데이터베이스 관리 시스템(RDBMS)에 저장되며, 내용은 데이터베이스의 2진 대형 오브젝트(BLOB)로 또는 서버 시스템의 파일 시스템 내에 저장된 파일로 저장됩니다. Documentum에 대한 자세한 정보는 Documentum 매뉴얼을 참조하십시오.

Documentum의 랩퍼는 Documentum 데이터 소스를 DB2 연합 시스템로 추가하게 합니다. Documentum 데이터 소스를 연합 시스템에 추가함으로써 SQL문을 사용하여 오브젝트 및 등록된 테이블을 Documentum Docbase에서 액세스하고 조

회할 수 있습니다. 그런 다음 원시(native) 데이터 소스로부터 데이터를 이동하지 않고 연합 시스템에서 다른 데이터 소스 데이터 소스와 이 데이터를 통합할 수 있습니다. Documentum 랩퍼는 Documentum 서버와 인터페이스하기 위해 클라이언트 라이브러리를 사용합니다. Documentum 랩퍼는 Documentum 서버의 EDMS 98(버전 3으로도 나타냄) 및 4i의 두 가지 버전에 대한 액세스를 제공합니다. 그림3에서는 Documentum 랩퍼 작업 방법을 설명합니다.

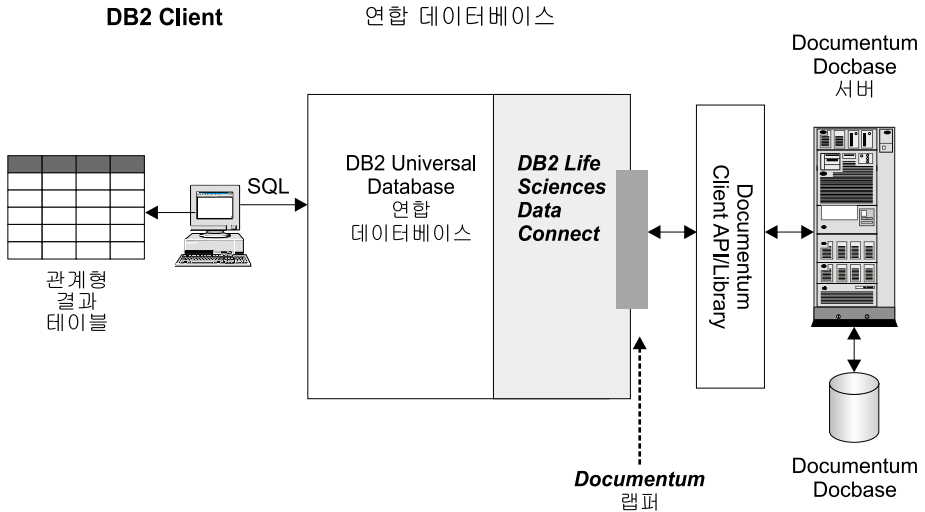


그림 3. Documentum 랩퍼 작업 방법

Documentum 랩퍼가 등록된 후에, Documentum Docbase 오브젝트 및 등록된 테이블을 관계 테이블로 매핑할 수 있습니다. 이것은 Docbase 속성을 DB2 관계형 테이블의 컬럼 이름으로 매핑하여 수행됩니다.

예를 들어, 25 페이지의 표5에서는 연관된 데이터와 함께 Documentum Docbase 기본 문서 유형, dm\_document에 대한 속성의 부속 집합을 나열합니다. 이 속성의 부속 집합이 중요하며, 이 속성을 연합 데이터베이스 시스템으로 연결하려고 함을 결정했습니다. 이 데이터 부속 집합의 이름을 DrugAB\_data라고 지정했습니다.

표 5. DrugAB\_data

제목	주제	작성자	키워드
The effect of drug A on rabbits	Drug A	Curran, L.	rabbits, drug A
Toxicity results for drug A	Drug A	Abelite, P., McMurtrey, K.	toxicity, drug A
Drug B interactions	Drug B	DeNiro, R., Stone, S.	interactions, drug B
Chemical structure of drug B	Drug B	Boyslim, F.	structure, drug B

Documentum 랩퍼를 등록한 후에, SQL문을 사용하여 데이터를 조회할 수 있습니다.

다음 조회는 주제가 Drug A인 제목과 작성자를 표시합니다. 결과 테이블은 표6에 표시됩니다.

```
SELECT title, authors
FROM drugAB_data
WHERE subject = 'Drug A'
```

표 6. 조회 결과

제목	작성자
The effect of drug A on rabbits	Curran, L.
Toxicity results for drug A	Abelite, P., McMurtrey, K.

## Documentum을 연함 시스템에 추가

연함 서버에 Documentum 데이터 소스를 추가하려면 다음을 수행하십시오.

1. 환경 변수를 설정하십시오.
2. Documentum 클라이언트 라이브러리에 링크하십시오.
3. DB2 인스턴스를 재순환시키십시오.
4. CREATE WRAPPER문을 사용하여 랩퍼를 등록하십시오.
5. 선택적: 조회 성능을 개선하려면 DB2\_DJ\_COMM 환경 변수를 설정하십시오.
6. CREATE SERVER문을 사용하여 서버를 등록하십시오.
7. CREATE USER MAPPING문을 사용하여 데이터 소스에 대한 사용자 액세스를 제공하십시오.

8. CREATE NICKNAME문을 사용하여 별칭을 등록하십시오.
9. CREATE FUNCTION문을 사용하여 사용자 정의 함수를 작성하십시오.

이 단계는 이 절에서 자세히 설명됩니다. 명령문은 DB2 명령행 처리기에서 수행될 수 있습니다. 일단 등록되면, 데이터 소스에 대한 조회를 수행할 수 있습니다.

## 1단계: 환경 변수 설정

Documentum Docbase에 대한 액세스는 Documentum 클라이언트의 dmcl.ini 파일에 의해 제어됩니다. DB2 인스턴스는 Documentum Docbase에 대한 액세스를 얻기 위해 Documentum 클라이언트의 dmcl.ini 파일에 설정된 환경 변수를 가져야 합니다.

환경 변수를 설정하려면, \$HOME/sqllib/cfg/에서 db2dj.ini 파일을 편집하고 다음 환경 변수 중 하나를 설정하십시오.

```
DOCUMENT=<path to location of dmcl.ini>  
DMCL_CONFIG=<path to location of dmcl.ini>/dmcl.ini
```

Documentum의 dmcl.ini 파일 위치에 대한 기본 경로는 /pkgs/documentum입니다. 두 행이 모두 포함되면, DMCL\_CONFIG가 사용됩니다.

주: DB2 인스턴스 보고서의 액세스 가능한 모든 Docbase에 대한 docbroker의 이름이 dmcl.ini 파일에 지정되었는지 확인하십시오.

## 2단계: Documentum 클라이언트 라이브러리에 대한 링크

Documentum 데이터 소스에 대한 액세스를 가능하게 하려면, DB2 연합 시스템이 클라이언트 라이브러리로 링크 편집되어야 합니다. 링크 편집 프로세스는 연합 서버가 통신하는 각 데이터 소스에 대한 랩퍼 라이브러리를 작성합니다. djxlinkDctm 스크립트를 수행하면 Documentum 랩퍼 라이브러리가 작성됩니다.

djxlinkDctm 스크립트를 수행하려면 다음을 수행 하십시오.

1. Documentum 클라이언트 라이브러리를 지시하도록 DM\_HOME 환경 변수를 설정하십시오. 예를 들어, 다음과 같습니다.

```
export DM_HOME=/pkgs/documentum/product/current
```

2. 다음 명령을 입력하십시오.

### 3단계: DB2 인스턴스 재순환

환경 변수가 프로그램에 설정되어 있는지 확인하려면, DB2 인스턴스를 재순환하십시오. 인스턴스 재순환시, 사용자가 작성한 변경사항을 적용하기 위해 DB2 인스턴스를 새로 고칩니다. 다음 명령을 실행하여 DB2 인스턴스를 재순환하십시오.

```
db2stop
db2start
```

### 4단계: 래퍼 등록

Documentum 래퍼를 등록하려면 CREATE WRAPPER문을 실행하십시오.

예를 들어, 기본 라이브러리 파일 liblsdctm.a에서 Dctm\_Wrapper라는 Documentum 래퍼를 작성하려면, 다음 명령문을 제출하십시오.

```
CREATE WRAPPER Dctm_Wrapper LIBRARY 'liblsdctm.a';
```

CREATE WRAPPER문에 대한 자세한 정보는 *DB2 SQL 참조서*를 참조하십시오.

### 5단계: 선택적: DB2\_DJ\_COMM 환경 변수 설정

성능을 개선하려면 DB2\_DJ\_COMM 환경 변수를 설정하십시오. 이 변수는 연합 서버가 초기화 중에 래퍼를 로드하는지 여부를 결정합니다. 『4단계: 래퍼 등록』에 지정된 래퍼에 해당하는 래퍼 라이브러리가 포함되도록 DB2\_DJ\_COMM 환경 변수를 설정하십시오. 예를 들어, 다음과 같습니다.

```
export DB2_DJ_COMM='liblsdctm.a'
```

등호(=)의 양쪽에 공백이 없게 하십시오.

DB2\_DJ\_COMM 환경 변수에 대한 자세한 정보는 *DB2 관리 안내서*를 참조하십시오.

### 6단계: 서버 등록

CREATE SERVER문을 사용하여 Documentum 서버를 등록하십시오.

예를 들어, 27 페이지의 『4단계: 래퍼 등록』에 작성된 Dctm\_Wrapper 래퍼에 대한 Dctm\_Server1이라는 서버가 있다고 가정하십시오. AIX에서 수행하며 Oracle 을 사용하여 데이터를 저장하는 Docbase가 서버에 들어 있다고 가정하십시오. 서버를 등록하려면 다음 명령을 실행하십시오.

```
CREATE SERVER Dctm_Server1
TYPE DCTM
VERSION '3'
WRAPPER Dctm_Wrapper
OPTIONS(
NODE 'Dctm_Docbase',
OS_TYPE 'AIX',
RDBMS 'ORACLE');
```

## 인수

**TYPE** 데이터 소스의 유형을 지정합니다. Documentum에 대한 유형은 DCTM 입니다. 이 인수는 필수입니다.

## VERSION

데이터 소스의 버전을 지정합니다. EDMS98에 대한 값은 '3'입니다. 4i 에 대한 값은 '4'입니다. 이 인수는 필수입니다.

## WRAPPER

이 서버와 연관된 래퍼의 이름을 지정합니다. 이 인수는 필수입니다.

## 옵션

### CONTENT\_DIR

GET\_FILE(), GET\_FILE\_DEL(), GET\_RENDITION() 및 GET\_RENDITION\_DEL() 사용자 정의 함수로 검색된 내용 파일을 저장하기 위해 지역적으로 액세스 가능한 루트 디렉토리의 이름을 지정합니다. 이들 함수를 사용할 수 있는 모든 사용자가 쓸 수 있어야 합니다. 기본값은 /tmp입니다. 이 옵션은 선택적입니다. 사용자 정의 함수에 대한 자세한 정보는 35 페이지의 『9단계: 사용자 정의 함수 등록』을 참조하십시오.

### NODE

Documentum Docbase의 실제 이름을 지정합니다. 이 옵션은 필수입니다.



## **OS\_TYPE**

Docbase 서버의 운영 체제를 지정합니다. 올바른 값은 AIX뿐입니다. 이 옵션은 필수입니다.

## **RDBMS\_TYPE**

Docbase가 사용하는 RDBMS를 지정합니다. 올바른 값은 ORACLE뿐입니다. 이 옵션은 필수입니다.

## **TRANSACTIONS**

서버 트랜잭션 모드를 지정합니다. 올바른 값은 다음과 같습니다.

- NONE -- 트랜잭션이 사용 가능하지 않습니다.
- QUERY -- 트랜잭션이 Dctm\_Query 메소드에 대해서만 사용 가능합니다.
- ALL -- 트랜잭션이 Dctm\_Query 메소드에 대해 사용 가능합니다. ALL 은 이 릴리스에서 QUERY와 동일한 기능을 가집니다.

기본값은 QUERY입니다. 이 옵션은 선택적입니다.

## **DEBUG\_FILE**

래퍼 활동 메시지를 수신하기 위해 파일의 완전한 이름을 지정합니다. 인스턴스 소유자가 작성할 수 있는 파일이어야 합니다. 파일의 지역 이름은 <server\_name>.log이어야 합니다. 기본값은 래퍼 활동 메시지를 로그하지 않음을 의미하는 ""입니다. 이 옵션은 선택적입니다.

## **DEBUG\_LEVEL**

로그될 디버그 메시지의 레벨을 지정합니다. 이 옵션은 DEBUG\_FILE이 지정되지 않으면 무시됩니다. 올바른 값은 다음과 같습니다.

- DEBUG\_ALL
- DEBUG\_INFO
- DEBUG\_WARN
- DEBUG\_ERROR - DEBUG\_FILE 옵션에 지정된 래퍼 활동 파일에 오류 메시지가 로그됩니다. 오류 메시지는 DEBUG\_LEVEL 옵션의 설정과 관계없이 DB2 오류 로그에 항상 로그됩니다.

- `DEBUG_CRITICAL` - 중요 메시지는 `DEBUG_FILE` 옵션에 지정된 래퍼 활동 파일에 로그됩니다. 중요 메시지는 `DEBUG_LEVEL` 옵션의 설정과 관계없이 DB2 오류 로그에 항상 로그됩니다.

기본값은 `DEBUG_WARN`입니다. 이 옵션은 선택적입니다.

`CREATE SERVER`문에 대한 자세한 정보는 *DB2 SQL* 참조서를 참조하십시오.

## 7단계: 사용자 맵핑

`CREATE USER MAPPING`문을 사용하여 이전에 정의된 서버로 사용자를 맵핑해야 합니다.

다음 예는 27 페이지의 『6단계: 서버 등록』에 작성된 `Dctm_Server1`에서 사용자 'Chuck'을 사용자 'Charles'로 맵핑합니다.

```
CREATE USER MAPPING FOR Chuck SERVER Dctm_Server1
OPTIONS(REMOTE_AUTHID 'Charles', REMOTE_PASSWORD 'Charles_pw');
```

또한 사용자 자신의 사용자 맵핑을 정의할 수 있습니다. 다음 예에서, `USER`는 현재 사용자를 의미하는 키워드로, 사용자 이름이 "USER"가 아닙니다.

```
CREATE USER MAPPING FOR USER SERVER Dctm_Server1
OPTIONS(REMOTE_AUTHID 'Lisa', REMOTE_PASSWORD 'Lisa_pw');
```

`CREATE USER MAPPING`문에 대한 자세한 정보는 *DB2 SQL* 참조서를 참조하십시오.

## 8단계: 별칭 등록

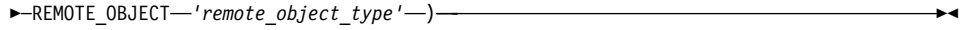
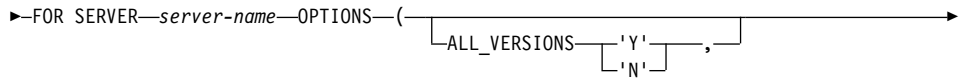
속성 이름을 DB2 관계형 컬럼 이름으로 맵핑하려면 `CREATE NICKNAME`문을 사용하여 관계가 있는 등록된 테이블이나 각 오브젝트 유형에 대한 각 Docbase의 별칭을 작성해야 합니다.

Documentum에 대한 `CREATE NICKNAME`문의 구문은 다음과 같습니다.

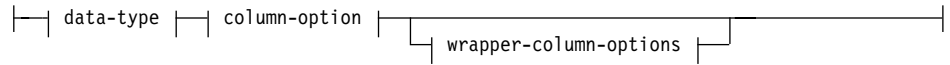
```

▶—CREATE NICKNAME—nickname—(—column-name—| column-information |—)——▶

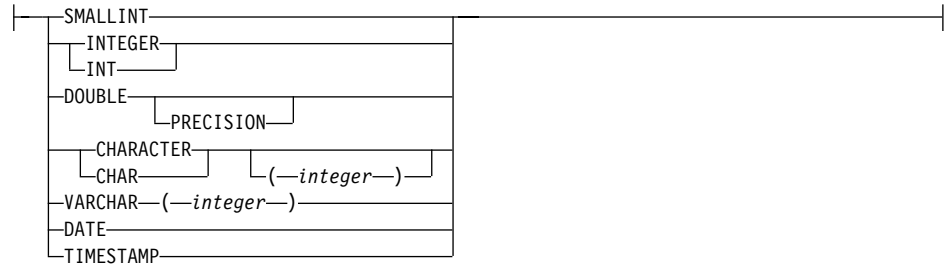
```



**column-information:**



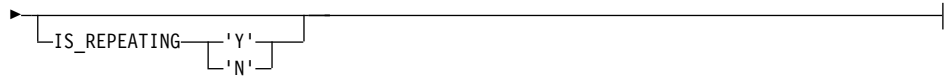
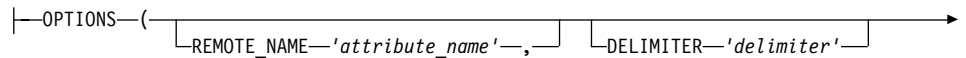
**data-type:**



**column-option:**



**wrapper-column-options:**



CREATE NICKNAME문에 대한 자세한 정보는 *DB2 SQL 참조서*를 참조하십시오.

### 컬럼 옵션

#### NOT NULL

TIMESTAMP로 정의된 컬럼을 제외한 모든 단일 값 컬럼은 NOT NULL로 정의되어야 합니다. 반복 속성은 별칭에서 NOT NULL로 정의되지 않아야 합니다.

#### 랩퍼 컬럼 옵션

랩퍼 컬럼 옵션 값은 작은따옴표 안에 있어야 합니다.

#### DELIMITER

반복 속성의 여러 값을 연결할 때 사용되는 분리문자 문자열을 지정합니다. 분리문자는 하나 이상의 문자일 수 있습니다. 기본 분리문자는 쉼표입니다. 이 옵션은 IS\_REPEATING 옵션이 'Y'로 설정된 컬럼에 대해서만 유효합니다. 이 옵션은 선택적입니다.

#### IS\_REPEATING

컬럼이 멀티 값인지를 표시합니다. 올바른 값은 'Y' 및 'N'입니다. 기본 값은 'N'입니다.

#### REMOTE\_NAME

상응하는 Documentum 속성 또는 컬럼의 이름을 지정합니다. 이 옵션은 원격 속성이나 컬럼 이름을 지역 DB2 컬럼 이름으로 맵핑합니다. 상응하는 기본값은 DB2 컬럼 이름입니다. 이 옵션은 선택적입니다.

#### 별칭 옵션

별칭 옵션 값은 작은따옴표 안에 있어야 합니다.

#### ALL\_VERSIONS

모든 오브젝트 버전의 검색 여부를 지정합니다. 올바른 값은 'y', 'Y', 'n' 및 'N'입니다. 기본값 'N'은 현재 오브젝트 버전만이 조회 처리에 포함됨을 의미합니다. 이 옵션은 IS\_REG\_TABLE = 'Y'일 때 유효합니다. 이 옵션은 선택적입니다.

## FOLDERS

논리적으로 결합되고 구문이 정확한 하나 이상의 Documentum FOLDER 술어가 들어 있는 문자열을 지정합니다. FOLDER 술어를 지정하면 이 별칭으로 표시된 문서 세트를 지정된 폴더에 있는 문서 세트로 제한합니다.

이 옵션 지정시, 작은따옴표로 FOLDERS 옵션의 전체 값을 묶고 문자열 내에서 작은따옴표 대신 큰따옴표를 사용하십시오.

예를 들어, 다음을 삽입하려는 경우

```
FOLDER('/Tools',DESCEND) OR FOLDER('/Cars')
```

다음 FOLDERS 옵션을 지정하십시오.

```
FOLDERS 'FOLDER("/Tools",DESCEND) OR FOLDER("/Cars")'
```

이 옵션은 IS\_REG\_TABLE = 'Y'일 때 유효합니다. 이 옵션은 선택적입니다.

## IS\_REG\_TABLE

REMOTE\_OBJECT 옵션으로 지정한 오브젝트가 Documentum 등록 테이블인지 여부를 지정합니다. 올바른 값은 'y', 'Y', 'n' 및 'N'입니다. 기본값은 'N'입니다. 이 옵션은 선택적입니다.

## REMOTE\_OBJECT

별칭과 연관된 Documentum 오브젝트 유형의 이름을 지정합니다. 이름은 Documentum 오브젝트 유형 또는 등록 테이블일 수 있습니다. 등록 테이블의 경우, 테이블 소유자의 이름이 접두부로 표시되어야 합니다. 등록된 테이블이 Docbase 소유자에 속하는 경우, 소유자 이름에 대해 dm\_dbo가 사용될 수 있습니다. 이 옵션은 필수입니다.

## CREATE NICKNAME의 예

다음 CREATE NICKNAME문은 별칭 std\_doc를 정의합니다. Std\_doc는 dm\_document의 오브젝트 유형과 함께 Documentum Docbase와 연관됩니다. 34 페이지의 표7은 CREATE NICKNAME문을 구성하는 데 사용되는 DB2 관계 컬럼 이름으로 Documentum 속성 및 데이터 유형을 맵핑합니다.

표 7. *std\_doc* 별칭에 대한 DB2 컬럼으로 *Documentum* 속성 맵핑

Documentum 속성 이름	Documentum 데이터 유형	DB2 컬럼 이름	DB2 데이터 유형	반복 여부	설명
object_name	string(255)	object_name	varchar	아니오	오브젝트의 사용자 정의 이름
r_object_id	ID	object_id	char(16)	아니오	작성 시간에 설정된 이 오브젝트에 대한 고유한 오브젝트 식별자.
r_object_type	string(32)	object_type	varchar	아니오	오브젝트가 작성될 때 설정된 오브젝트의 유형.
title	string(255)	title	varchar	아니오	오브젝트의 사용자 정의 제목
subject	string(128)	subject	varchar	아니오	오브젝트의 사용자 정의 주제
authors	string(32)	author	varchar	예	오브젝트에 대한 작성자의 사용자 정의 목록
keywords	string(32)	keyword	varchar	예	오브젝트에 대한 사용자 정의 키워드 목록
r_creation_date	time	creation_date	timestamp	아니오	오브젝트가 작성된 날짜 및 시간
r_modify_date	time	modified_date	timestamp	아니오	오브젝트가 최종 수정된 날짜 및 시간
a_status	string(16)	status	varchar	아니오	라우터 태스크가 이송될 때 서버에 의해 설정됩니다. 값은 라우터 상태에서 <code>attached_task_status</code> 로 지정된 값에서 얻습니다.
a_content_type	string(32)	content_type	varchar	아니오	오브젝트 내용의 파일 형식.
r_content_size	double	content_size	integer	아니오	내용의 바이트 수. 여러 페이지 문서의 경우, 이 속성은 문서와 연관된 첫 번째 내용의 크기를 기록합니다.

표 7. std\_doc 별칭에 대한 DB2 컬럼으로 Documentum 속성 맵핑 (계속)

Documentum 속성 이름	Documentum 데이터 유형	DB2 컬럼 이름	DB2 데이터 유형	반복 여부	설명
owner_name	string(32)	owner_name	varchar	아니오	오브젝트 소유자의 이름(오브젝트를 작성한 사용자).

34 페이지의 표7은 다음 CREATE NICKNAME문으로 변환합니다.

```
CREATE NICKNAME std_doc (
  object_name varchar(255) not null,
  object_id char(16) not null OPTIONS(REMOTE_NAME 'r_object_id'),
  object_type varchar(32) not null OPTIONS(REMOTE_NAME 'r_object_type'),
  title varchar(255) not null,
  subject varchar(128) not null,
  author varchar(32) OPTIONS(REMOTE_NAME 'authors', IS_REPEATING 'Y'),
  keyword varchar(32) OPTIONS(REMOTE_NAME 'keywords', IS_REPEATING 'Y'),
  creation_date timestamp OPTIONS(REMOTE_NAME 'r_creation_date'),
  modified_date timestamp OPTIONS(REMOTE_NAME 'r_modify_date'),
  status varchar(16) not null OPTIONS(REMOTE_NAME 'a_status'),
  content_type varchar(32) not null OPTIONS(REMOTE_NAME 'a_content_type'),
  content_size integer not null OPTIONS(REMOTE_NAME 'r_content_size'),
  owner_name varchar(32))
FOR SERVER Dctm_Server2 OPTIONS (REMOTE_OBJECT 'dm_document', IS_REG_TABLE 'N')
```

CREATE NICKNAME문을 제출한 후에, 별칭 std\_doc를 사용하여 연합 시스템을 조회할 수 있습니다. 또한 연합 시스템의 다른 별명 및 테이블로 std\_doc 별칭을 조인할 수 있습니다.

CreateNicknameFile 유틸리티를 사용하여 Documentum 유형을 자동으로 DB2 유형으로 맵핑하고 초기 CREATE NICKNAME문을 작성할 수 있습니다.

CreateNicknameFile 유틸리티에 대한 자세한 정보는

46 페이지의 『CreateNicknameFile 유틸리티』를 참조하십시오.

## 9단계: 사용자 정의 함수 등록

여러 사용자 정의 함수를 등록하려면 CREATE FUNCTION문을 사용해야 합니다. 전체 텍스트 검색 및 조회 내의 문서 내용 검색과 같이 Documentum의 고유한 일부 성능을 액세스하기 위해 이 함수를 사용할 수 있습니다.

술어에 대한 사용자 정의 함수가 표8에서 나열됩니다. SELECT절에서만 지정될 수 있는 사용자 정의 함수가 41 페이지의 표9에 나열되어 있습니다. 명령문에 검색 절이 들어 있을 때의 SELECT문에 대한 사용자 정의 함수가 43 페이지의 표10에 나열되어 있습니다.

DB2는 BOOLEAN 데이터 유형을 지원하지 않습니다. 그러므로 유효한 SQL문을 작성하려면, 각 사용자 정의 함수 값이 명시적으로 테스트되어야 합니다. 래퍼 구현만이 지정된 테스트 비교 연산자와 관계없이 "<function>() = 1"에 대한 의미론을 지원합니다.

주: TOPIC 함수에 대한 참조는 Verity, Inc의 써드 파티 전체 텍스트 색인 시스템의 일부로서 제공된 Documentum 함수에 대한 것입니다.

표 8. 술어에 대한 사용자 정의 함수

함수 이름	설명
ANY_EQ(arg1, arg2)	<p>지정된 값과 동일한 값에 대한 반복 속성을 테스트합니다. 다음 두 개의 필수 인수를 갖습니다.</p> <p><b>arg1</b>    반복 속성을 나타내는 컬럼의 이름을 지정합니다.</p> <p><b>arg2</b>    비교되는 값을 지정합니다.</p> <p>예를 들어, 다음과 같습니다.</p> <pre>... WHERE DCTM.ANY_EQ(authors, 'Dave Winters')=1</pre>
ANY_NE(arg1, arg2)	<p>지정된 값과 동일하지 않은 값에 대한 반복 속성을 테스트합니다. 다음 두 개의 필수 인수를 갖습니다.</p> <p><b>arg1</b>    반복 속성을 나타내는 컬럼의 이름을 지정합니다.</p> <p><b>arg2</b>    비교되는 값을 지정합니다.</p> <p>예를 들어, 다음과 같습니다.</p> <pre>... WHERE DCTM.ANY_NE(authors, 'Dave Winters')=1</pre>



표 8. 술어에 대한 사용자 정의 함수 (계속)

함수 이름	설명
ANY_LT(arg1, arg2)	<p>지정된 값 미만의 모든 값에 대한 반복 속성을 테스트합니다. 다음 두 개의 필수 인수를 갖습니다.</p> <p><b>arg1</b> 반복 속성을 나타내는 컬럼의 이름을 지정합니다.</p> <p><b>arg2</b> 비교되는 값을 지정합니다.</p> <p>예를 들어, 다음과 같습니다.</p> <pre>... WHERE DCTM.ANY_LT(num approvers,4)=1</pre>
ANY_GT(arg1, arg2)	<p>지정된 값보다 큰 모든 값에 대한 반복 속성을 테스트합니다. 다음 두 개의 필수 인수를 갖습니다.</p> <p><b>arg1</b> 반복 속성을 나타내는 컬럼의 이름을 지정합니다.</p> <p><b>arg2</b> 비교되는 값을 지정합니다.</p> <p>예를 들어, 다음과 같습니다.</p> <pre>... WHERE DCTM.ANY_GT(num approvers,3)=1</pre>
ANY_LE(arg1, arg2)	<p>지정된 값 이하의 모든 값에 대한 반복 속성을 테스트합니다. 다음 두 개의 필수 인수를 갖습니다.</p> <p><b>arg1</b> 반복 속성을 나타내는 컬럼의 이름을 지정합니다.</p> <p><b>arg2</b> 비교되는 값을 지정합니다.</p> <p>예를 들어, 다음과 같습니다.</p> <pre>... WHERE DCTM.ANY_LE(num approvers,2)=1</pre>
ANY_GE(arg1, arg2)	<p>지정된 값 이상의 모든 값에 대한 반복 속성을 테스트합니다. 다음 두 개의 필수 인수를 갖습니다.</p> <p><b>arg1</b> 반복 속성을 나타내는 컬럼의 이름을 지정합니다.</p> <p><b>arg2</b> 비교되는 값을 지정합니다.</p> <p>예를 들어, 다음과 같습니다.</p> <pre>... WHERE DCTM.ANY_GE(num approvers,1)=1</pre>

표 8. 술어에 대한 사용자 정의 함수 (계속)

함수 이름	설명
ANY_IN(arg1, arg2 - arg11)	<p>지정된 값 목록에서 10개의 값에 대한 반복 속성을 테스트합니다. 동일한 데이터 유형의 3-11개의 인수를 갖습니다.</p> <p><b>arg1</b>    반복 속성을 나타내는 컬럼의 이름을 지정합니다.</p> <p><b>arg2-arg11</b>                      실행으로 구분되는 비교 값 목록을 지정합니다.</p> <p>예를 들어, 다음과 같습니다.</p> <pre>... WHERE DCTM.ANY_IN(authors,'Crick','Watson')=1</pre>
ANY_LIKE(arg1, arg2)	<p>지정된 값과 같은 모든 값에 대한 반복 속성을 테스트합니다. 다음 두 개의 필수 인수를 갖습니다.</p> <p><b>arg1</b>    반복 속성을 나타내는 컬럼의 이름을 지정합니다.</p> <p><b>arg2</b>    작은따옴표 안의 부속 문자열과 비교 중인 패턴을 지정합니다.</p> <p>예를 들어, 다음과 같습니다.</p> <pre>... WHERE DCTM.ANY_LIKE(authors,'Dave Win%')=1 OR DCTM.ANY_LIKE(keywords,'%\_%')=1</pre> <p>주: Escape절은 ANY_LIKE() 술어에서 지원되지 않습니다.</p>
ANY_NOT_LIKE (arg1, arg2)	<p>지정된 값과 같지 않은 모든 값에 대한 반복 속성을 테스트합니다. 다음 두 개의 필수 인수를 갖습니다.</p> <p><b>arg1</b>    반복 속성을 나타내는 컬럼의 이름을 지정합니다.</p> <p><b>arg2</b>    작은따옴표 안의 부속 문자열과 비교 중인 패턴을 지정합니다.</p> <p>예를 들어, 다음과 같습니다.</p> <pre>... WHERE DCTM.ANY_NOT_LIKE(authors,'Dave Win%')=1 OR DCTM.ANY_NOT_LIKE(keywords,'%\_%')=1</pre> <p>주: Escape절은 ANY_NOT_LIKE() 술어에서 지원되지 않습니다.</p>

표 8. 술어에 대한 사용자 정의 함수 (계속)

함수 이름	설명
ANY_NULL(arg)	<p>IS NULL에 대한 반복 속성을 테스트합니다. 반복 속성이나 단일 값 DATE 또는 TIMESTAMP 속성의 이름인 하나의 필수 인수를 갖습니다.</p> <p>예를 들어, 다음과 같습니다.</p> <pre>... WHERE DCTM.ANY_NULL(authors)=1</pre>
ANY_NOT_NULL(arg)	<p>IS NOT NULL에 대한 반복 속성을 테스트합니다. 반복 속성의 이름인 하나의 필수 인수를 갖습니다.</p> <p>예를 들어, 다음과 같습니다.</p> <pre>... WHERE DCTM.ANY_NOT_NULL(authors)=1</pre>
ANY_SAME_INDEX (arg1 - arg10)	<p>각 속성의 동일한 색인에서 값에 대한 반복 속성을 테스트합니다. 2 - 10개의 다른 ANY_xx() 함수를 갖습니다.</p> <p>다음 예는 UCD와 관계되지 않은 Ken이라는 최소한 한 명의 작성자를 문서가 갖는지 여부를 검사합니다.</p> <pre>... WHERE DCTM.ANY_SAME_INDEX( ANY_EQ(author_name,'Ken'), DCTM.ANY_NE(author_affiliation,'UCD'))</pre>
CABINET(arg) 및 CABINET_TREE(arg)	<p>Docbase 캐비닛의 완전한 이름인 필수 인수를 하나 갖습니다.</p> <p>예를 들어, 다음과 같습니다.</p> <pre>... WHERE DCTM.CABINET('/Tools')=1 ... WHERE DCTM.CABINET_TREE('/MyDocs')=1</pre> <p>CABINET 및 CABINET_TREE의 여러 인스턴스를 사용하여 여러 캐비닛을 지정하십시오.</p> <p>예를 들어, 다음과 같습니다.</p> <pre>... WHERE DCTM.CABINET('/Tools')=1 OR DCTM.CABINET_TREE('/Parts')=1</pre>

표 8. 술어에 대한 사용자 정의 함수 (계속)

함수 이름	설명
FOLDER(arg) 및 FOLDER_TREE(arg)	<p>Docbase 폴더 또는 캐비닛의 완전한 이름인 필수 인수를 하나 갖습니다.</p> <p>예를 들어, 다음과 같습니다.</p> <pre>... DCTM.FOLDER('/Tools/Drills')=1 ... DCTM.FOLDER_TREE('/MyDocs/WhitePapers')=1</pre> <p>FOLDER 및 FOLDER_TREE의 여러 인스턴스를 사용하여 여러 폴더를 지정하십시오.</p> <p>예를 들어, 다음과 같습니다.</p> <pre>... DCTM.FOLDER('/Tools/Drills')=1 OR DCTM.FOLDER_TREE('/Animals/Horses')=1</pre>
USER(1)	<p>현재 사용자의 Documentum 작성자 ID와 값을 비교합니다. 1이어야 하는 더미 인수를 갖습니다.</p> <p>예를 들어, 다음과 같습니다.</p> <pre>... WHERE approver = DCTM.USER(1)</pre> <p>주: Documentum 작성자 ID가 DB2 작성자 ID에 해당하게 하려면, CREATE USER MAPPING문을 사용하십시오. 사용자 맵핑에 대한 자세한 정보는 30 페이지의 『7단계: 사용자 맵핑』을 참조하십시오.</p>
SEARCH_WORDS(arg)	<p>AND, OR 또는 NOT으로 구분되며 작은따옴표로 묶인 각 단어의 목록인 하나의 필수 문자열 인수를 갖고 괄호를 사용하여 우선 순위를 제어합니다. 단어는 공백을 포함할 수 없으며 작은따옴표로 묶여야 합니다.</p> <p>예를 들어, 다음과 같습니다.</p> <pre>... DCTM.SEARCH_WORDS(''yeast'' AND (''bread'' OR ''cake'') AND NOT ''wedding'' )=1</pre>
SEARCH_TOPIC(arg)	<p>Documentum 및 Verity verbatim으로 패스되는 Verity TOPIC 조 회문인 하나의 필수 인수를 갖습니다.</p> <p>예를 들어, 다음과 같습니다.</p> <pre>... WHERE DCTM.SEARCH_TOPIC('"quick"')=1</pre>

표9에서는 SELECT절에 대한 사용자 정의 함수를 나열합니다.

표 9. SELECT절에 대한 사용자 정의 함수

함수 이름	설명
GET_FILE(1)	<p>컬럼 값 외에 현재 행에 대한 내용 파일을 검색합니다. 1이어야 하는 더미 인수를 갖습니다.</p> <p>해당 오브젝트 ID에 대한 내용 파일은 행에 대해 검색되며 지역 저장에 해당 오브젝트 이름으로 제공되므로 r_object_id 및 object_name은 SELECT 목록에서도 지정되어야 합니다. 내용 파일에 대한 확장자는 Documentum 형식 이름입니다. 동일한 이름의 파일이 존재하는 경우, 겹쳐쓰여집니다.</p> <p>GET_FILE(1)은 오브젝트의 기본 형식을 가져오려고 시도합니다. 행에서의 해당 값은 오브젝트의 a_content_type입니다. 오브젝트가 내용 파일을 갖지 않는 경우 해당 값은 문자열 "no_content"입니다.</p> <p>예를 들어, 다음과 같습니다.</p> <pre>SELECT object_name, r_object_id, DCTM.GET_FILE(1) FROM ...</pre> <p>내용 파일은 서버의 CONTENT_DIR 옵션으로 지정된 서버 디렉토리에 위치합니다. 또한 사용자의 DB2 지역 이름으로 지정된 서브디렉토리에 위치합니다. 서브디렉토리가 없는 경우 작성됩니다.</p> <p>해당 확장자는 문서의 형식 유형에 대한 Docbase에 정의된 관련 DOS 확장자입니다. 예를 들어, MS Word 문서에 대해서는 ".doc"입니다.</p> <p>문자열 "no_content" 또는 파일의 완전한 이름을 리턴합니다.</p>
GET_FILE_DEL(1)	<p>GET_FILE_DEL(1)이 해당 조회에서 이전 행에 대해 검색된 파일이 있는 경우 우선 이 파일을 삭제하는 것만 제외하고 이 함수는 GET_FILE(1)과 동일합니다. 1이어야 하는 더미 인수를 갖습니다. 문자열 "no_content" 또는 파일의 완전한 이름을 리턴합니다.</p>

표 9. SELECT절에 대한 사용자 정의 함수 (계속)

함수 이름	설명
GET_RENDITION(arg)	<p>해당 렌디션(rendition)의 내용 파일, 다른 형식으로 된 원래 문서의 사본을 컬럼 값 이외에 현재 행에 대해서도 검색합니다. 원하는 렌디션(rendition)의 이름인 하나의 인수를 갖습니다.</p> <p>해당 오브젝트 ID에 대한 내용 파일은 행에 대해 검색되며 지역 저장에 해당 오브젝트 이름으로 제공되므로 r_object_id 및 object_name은 SELECT 목록에서도 지정되어야 합니다. 내용 파일에 대한 확장자는 Documentum 형식 이름입니다. 동일한 이름의 파일이 존재하는 경우, 겹쳐쓰여집니다.</p> <p>GET_RENDITION()은 오브젝트의 이름 지정된 렌디션(rendition)을 가져오려고 시도합니다. 오브젝트가 내용 파일을 갖지 않아 해당 값이 문자열 "no_content"인 경우나 렌디션(rendition)이 없어 해당 값이 문자열 "not_found"인 경우를 제외하고는 해당 행의 값은 오브젝트의 a_content_type입니다.</p> <p>예를 들어, 다음과 같습니다.</p> <pre>SELECT object_name, r_object_id, DCTM.GET_RENDITION('pdf') FROM ...</pre> <p>내용 파일은 서버의 CONTENT_DIR 옵션으로 지정된 서버 디렉토리에 위치합니다. 또한 사용자의 DB2 지역 이름으로 지정된 서브디렉토리에 위치합니다. 서브디렉토리가 없는 경우 작성됩니다.</p> <p>해당 확장자는 문서의 형식 유형에 대한 Docbase에 정의된 관련 DOS 확장자입니다. 예를 들어, MS Word 문서에 대해서는 ".doc"입니다.</p> <p>문자열 "no_content" 또는 파일의 완전한 이름을 리턴합니다.</p>
GET_RENDITION_DEL(arg)	<p>GET_RENDITION_DEL()이 해당 조회에서 이전 행에 대해 검색된 파일이 있는 경우 우선 이 파일을 삭제하는 것만 제외하고 이 함수는 GET_RENDITION()과 동일합니다. 문자열 "no_content" 또는 파일의 완전한 이름을 리턴합니다.</p>

43 페이지의 표10에서는 검색 절이 들어 있는 조회에서 SELECT절에 대한 사용자 정의 함수를 나열합니다.

표 10. 검색 절이 들어 있는 조회에서 SELECT절에 대한 사용자 정의 함수

함수 이름	설명
HITS(1)	<p>검색 기준이 일치하는 문서에서 갯수를 표시하는 정수를 리턴합니다.</p> <p>예를 들어, 다음과 같습니다.</p> <pre>SELECT r_object_id, object_name, DCTM.HITS(1) FROM std_doc DCTM.SEARCH WORDS('workflow' OR 'flowchart')</pre> <p>리턴되는 각 문서의 경우, 문서 내용 안에서 "workflow" 및 "flowchart" 단어의 갯수를 합해 HITS 값으로 리턴합니다.</p> <p>문서가 오직 하나의 내용 파일을 가지면 HITS 함수가 적절합니다. 이것은 일반적인 경우입니다. 이 키워드는 SELECT문에 대한 WHERE절 조건에서 사용될 수 있습니다. 그러나 SELECT절에서도 지정되어야 합니다.</p>
SCORE(1)	<p>문서의 관련 순위를 리턴합니다.</p> <p>Documentum의 ACCRUE 개념 연산자와 결합하여 이 사용자 정의 함수를 사용하십시오. 둘다 각 리턴된 문서에서 찾은 지정된 단어의 갯수를 나타내는 숫자를 리턴합니다.</p> <p>예를 들어, 다음과 같습니다.</p> <pre>SELECT object name, DCTM.SCORE(1) FROM std_doc DCTM.SEARCH TOPIC('&lt;ACCRUE&gt;("document", "management", "workflow")') WHERE DCTM.SCORE(1) &gt;=75</pre> <p>명령문은 해당 내용에서 지정된 단어의 2 - 3개를 갖는 모든 문서를 리턴합니다. 문서가 오직 하나의 단어를 갖는 경우, 점수 50이 지정되므로 WHERE절 기준에 맞지 않아 리턴하지 않습니다. 세 개의 단어 중 두 개를 찾은 경우, 문서에 75의 점수가 지정됩니다. 세 개의 모든 단어를 찾은 경우, 문서의 점수는 88입니다.</p> <p>SCORE(1) 함수는 하나의 내용 파일을 갖는 문서용으로 사용됩니다. 이것은 일반적인 경우입니다.</p> <p>WHERE에 SEARCH_WORDS() 또는 SEARCH_TOPIC() 함수가 들어 있는 경우에만 SCORE(1)이 SELECT절에 있을 수 있습니다. WHERE절에서, ACCRUE 개념 연산자와 결합하여 사용됩니다.</p> <p>ACCRUE 개념 연산자에 대한 정보는 Documentum 문서를 참조하십시오.</p>

## 사용자 정의 함수 문자열 인수 규칙

문자열로 패스되는 모든 인수는 다음 규칙을 지켜야 합니다.

- 각 문자열은 작은따옴표로 묶입니다.
- 문자열 내의 작은따옴표는 두 개의 작은따옴표로 표시됩니다.

## CREATE FUNCTION문으로 사용자 정의 함수 지정

모든 사용자 정의 함수는 스키마 이름 DCTM으로 등록되어야 합니다. 각 함수의 완전한 이름은 DCTM.<function\_name>입니다.

다음 예는 ANY\_EQ 사용자 정의 함수를 등록합니다.

```
CREATE FUNCTION DCTM.ANY_EQ (CHAR(), CHAR()) RETURNS INTEGER AS TEMPLATE
```

Documentum 랩퍼를 설치하는 각 DB2 데이터베이스에 대해 각 사용자 정의 함수를 한 번 등록해야 합니다.

사용자 정의 함수 등록을 돕기 위해, 함수 파일 create\_fuction\_mappings.ddl이 sqllib/samples/lifesci 디렉토리에서 제공됩니다. 이 파일에는 각 사용자 정의 함수에 대한 정의가 들어 있습니다. Documentum 랩퍼가 설치된 각 DB2 데이터베이스에 대한 사용자 정의 함수를 등록하기 위해 이 ddl 파일을 수행할 수 있습니다.

## 조회에서 사용자 정의 함수 사용

다음 예는 조회에서 사용자 정의 함수의 사용을 예시합니다.

‘Dave Winters’라는 하나 이상의 작성자를 갖는 문서에 대한 std\_doc 별칭에서 오브젝트 이름 및 작성자를 표시하려면 다음을 수행하십시오.

```
SELECT object_name,authors FROM std_doc  
WHERE DCTM.ANY_EQ(authors,'Dave Winters')=1
```

‘Dave Winters’ 또는 ‘Jon Doe’라는 하나 이상의 작성자를 갖는 문서에 대한 std\_doc 별칭에서 오브젝트 이름 및 작성자를 표시하려면 다음을 수행하십시오.

```
SELECT object_name,authors FROM std_doc  
WHERE DCTM.ANY_IN(authors,'Dave Winters','John Doe')=1
```



오브젝트 이름 및 r\_object\_id를 표시하고, 작성자 컬럼에 'Dave Win%'와 같은 문자열이 들어 있는 문자에 대한 std\_doc 별칭에서 내용 파일을 검색하려면 다음을 수행하십시오.

```
SELECT object_name,r_object_id,DCTM.GET_FILE(1) FROM std_doc
WHERE DCTM.ANY_LIKE(authors,'Dave Win%')=1
```

오브젝트 이름 및 r\_object\_id를 표시하고, 제목 컬럼에 'IBM DiscoveryLink%'와 같은 문자열이 들어 있는 문서에 대한 std\_doc 별칭에서 오브젝트 이름 및 r\_object\_id를 표시하고 'pdf' rendition의 내용 파일을 검색하려면, 다음을 수행하십시오.

```
SELECT object_name,r_object_id,DCTM.GET_RENDITION('pdf') FROM std_doc
WHERE title like 'IBM DiscoveryLink%'
```

CREATE FUNCTION문에 대한 자세한 내용은 *DB2 SQL 참조서*를 참조하십시오.

---

## 조회 실행

래퍼를 등록한 후에, Documentum 데이터 소스에 대해 SQL 조회를 수행할 수 있습니다. 이 절에서는 조회의 몇가지 예를 제공합니다.

다음 조회는 'Test Document'라는 문서에 대한 모든 Docbase 문서를 표시합니다.

```
SELECT object_name
FROM std_doc
WHERE object_name='Test Document';
```

다음 조회는 사용자 정의 함수 ANY\_EQ를 사용하여 작성자 중 한 명이 'Joe Doe'인 모든 문서를 표시합니다.

```
SELECT *
FROM std_doc
WHERE DCTM.ANY_EQ(author,'Joe Doe')=1
```

다음 조회는 FOLDER\_TREE 함수 및 SEARCH\_WORDS 함수를 사용하여 텍스트 "protein"이 들어 있는 승인된 캐비닛에서 모든 문서를 찾습니다.

```

SELECT object_name
FROM std_doc
WHERE DCTM.FOLDER_TREE('/Approved')=1
      AND DCTM.SEARCH_WORDS('protein')=1

```

다음 조회는 GET\_FILE, FOLDER\_TREE 및 ANY\_IN 사용자 정의 함수를 사용하여 나열된 작성자를 갖는 승인된 캐비닛의 모든 문서에 대한 내용이 위치되는 DB2 서버에서 파일의 이름을 검색합니다.

```

SELECT object_name, object_id, DCTM.GET_FILE(1)
FROM std_doc
WHERE DCTM.FOLDER_TREE('/Approved')=1
      AND DCTM.ANY_IN(author, 'Mary Black', 'Joe Carson', 'Peter Miller')=1

```

---

## CreateNicknameFile 유틸리티

자유롭게 다운로드하여 사용할 수 있는 CreateNicknameFile이라는 Docbasic 유틸리티를 사용하여 임의의 Docbase 오브젝트나 등록된 테이블의 완전한 정의가 들어 있는 ASCII 파일을 작성할 수 있습니다. 다음을 위해 출력 파일을 편집할 수 있습니다.

- 컬럼 및 속성에 대한 지역 이름을 정의. 지역 및 원격 이름은 처음에 Docbase에 알려진 이름입니다.
- 원치않는 컬럼 및 속성을 삭제. 사전정의된 Documentum 문서 유형(dm\_document)은 EDMS98에서 59개의 속성을 가지며 4i에서는 76개의 속성을 갖습니다. 대부분의 속성에는 하위 레벨 문서 관리 및 응용프로그램 개발을 위한 메타데이터가 들어 있습니다. 관심 대상이 아닌 속성을 삭제하면 성능에 영향을 미치지 않고 SELECT \* SQL문을 더 유용하게 만들 수 있습니다.
- 특정 Documentum 폴더로 이 별칭에 대한 검색을 제한하기 위해 FOLDERS 옵션의 값을 추가.
- 원하는 경우 TIMESTAMP에 대한 DATE 매핑을 변경. 유틸리티는 가장 유용해 보이는 DQL DATE에서 DB2 DATE로의 매핑을 생성합니다.
- 응용프로그램에 따라 CHAR 매핑을 VARCHAR로, 또는 역으로 변경.

Docbase에서 유틸리티를 설치하고 Documentum Windows 그래픽 사용자 인터페이스로부터 실행해야 합니다. 유틸리티가 생성하는 파일은 설치된 Docbase에 고유합니다.

## CreateNicknameFile 유틸리티 설치

유틸리티를 설치하려면 다음을 수행하십시오.

1. <http://www.ibm.com/software/data/db2/lifesciencesdataconnect/>에서 DB2 Life Sciences Data Connect 제품 웹사이트의 다운로드 섹션에서 CreateNicknameFile 유틸리티를 다운로드하십시오.
2. CreateNicknameFile.txt라는 유틸리티를 가져오려면 EDMS98 Workspace 그래픽 사용자 인터페이스 또는 4i Desktop Client를 사용하십시오. Docbase 캐비닛 또는 폴더로 유틸리티를 가져오거나, 원하는 이름을 부여할 수 있습니다.
3. 새로 가져온 CreateNicknameFile.txt 오브젝트에 대한 등록 정보 대화 상자에서 사용자가 실행할 수 있는 상자를 체크하십시오.

## CreateNicknameFile 유틸리티 구성

유틸리티를 설치 후 구성하려면 다음을 수행하십시오.

1. 유틸리티의 아이콘을 더블 클릭하여 유틸리티를 수행하십시오.
2. Documentum 문서/오브젝트 유형 이름을 입력하십시오. 기본값은 dm\_document입니다.

주: 등록된 테이블에 대한 별칭 파일을 작성하는 것이 필요한 경우 이름으로 dm\_registered를 지정하십시오. dm\_registered를 지정하는 경우, <owner>.<table\_name> 형식으로 완전한 테이블 이름에 대해 프롬프트됩니다. Docbase 소유자가 테이블을 소유하는 경우(일반적인 경우) 소유자 이름에 대해 dm\_dbo를 사용할 수 있습니다.

유틸리티는 등록된 테이블에 대한 별칭 이름으로 이름 지정 규칙을 따릅니다. 규칙은 "등록된 테이블"을 나타내기 위해 테이블 이름에 "rt\_" 접두부를 붙입니다. 이 규칙을 사용하지 않으려는 경우 유틸리티가 제안한 별칭을 변경할 수 있습니다.

3. 작성 중인 별칭과 연관된 서버 이름을 입력하십시오.
4. 별칭의 이름을 입력하십시오.

별칭 이름은 설명적이어야 하며 DB2 인스턴스 내에서 고유해야 합니다. 동일한 <object\_type>이 여러 서버를 정의하는 데 필요할 수 있으므로 유틸리티는

<server\_name>.<object\_type>의 이름 지정 규칙을 따릅니다. 이 규칙을 따르지 않으려는 경우 유틸리티가 제안한 별칭을 변경할 수 있습니다.

##### 5. 출력 파일의 이름을 입력하십시오.

기본값은 C:\Temp\nickname.txt입니다. 출력 파일을 받은 디렉토리는 이미 존재해야 하며 유틸리티를 수행하는 사용자가 쓸 수 있어야 합니다.

프롬프트에 응답한 후, 별칭 파일이 작성되고 텍스트 편집기에서 열립니다.

## Documentum 등록 테이블에서 DM\_ID 오브젝트 유형 맵핑

유틸리티가 작성한 컬럼 정의는 해당 DB2 데이터 유형에 대한 각 데이터 유형의 올바른 맵핑을 포함하여 Documentum 랩퍼의 요구사항을 준수합니다. 유일한 예외는 Documentum이 등록된 테이블에서 DM\_ID 데이터 유형을 지원하지 않는다는 것입니다.

유틸리티는 등록된 테이블에 있는 컬럼이 문자열로 정의된 경우 오브젝트 ID를 포함하기 위해 사용되며, 16문자 길이며 "\_id"로 끝나는 이름을 갖는다고 가정합니다. DM\_ID 데이터 유형의 경우, 유틸리티는 컬럼을 DB2 CHAR(16) 데이터 유형으로 맵핑합니다. 다른 모든 경우에, 모든 문자열/varchar 컬럼은 DB2 VARCHAR 데이터 유형으로 맵핑됩니다. 유틸리티가 컬럼을 잘못 맵핑한 경우, 별칭을 DB2로 정의하기 위해 파일을 사용하기 전에 DB2 데이터 유형을 변경하십시오.

---

## 반복 속성 정의 이중화

랩퍼의 조회 성능을 최대화하려면, 각 속성이 참(true)과 대등한 DB2 데이터 유형으로 정의되어야 합니다. 즉, Documentum 정수는 DB2 정수 등으로 정의되어야 합니다. 그러나 이 정의는 비VARCHAR 반복 속성에 대한 다중 값을 리턴하지 못하게 합니다. 이러한 컬럼의 경우, index[0]의 값만이 리턴됩니다.

가능할 때마다 랩퍼는 Docbase 오브젝트당 오직 한 행의 결과만을 리턴하므로 이 제한사항이 존재합니다. 이 제한사항은 반복 속성이 선택될 때에만 나타납니다. 그러나 동일한 원격 반복 속성에 대한 두 번째 컬럼을 정의할 수 있지만 데이터 유형은 VARCHAR입니다.

이 컬럼 이름은 모든 관련 값의 분리문자로 분리된 목록으로 모든 값을 리턴하기 위해 SELECT 목록에서 사용됩니다(각 컬럼의 DELIMITER 옵션은 사용되는 분리문자를 지정함).

다중 값 컬럼의 지역 이름을 표준화해야 합니다. 참(true) 데이터 유형으로 정의된 컬럼의 지역 이름에 접두부 "m\_"을 추가하여 각 다중 값 컬럼의 지역 이름을 표준화할 수 있습니다.

예를 들어, 데이터 유형 TIMESTAMP로 정의된 approval\_dates라는 Documentum 반복 속성의 별칭 컬럼을 갖는다고 가정하십시오. m\_approval\_dates라는 두 번째 별칭을 작성하고 VARCHAR 데이터 유형으로 정의할 수 있습니다. 그런 다음 SELECT 목록에서 m\_approval\_dates를 사용하여 분리문자 분리 목록에서 모든 승인 날짜를 리턴할 수 있습니다.

참(true) 데이터 유형이 VARCHAR인 반복 속성에 대해서는 이중화 정의를 사용할 필요가 없습니다.

---

## 제한 및 고려사항

이 절에는 Documentum 랩퍼의 사용과 연관된 제한사항 및 고려사항의 목록이 들어 있습니다.

- 반복 속성 값 리턴에 관련된 제한사항: index[0]에 있는 값만이 비VARCHAR 반복 속성에 대해 리턴됩니다. 이 제한을 극복하기 위해, 반복 속성 컬럼에 대한 이중화 정의를 작성할 수 있습니다. 반복 속성에 대한 이중화 정의 작성에 대한 자세한 정보는 48 페이지의 『반복 속성 정의 이중화』를 참조하십시오.

또한, VARCHAR로 정의된 반복 속성의 여러 값이 하나의 분리문자로 분리된 문자열로 리턴됩니다. 분리문자는 32 페이지의 『랩퍼 컬럼 옵션』에 기술된 DELIMITER 별칭의 설정에 의해 좌우됩니다.

- Passthru는 지원되지 않습니다.
- DB2 응용프로그램에 의해 작성된 DB2 응용프로그램에 대한 각 연결의 경우, Documentum 랩퍼는 최대 10개의 동시 Documentum 세션을 지원하며, 그러한 각 세션은 최대 10개의 Documentum 조회를 동시에 관리할 수 있습니다. 단일 DB2 응용프로그램은 동시에 진행 중인 여러 조회를 가질 수 있으며, DB2

로 조회가 제출될 때 조회의 수명이 시작되며 결과 세트에서 해당 커서가 닫힐 때 종료합니다. 주어진 시간에, 해당 시간에 진행 중인 전체 조회 세트에서 다음 제한사항이 유지되어야 합니다.

- 모든 조회에 의해 참조되는 모든 별칭은 단지 10개의 다른 Documentum 서버에 상주해야 합니다.
- 하나의 Documentum 서버에서 단지 10개의 별칭이 참조될 수 있습니다.

둘 이상의 조회에서 언급되거나 단일 조회에서 여러번 참조되는 별칭은 나타날 때마다 한 번만 카운트됩니다.

- Documentum 랩퍼는 클라이언트 라이브러리의 AIX용 버전 3.1.7a를 사용합니다. Documentum 4i를 사용 중인 경우, Documentum에서 클라이언트 라이브러리의 이전 버전을 획득하는 것이 필요합니다(아직 설치되지 않은 경우).
- DB2가 Boolean 유형을 지원하지 않으므로, 이 함수가 정수를 리턴하도록 정의되었으므로 WHERE절에서 사용된 대부분의 사용자 정의 함수(USER, HITS & SCORE 제외)는 "=1" 검사를 수행해야 합니다.

예를 들면,

```
"... WHERE DCTM.ANY_EQ(authors,'Dave Winters')=1"
```

- DB2의 제한 때문에, 사용자 정의 함수는 인수 없이 정의되지 않습니다. 대신, 이 함수는 사용되지 않는 정수 인수로 정의됩니다. 함수는 USER, GET\_FILE, GET\_FILE\_DEL, HITS 및 SCORE입니다.
- DB2의 동일한 인스턴스에 대해 실행하는 모든 서버는 동일한 Documentum dmcl.ini 구성 매개변수를 공유해야 합니다.
- 반복 속성에 대한 ANY\_IN 사용자 정의 함수에서 단일 명령문에 대한 값의 최대 수는 10입니다. 그러나 여러 명령문은 OR될 수 있습니다.
- ANY\_SAME\_INDEX 사용자 정의 함수의 경우 반복 속성의 동일 색인에서 값에 대한 테스트의 최대 수는 10입니다. 테스트는 왼쪽에서 오른쪽으로 평가되는 AND 테스트이어야 합니다.
- 랩퍼는 특정 코드 페이지에 대하여 특정한 성능을 갖지 않습니다.

---

## 액세스 제어

조회는 Docbase에서 사용자 권한에 종속됩니다. 사용자가 적어도 읽기 액세스를 갖는 문서만이 조회 결과에 포함됩니다.

---

## 메시지

이 절에서는 Documentum의 랩퍼로 작업하는 중에 발생할 수 있는 메시지 목록을 나열하고 설명합니다. 메시지에 대한 자세한 정보는 *DB2 메시지 참조서*를 참조하십시오.

표 11. Documentum의 랩퍼가 발행한 메시지

오류 코드	메시지	설명
SQL0901N	심각하지 않은 시스템 오류로 인해 SQL문이 실패했습니다. 후속 SQL 문은 처리될 수 있습니다(이유: "dmAPI exec 실패: [DM_QUERY_E_BAD_QUAL] 오류: "속성 <column_name>에 대한 속성 규정자 A0가 유효한 규정자가 아닙니다").	올바르지 않은 Documentum 유형 또는 등록된 테이블이 REMOTE_OBJECT 별칭 옵션에 대해 입력되었습니다. 올바른 Documentum 오브젝트 유형 또는 등록된 테이블을 사용하려면 별칭을 변경하십시오.
SQL0901N	심각하지 않은 시스템 오류로 인해 SQL문이 실패했습니다. 후속 SQL 문은 처리될 수 있습니다(이유: "올바르지 않은 널(NULL) 컬럼이 지정되었습니다").	내부 프로그래밍 오류. IBM 고객만족센터에 문의하십시오.
SQL0901N	심각하지 않은 시스템 오류로 인해 SQL문이 실패했습니다. 후속 SQL 문은 처리될 수 있습니다(이유: "Nickname 스펙이 비어 있습니다").	내부 프로그래밍 오류. IBM 고객만족센터에 문의하십시오.
SQL0901N	심각하지 않은 시스템 오류로 인해 SQL문이 실패했습니다. 후속 SQL 문은 처리될 수 있습니다(이유: "출력 오브젝트가 비어 있거나 불완전합니다").	내부 프로그래밍 오류. IBM 고객만족센터에 문의하십시오.

표 11. Documentum의 랩퍼가 발행한 메시지 (계속)

오류 코드	메시지	설명
SQL0901N	심각하지 않은 시스템 오류로 인해 SQL문이 실패했습니다. 후속 SQL 문은 처리될 수 있습니다(이유: "예상치 못한 컬럼 수가 요청되었습니다").	내부 프로그래밍 오류. IBM 고객만족센터에 문의하십시오.
SQL0901N	심각하지 않은 시스템 오류로 인해 SQL문이 실패했습니다. 후속 SQL 문은 처리될 수 있습니다(이유: "컬럼 정보가 없습니다").	내부 프로그래밍 오류. IBM 고객만족센터에 문의하십시오.
SQL0901N	심각하지 않은 시스템 오류로 인해 SQL문이 실패했습니다. 후속 SQL 문은 처리될 수 있습니다(이유: "지원되지 않는 컬럼 유형이 요청되었습니다").	내부 프로그래밍 오류. IBM 고객만족센터에 문의하십시오.
SQL0901N	심각하지 않은 시스템 오류로 인해 SQL문이 실패했습니다. 후속 SQL 문은 처리될 수 있습니다(이유: "올바르지 않은 컬럼 정의").	내부 프로그래밍 오류. IBM 고객만족센터에 문의하십시오.
SQL0901N	심각하지 않은 시스템 오류로 인해 SQL문이 실패했습니다. 후속 SQL 문은 처리될 수 있습니다(이유: "일치하지 않는 유형, DB2 요청 != 별칭 유형").	내부 프로그래밍 오류. IBM 고객만족센터에 문의하십시오.
SQL0901N	심각하지 않은 시스템 오류로 인해 SQL문이 실패했습니다. 후속 SQL 문은 처리될 수 있습니다(이유: "출력 매개변수가 널(NULL)이 아닙니다").	내부 프로그래밍 오류. IBM 고객만족센터에 문의하십시오.
SQL0901N	심각하지 않은 시스템 오류로 인해 SQL문이 실패했습니다. 후속 SQL 문은 처리될 수 있습니다(이유: "조회 출력 변수가 널(NULL)이 아닙니다").	내부 프로그래밍 오류. IBM 고객만족센터에 문의하십시오.
SQL0901N	심각하지 않은 시스템 오류로 인해 SQL문이 실패했습니다. 후속 SQL 문은 처리될 수 있습니다(이유: "올바르지 않은 시간소인 길이").	내부 프로그래밍 오류. IBM 고객만족센터에 문의하십시오.



표 11. Documentum의 래퍼가 발행한 메시지 (계속)

오류 코드	메시지	설명
SQL0901N	심각하지 않은 시스템 오류로 인해 SQL문이 실패했습니다. 후속 SQL 문은 처리될 수 있습니다(이유: "일치하지 않는 컬럼 수").	내부 프로그래밍 오류. IBM 고객만족센터에 문의하십시오.
SQL0901N	심각하지 않은 시스템 오류로 인해 SQL문이 실패했습니다. 후속 SQL 문은 처리될 수 있습니다(이유: "반복 함수 build_predicate_string의 오류").	내부 프로그래밍 오류. IBM 고객만족센터에 문의하십시오.
SQL0901N	심각하지 않은 시스템 오류로 인해 SQL문이 실패했습니다. 후속 SQL 문은 처리될 수 있습니다(이유: "함수 build_function_string의 오류").	내부 프로그래밍 오류. IBM 고객만족센터에 문의하십시오.
SQL0901N	심각하지 않은 시스템 오류로 인해 SQL문이 실패했습니다. 후속 SQL 문은 처리될 수 있습니다(이유: "값을 변환할 때 데이터에 액세스할 수 없음").	내부 프로그래밍 오류. IBM 고객만족센터에 문의하십시오.
SQL0901N	심각하지 않은 시스템 오류로 인해 SQL문이 실패했습니다. 후속 SQL 문은 처리될 수 있습니다(이유: "올바르지 않은 RDBMS_TYPE이 DB2, INFORMIX, ORACLE, SQLSERVER 또는 SYBASE이어야 합니다").	RDBMS_TYPE 서버 옵션에 대해 올바른 값이 지정되었습니다. 올바른 값을 지정하십시오.
SQL0901N	심각하지 않은 시스템 오류로 인해 SQL문이 실패했습니다. 후속 SQL 문은 처리될 수 있습니다(이유: "올바르지 않은 TRANSACTIONS이 NONE, QUERY, PASSTHRU 또는 ALL이어야 합니다").	TRANSACTIONS 서버 옵션에 대해 올바른 값이 지정되었습니다. 올바른 값을 지정하십시오.
SQL0901N	심각하지 않은 시스템 오류로 인해 SQL문이 실패했습니다. 후속 SQL 문은 처리될 수 있습니다(이유: "DMCL 클라이언트를 초기화하는데 실패했습니다").	Documentum 클라이언트가 초기화할 수 없습니다. 시스템 관리자에게 문의하십시오.

표 11. Documentum의 랩퍼가 발행한 메시지 (계속)

오류 코드	메시지	설명
SQL0901N	심각하지 않은 시스템 오류로 인해 SQL문이 실패했습니다. 후속 SQL 문은 처리될 수 있습니다(이유: "Get_User가 널(NULL)을 리턴했습니다").	내부 프로그래밍 오류. IBM 고객만족센터에 문의하십시오.
SQL0901N	심각하지 않은 시스템 오류로 인해 SQL문이 실패했습니다. 후속 SQL 문은 처리될 수 있습니다(이유: "Get_Local_User가 널(NULL)을 리턴했습니다").	내부 프로그래밍 오류. IBM 고객만족센터에 문의하십시오.
SQL0901N	심각하지 않은 시스템 오류로 인해 SQL문이 실패했습니다. 후속 SQL 문은 처리될 수 있습니다(이유: "트랜잭션 시작에 실패했습니다").	Documentum이 begintrans에 실패했음을 보고했습니다. 시스템 관리자에게 문의하십시오.
SQL0901N	심각하지 않은 시스템 오류로 인해 SQL문이 실패했습니다. 후속 SQL 문은 처리될 수 있습니다(이유: "입력 매개변수가 널(NULL)이 아닙니다").	내부 프로그래밍 오류. IBM 고객만족센터에 문의하십시오.
SQL0901N	심각하지 않은 시스템 오류로 인해 SQL문이 실패했습니다. 후속 SQL 문은 처리될 수 있습니다(이유: "올바르지 않은 컬럼 번호가 요청되었습니다").	내부 프로그래밍 오류. IBM 고객만족센터에 문의하십시오.
SQL0901N	심각하지 않은 시스템 오류로 인해 SQL문이 실패했습니다. 후속 SQL 문은 처리될 수 있습니다(이유: "IS_REG_TABLE에 대한 올바른 값이 없습니다. 'Y' 또는 'N'이어야 합니다").	IS_REG_TABLE 별칭에 대한 올바른 값이 CREATE NICKNAME문에 지정되었습니다. 올바른 값을 지정하십시오.
SQL0901N	심각하지 않은 시스템 오류로 인해 SQL문이 실패했습니다. 후속 SQL 문은 처리될 수 있습니다(이유: "ALL_VERSIONS에 대한 올바른 값이 없습니다. 'Y' 또는 'N'이어야 합니다").	ALL_VERSIONS 별칭 옵션에 대한 올바른 값이 CREATE NICKNAME문에 지정되었습니다. 올바른 값을 지정하십시오.

표 11. Documentum의 랩퍼가 발행한 메시지 (계속)

오류 코드	메시지	설명
SQL30090N	응용프로그램 실행 환경에 대해 유효하지 않은 조작입니다. 이유 코드 = "올바르지 않은 컬럼 이름, IS_REG_TABLE 또는 IS_REPEATING이 별칭에 지정되었습니다"	IS_REG_TABLE, IS_REPEATING, REMOTE_NAME 옵션 및 컬럼 이름의 올바른 스펙에 대해서는 별칭 명령문을 검사하십시오.
SQL30090N	응용프로그램 실행 환경에 대해 유효하지 않은 조작입니다. 이유 코드 = "VERSIONS가 등록된 테이블에서 지원되지 않습니다"	VERSIONS 옵션이 등록된 테이블 별칭에 대해 지정되었습니다. 별칭 정의에서 VERSIONS 옵션을 제거하십시오.
SQL30090N	응용프로그램 실행 환경에 대해 유효하지 않은 조작입니다. 이유 코드 = "올바르지 않은 함수가 선택 목록에 포함되었습니다"	GET_FILE, GET_FILE_DELETE, GET_RENDITION, GET_RENDITION_DELETE, HITS 및 SCORE만이 SELECT 목록에서 허용됩니다. SELECT 목록에서 올바른 함수를 제거하십시오.
SQL30090N	응용프로그램 실행 환경에 대해 유효하지 않은 조작입니다. 이유 코드 = "Dctm 함수는 DCTM.function(..)=1과 유사해야 합니다"	사용자가 Dctm 함수에 대한 술어의 RHS로서 =1을 사용하지 않았습니다. 구문을 수정하고 조회를 다시 수행하십시오.
SQL30090N	응용프로그램 실행 환경에 대해 유효하지 않은 조작입니다. 이유 코드 = "SELECT절에 올바른 상수가 있습니다"	테이블에서 상수를 선택하는 것은 허용되지 않습니다. SELECT 목록에서 상수를 제거하고 다시 시도하십시오.
SQL30090N	응용프로그램 실행 환경에 대해 유효하지 않은 조작입니다. 이유 코드 = "db2dj.ini에서 DOCUMENTUM 또는 DMCL_CONFIG env var가 누락되었습니다"	필수 환경 변수가 설정되지 않았습니다. db2dj.ini 파일에서 설정하십시오.
SQL30090N	응용프로그램 실행 환경에 대해 유효하지 않은 조작입니다. 이유 코드 = "디버깅을 위해 로그 파일을 여는 데 실패했습니다"	문제점 해결에 사용된 로그 파일에 액세스할 수 없습니다. 시스템 관리자에게 문의하십시오.
SQL30090N	응용프로그램 실행 환경에 대해 유효하지 않은 조작입니다. 이유 코드 = "올바르지 않은 디버그 레벨이 있습니다"	올바르지 않은 디버그 레벨이 서버 옵션으로 지정되었습니다. 올바른 디버그 레벨을 지정하십시오.

표 11. Documentum의 랩퍼가 발행한 메시지 (계속)

오류 코드	메시지	설명
SQL30090N	응용프로그램 실행 환경에 대해 유효하지 않은 조작입니다. 이유 코드 = "올바르지 않은 OS_TYPE, AIX, HPUX, SOLARIS 또는 WINDOWS여야 합니다"	OS_TYPE 서버 옵션에 대해 올바른 값을 지정하십시오. 이 지정되었습니다. 올바른 값을 지정하십시오.
SQL30090N	응용프로그램 실행 환경에 대해 유효하지 않은 조작입니다. 이유 코드 = "FOLDERS절이 등록된 테이블에 대해 유효하지 않습니다"	IS_REG_TABLE 옵션이 "Y"로 설정되고 FOLDERS 옵션이 설정되었습니다. FOLDERS 가 등록된 테이블에 사용되지 않습니다. CREATE NICKNAME문 옵션을 갱신하십시오.
SQL30090N	응용프로그램 실행 환경에 대해 유효하지 않은 조작입니다. 이유 코드 = "오직 하나의 검색 조건만이 지정될 수 있습니다"	오직 하나의 사용자 정의 검색 함수가 조회마다 지정될 수 있습니다.
SQL30090N	응용프로그램 실행 환경에 대해 유효하지 않은 조작입니다. 이유 코드 = "내용 디렉토리를 작성하는 데 실패했습니다"	DB2 에이전트에 의해 대상 디렉토리를 쓸 수 있는지 확인하십시오.
SQL30090N	응용프로그램 실행 환경에 대해 유효하지 않은 조작입니다. 이유 코드 = "내용 파일에서 권한을 변경하는 데 실패했습니다"	db2 에이전트에 의해 목표 내용 디렉토리를 쓸 수 있는지 확인하십시오.
SQL30090N	응용프로그램 실행 환경에 대해 유효하지 않은 조작입니다. 이유 코드 = "IS_REPEATING = 'Y'인 경우를 제외하고는 DELIMITER가 유효하지 않습니다"	IS_REPEATING 및 DELIMITER의 별칭 값을 검사하십시오.

---

## 제5장 데이터 소스로서 Excel 사용

이 장에서는 다음에 대해 설명합니다.

- Excel 스프레드시트 데이터 소스
- 연합 시스템에 Excel 데이터 소스를 추가하는 방법
- Excel 데이터 소스에 대한 조회를 수행하는 방법
- 샘플 사용자 시나리오
- 제한 및 고려사항
- 사용된 파일 액세스 제어 모델
- Excel 랩퍼에서 작업시 발생할 수 있는 메시지

---

### Excel의 개념

Excel 스프레드시트 또는 워크북은 Microsoft(MS) Excel 응용프로그램을 사용하여 작성된 파일이며 xls 파일 확장자를 갖습니다. DB2 Life Sciences Data Connect는 Excel 97, Excel 98 및 Excel 2000에서 스프레드시트를 지원합니다. 58 페이지의 그림4에서는 Excel 랩퍼가 스프레드시트를 연합 시스템에 연결하는 방법을 설명합니다.

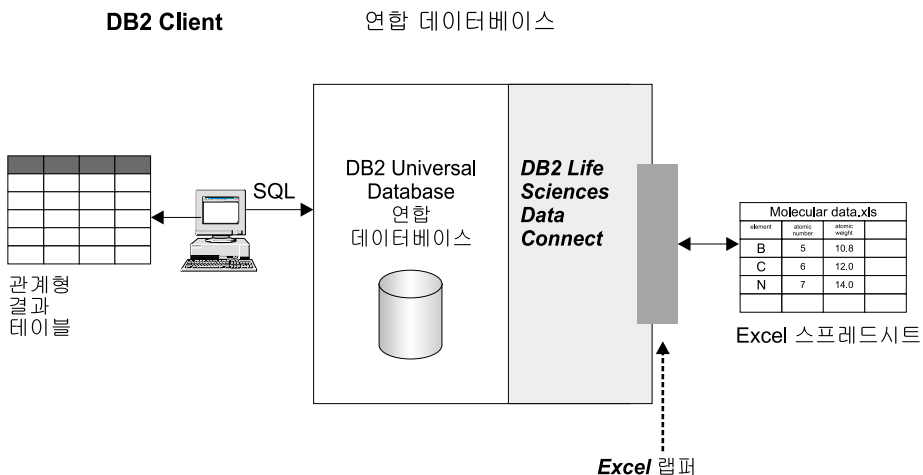


그림 4. Excel 래퍼 작업 방법

Excel 래퍼는 CREATE NICKNAME문을 사용하여 Excel 스프레드시트의 컬럼을 DB2 연합 시스템의 컬럼으로 맵핑합니다. 표12에서는 Compound\_Master.xls 라는 파일에 저장되는 샘플 스프레드시트 데이터를 표시합니다.

표 12. Compound\_Master.xls의 샘플 스프레드시트

	A	B	C	D
1	compound_A	1.23	367	tested
2	compound_G		210	
3	compound_F	0.000425536	174	tested
4	compound_Y	1.00256		tested
5	compound_Q		1024	
6	compound_B	33.5362		
7	compound_S	0.96723	67	tested
8				
9	compound_O	1.2		tested

이 정보는 보통 표준 SQL 명령어를 통해서만 사용 불가능합니다. Excel 래퍼가 설치 및 등록되면, 표준 관계형 데이터 소스에 있는 것처럼 이 정보에 액세스할 수 있습니다. 예를 들어, 분자 수가 100보다 큰 모든 화합물 데이터를 알려는 경우, 다음 SQL 조회를 실행합니다.

```
SELECT * FROM compound_master WHERE mol_count > 100
```

조회 결과는 표13에 표시됩니다.

표 13. 조회 결과

COMPOUND_NAME	WEIGHT	MOL_COUNT	WAS_TESTED
compound_A	1.23	367	tested
compound_G		210	
compound_F	0.000425536	174	tested
compound_Q		1024	

## 전제조건

Excel 데이터 소스 래퍼 이용에 대한 전제조건은 다음과 같습니다.

- Excel 응용프로그램은 Excel 래퍼를 이용하기 전에 DB2 Life Sciences Data Connect가 설치되는 서버에 설치되어야 합니다.
- Excel 97 또는 98이 설치되면, Excel\_9x 래퍼가 등록되어야 합니다. Excel 2000이 설치되면, Excel\_2000 래퍼가 등록되어야 합니다.

## 연합 시스템에 Excel 추가

연합 시스템에 Excel 데이터 소스를 추가하려면 다음을 수행하십시오.

1. CREATE WRAPPER문을 사용하여 래퍼를 등록하십시오.
2. CREATE SERVER문을 사용하여 서버를 등록하십시오.
3. 액세스하려는 각 Excel 스프레드시트에 대해 CREATE NICKNAME문을 사용하여 별칭을 등록하십시오.

이 단계는 이 절에서 자세히 설명됩니다. 명령은 DB2 명령행 처리기에서 수행될 수 있습니다.

### 1단계: 래퍼 등록

Excel 데이터 소스 래퍼를 등록하려면, CREATE WRAPPER문을 제출하십시오.

라이브러리 파일 liblsexcel97.dll을 사용하여 Excel\_9x\_Wrapper라는 Excel 97 또는 98에 대한 Excel 래퍼를 작성하려면, 다음 명령문을 제출하십시오.

```
CREATE WRAPPER Excel_9x_Wrapper LIBRARY 'liblsexcel97.dll';
```

라이브러리 파일 liblsexcel2k.dll을 사용하여 Excel\_2000\_Wrapper라는 Excel 2000에 대한 Excel 래퍼를 작성하려면, 다음 명령문을 제출하십시오.

```
CREATE WRAPPER Excel_2000_Wrapper LIBRARY 'liblsexcel2k.dll';
```

CREATE WRAPPER문에 대한 자세한 정보는 *DB2 SQL 참조서*를 참조하십시오.

## 2단계: 서버 등록

CREATE SERVER문을 사용하여 연합 시스템에 Excel 서버를 등록하십시오.

예를 들어, 59 페이지의 『1단계: 래퍼 등록』에서 작성된 Excel\_2000\_Wrapper 래퍼에 대한 서버를 등록하는 biochem\_node1의 노드 이름으로, biochem\_lab이라는 서버를 작성하려면 다음 명령문을 제출하십시오.

```
CREATE SERVER biochem_lab  
TYPE Excel_2000  
VERSION '2000'  
WRAPPER Excel_2000_Wrapper  
OPTIONS(  
NODE 'biochem_node1');
```

### 인수 정의

**TYPE** Excel\_9x 또는 Excel\_2000 중 하나로 서버 유형을 지정합니다. 이 인수는 필수입니다.

### VERSION

‘9x’ 또는 ‘2000’ 중 하나로 서버에 설치된 Excel 버전을 지정합니다. 이 인수는 필수입니다.

### WRAPPER

59 페이지의 『1단계: 래퍼 등록』에 등록한 래퍼의 이름을 지정합니다. 이 인수는 필수입니다.



## 옵션 정의

### NODE

선택한 노드 이름을 지정합니다. 이는 지역 노드에 제공하는 이름입니다. 원하는 텍스트 문자열을 포함할 수 있습니다. 이 옵션은 필수입니다.

CREATE SERVER문에 대한 자세한 정보는 *DB2 SQL* 참조서를 참조하십시오.

## 3단계: 별칭 등록

Excel 데이터 소스를 관계형 테이블로 매핑하려면, CREATE NICKNAME문을 사용하여 별칭을 작성하십시오.

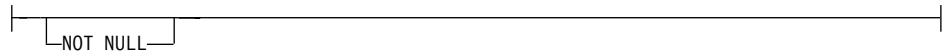
### CREATE NICKNAME 구문(Excel용)

```
► CREATE NICKNAME nickname ( column-name | data-type | column-option ) FOR SERVER server-name  
► OPTIONS ( FILE_PATH 'path' )
```

#### data-type:



#### column-option:



CREATE NICKNAME문에 대한 자세한 정보는 *DB2 SQL* 참조서를 참조하십시오.

### FOR SERVER

60 페이지의 『2단계: 서버 등록』에서 등록된 서버를 식별합니다. 이 서버

는 Excel 스프레드시트를 액세스하기 위해 사용됩니다. 서버 이름, 기간, 기본 파일 이름 순으로 지정하십시오. 예를 들어, biochem\_lab 서버가 CompoundMaster.xls 파일을 액세스하는 경우, biochem\_lab.CompoundMaster를 지정합니다.

## 옵션 정의

### FILE\_PATH

액세스하려는 Excel 스프레드시트의 완전한 디렉토리 경로 및 파일 이름을 지정합니다.

다음 예에서의 명령문은 CompoundMaster.xls라는 Excel 스프레드시트에서 화합물 별칭을 작성합니다. 파일에는 Compound\_ID, CompoundName 및 MolWeight로 연합 시스템에 정의 중인 세 개의 데이터 컬럼이 들어 있습니다.

```
CREATE NICKNAME Compounds (  
Compound_ID INTEGER,  
CompoundName VARCHAR(50),  
MolWeight FLOAT)  
FOR SERVER biochem_lab.CompoundMaster  
OPTIONS(PATH 'C:\My Documents\CompoundMaster.xls');
```

---

## 조회 실행

이 절은 61 페이지의 『3단계: 별칭 등록』에서 별칭 화합물의 예를 사용하여 여러 샘플 Excel 스프레드시트를 나열합니다.

다음 조회는 분자량이 200보다 큰 모든 compound\_ID를 표시합니다.

```
SELECT compound_ID  
FROM Compounds  
WHERE MolWeight > 200;
```

다음 조회는 화합물 이름 또는 분자량이 널(NULL)인 모든 레코드를 표시합니다.

```
SELECT *  
FROM Compounds  
WHERE CompoundName IS NULL  
OR MolWeight IS NULL;
```

다음 조회는 화합물 이름에 문자열 'ase'가 들어 있으며 분자량이 300 이상인 모든 레코드를 표시합니다.

```
SELECT *
FROM Compounds
WHERE CompoundName LIKE '%ase%'
AND MolWeight >=300;
```

## 샘플 시나리오

이 절에서는 C:\Data 디렉토리에 위치한 Excel 2000 스프레드시트에 액세스하는 Excel\_2000 랩퍼의 샘플 구현을 설명합니다. 시나리오는 랩퍼를 등록하며, 서버를 등록하며, 스프레드시트를 액세스하는 데 사용할 하나의 별칭을 등록합니다. 시나리오에 표시된 명령문은 DB2 명령행 처리기를 사용하여 입력됩니다. 랩퍼가 등록된 후에, 스프레드시트에 대한 조회를 실행할 수 있습니다.

시나리오는 4 컬럼, 9행인 Compound\_Master.xls라는 화합물 스프레드시트로 시작합니다. 파일에 대한 완전한 경로 이름은 C:\Data\Compound\_Master.xls입니다. 내용은 표14에 표시됩니다.

표 14. Compound\_Master.xls 샘플 스프레드시트

	A	B	C	D
1	compound_A	1.23	367	tested
2	compound_G		210	
3	compound_F	0.000425536	174	tested
4	compound_Y	1.00256		tested
5	compound_Q		1024	
6	compound_B	33.5362		
7	compound_S	0.96723	67	tested
8				
9	compound_O	1.2		tested

1. Excel\_2000 랩퍼를 등록하십시오.

```
db2 => CREATE WRAPPER Excel_2000 LIBRARY 'liblsexcel2k.dll'
```

2. 서버를 등록하십시오.

```
db2 => CREATE SERVER biochem_lab TYPE Excel2000 VERSION '2000'
WRAPPER Excel_2000 OPTIONS(NODE 'biochem_node1')
```

3. Excel 스프레드시트를 나타내는 별칭을 등록하십시오.

```
db2 => CREATE NICKNAME Compound_Master (compound_name VARCHAR(40), weight FLOAT,
mol_count INTEGER, was_tested VARCHAR(20)) FOR bTochem_lab.compound_master
OPTIONS ( PATH 'C:\Data\Compound_Master.xls')
```

등록 프로세스가 완료되었습니다. Excel 데이터 소스는 이제 연합 시스템의 일부이며, SQL 조회에서 사용될 수 있습니다.

다음 예는 Excel 데이터 소스를 사용하여 획득한 샘플 SQL 조회 및 결과를 표시합니다.

- 샘플 SQL 조회: "mol\_count가 100보다 큰 모든 화합물 데이터를 제공하십시오"

```
SELECT * FROM compound_master WHERE mol_count > 100
```

결과: 행 1, 2, 3, 5 및 7에 대한 모든 필드.

- 샘플 SQL 조회: "mol\_count가 아직 결정되지 않은 모든 화합물에 대한 compound\_name 및 mol\_count를 제공하십시오."

```
SELECT compound_name, mol_count FROM compound_master
WHERE mol_count IS NULL
```

결과: 스프레드시트로부터 행 4, 6, 8 및 9의 필드 compound\_name 및 mol\_count.

- 샘플 SQL 조회: "테스트되지 않았으며 무게가 1보다 큰 화합물의 수를 카운트 하십시오."

```
SELECT count(*) FROM compound_master
WHERE was_tested IS NULL AND weight > 1
```

결과: 기준과 일치하는 스프레드시트에서 단일 행 6을 나타내는 레코드 카운트 1.

- 샘플 SQL 조회: "mol\_count가 결정되고 average mol\_count 미만인 모든 화합물에 대한 compound\_name 및 mol\_count를 제공하십시오."

```
SELECT compound_name, mol_count
FROM compound_master
WHERE mol_count IS NOT NULL
AND mol_count < (SELECT AVG(mol_count) FROM compound_master
WHERE mol_count IS NOT NULL AND was_tested IS NOT NULL)
```

부속 조회는 표15를 리턴하는 주 조회로 평균 368을 리턴합니다.

표 15. 조회 결과

COMPOUND_NAME	MOL_COUNT
compound_A	367
compound_G	210
compound_F	174
compound_S	67

## 제한 및 고려사항

이 절에는 Excel 랩퍼의 사용과 연관된 제한사항 및 고려사항의 목록이 들어 있습니다.

### 랩퍼 고려사항

- MS Excel 97 또는 98 응용프로그램 사용 시 Excel\_9x 랩퍼를 사용하십시오.
- MS Excel 2000 응용프로그램 사용 시 Excel\_2000 랩퍼를 사용하십시오.

### 랩퍼 제한사항

- Excel 랩퍼는 DB2 Universal Database Enterprise Edition 및 DB2 Universal Database Enterprise-Extended Edition을 지원하는 Microsoft Windows 운영 체제에만 사용 가능합니다.
- Passthru 세션은 Excel 랩퍼에서 허용되지 않습니다.
- Excel 스프레드시트 데이터는 읽기만 가능하며 쓸 수 없습니다.
- Excel 랩퍼는 ALTER NICKNAME문을 사용하여 컬럼 이름을 변경하도록 허용하지 않습니다.

### Excel 파일 제한사항

- 데이터 유형은 각 컬럼 내에서 일치되어야 하며 컬럼 데이터 유형은 별칭 등록 프로세스 중 제대로 기술되어야 합니다.
- Excel 랩퍼는 Excel 워크북 내에서 기본 스프레드시트에만 액세스할 수 있습니다.

- Excel\_2000 랩퍼는 Excel 97, 98 및 2000 스프레드시트에 액세스할 수 있습니다.
- Excel\_9x 랩퍼는 Excel 97 및 98 스프레드시트에 액세스할 수 있습니다.
- 스프레드시트에서 공백 셀은 널(NULL)로서 해석됩니다.
- 최대 10개의 연속 공백 행이 스프레드시트에 존재할 수 있으며 데이터 세트에 포함될 수 있습니다. 11개 이상의 연속 공백 행은 데이터 세트의 끝으로 해석됩니다.
- 공백 컬럼이 스프레드시트에 존재할 수 있습니다. 그러나 이 컬럼이 사용되지 않는 경우일지라도 유효한 필드로서 등록 및 설명되어야 합니다.

## 파일 액세스 제어 모델

데이터베이스 관리 시스템은 DB2 데이터베이스 서비스의 LOG ON AS 등록 정보 권한으로 Excel 파일에 액세스합니다. 이 설정은 DB2 인스턴스에 대한 LOG ON 등록 정보에서 볼 수 있습니다. 등록 정보 페이지는 Windows NT 서비스 제어판을 통해 액세스됩니다.

## 메시지

이 절에서는 Excel의 랩퍼로 작업하는 중에 발생할 수 있는 메시지 목록을 나열하고 설명합니다. 메시지에 대한 자세한 정보는 *DB2 메시지 참조서*를 참조하십시오.

표 16. Excel의 랩퍼가 발행한 메시지

오류 코드	메시지	설명
SQL1817N	CREATE SERVER문은 연함 데이터베이스에 정의되게 하려는 데이터 소스의 문 실행 중 지정되지 않았습니다. SQL문을 "VERSION"을 식별하지 않습니다.	VERSION 매개변수는 CREATE SERVER 문 실행 중 지정되지 않았습니다. SQL문을 수정하고 다시 수행하십시오.
SQL1822N	데이터 소스 "Excel 랩퍼"에서 예기치 않은 오류 코드 "-1000.<internal program code>"를 받았습니다. 연관된 텍스트 및 토큰은 "메모리 할당 오류"입니다.	IBM 고객만족센터에 문의하십시오.

표 16. Excel의 랩퍼가 발행한 메시지 (계속)

오류 코드	메시지	설명
SQL1822N	데이터 소스 "Excel 랩퍼"에서 예기치 않은 오류 코드 "-1001.<internal program code>"를 받았습니다. 연관된 텍스트 및 토큰은 "알 수 없는 옵션"입니다.	DDL문에 지정된 옵션은 지원되지 않습니다. SQL문을 수정하고 다시 수행하십시오.
SQL1822N	데이터 소스 "Excel 랩퍼"에서 예기치 않은 오류 코드 "-1002.<internal program code>"를 받았습니다. 연관된 텍스트 및 토큰은 "DELTA 오브젝트 작성에 실패함"입니다.	내부 프로그램 오류가 발생했습니다. IBM 고객만족센터에 문의하십시오.
SQL1822N	데이터 소스 "Excel 랩퍼"에서 예기치 않은 오류 코드 "-1100.<internal program code>"를 받았습니다. 연관된 텍스트 및 토큰은 "랩퍼 옵션이 지원되지 않음"입니다.	랩퍼 OPTIONS는 이 랩퍼에서 지원되지 않습니다. SQL문을 수정하고 다시 수행하십시오.
SQL1822N	데이터 소스 "Excel 랩퍼"에서 예기치 않은 오류 코드 "-1200.<internal program code>"를 받았습니다. 연관된 텍스트 및 토큰은 "<option>이 지원되지 않는 서버 옵션입니다"입니다.	지정된 옵션은 이 랩퍼에서 지원되지 않습니다. SQL문을 수정하고 다시 수행하십시오.
SQL1822N	데이터 소스 "Excel 랩퍼"에서 예기치 않은 오류 코드 "-1201.<internal program code>"를 받았습니다. 연관된 텍스트 및 토큰은 "서버 이름 확보 중 오류"입니다.	내부 프로그램 오류가 발생했습니다. IBM 고객만족센터에 문의하십시오.
SQL1822N	데이터 소스 "Excel 랩퍼"에서 예기치 않은 오류 코드 "-1209.<internal program code>"를 받았습니다. 연관된 텍스트 및 토큰은 "VARCHAR 데이터 변환 중 오류"입니다.	내부 프로그램 오류가 발생했습니다. IBM 고객만족센터에 문의하십시오.
SQL1822N	데이터 소스 "Excel 랩퍼"에서 예기치 않은 오류 코드 "-1211.<internal program code>"를 받았습니다. 연관된 텍스트 및 토큰은 "INTEGER 데이터 변환 중 오류"입니다.	내부 프로그램 오류가 발생했습니다. IBM 고객만족센터에 문의하십시오.

표 16. Excel의 랩퍼가 발행한 메시지 (계속)

오류 코드	메시지	설명
SQL1822N	데이터 소스 "Excel 랩퍼"에서 예기치 않은 오류 코드 "-1212.<internal program code>"를 받았습니다. 연관된 텍스트 및 토큰은 "FLOAT 데이터 변환 중 오류"입니다.	내부 프로그램 오류가 발생했습니다. IBM 고객만족센터에 문의하십시오.
SQL1822N	데이터 소스 "Excel 랩퍼"에서 예기치 않은 오류 코드 "-1400.<internal program code>"를 받았습니다. 연관된 텍스트 및 토큰은 "<option>이 지원되지 않는 사용자 옵션입니다"입니다.	지정한 옵션은 이 랩퍼에서 지원되지 않습니다. SQL문을 수정하고 다시 수행하십시오.
SQL1822N	데이터 소스 "Excel 랩퍼"에서 예기치 않은 오류 코드 "-1401.<internal program code>"를 받았습니다. 연관된 텍스트 및 토큰은 "USER 델타 오브젝트 작성에 실패함"입니다.	내부 프로그램 오류가 발생했습니다. IBM 고객만족센터에 문의하십시오.
SQL1822N	데이터 소스 "Excel 랩퍼"에서 예기치 않은 오류 코드 "-1500.<internal program code>"를 받았습니다. 연관된 텍스트 및 토큰은 "<option>이 지원되지 않는 별칭 옵션입니다"입니다.	지정한 옵션은 이 랩퍼에서 지원되지 않습니다. SQL문을 수정하고 다시 수행하십시오.
SQL1822N	데이터 소스 "Excel 랩퍼"에서 예기치 않은 오류 코드 "-1501.<internal program code>"를 받았습니다. 연관된 텍스트 및 토큰은 "필수 옵션 PATH가 지정되지 않았습니다"입니다.	PATH 옵션이 NICKNAME을 등록하는 데 필요합니다. SQL문을 수정하고 다시 수행하십시오.
SQL1822N	데이터 소스 "Excel 랩퍼"에서 예기치 않은 오류 코드 "-1502.<internal program code>"를 받았습니다. 연관된 텍스트 및 토큰은 "NICKNAME 델타 오브젝트 작성에 실패함"입니다.	내부 프로그램 오류가 발생했습니다. IBM 고객만족센터에 문의하십시오.
SQL1822N	데이터 소스 "Excel 랩퍼"에서 예기치 않은 오류 코드 "-1503.<internal program code>"를 받았습니다. 연관된 텍스트 및 토큰은 "별칭 컬럼 유형 확보 중 오류"입니다.	내부 프로그램 오류가 발생했습니다. IBM 고객만족센터에 문의하십시오.



표 16. Excel의 랩퍼가 발행한 메시지 (계속)

오류 코드	메시지	설명
SQL1822N	데이터 소스 "Excel 랩퍼"에서 예기치 않은 오류 코드 "-1504.<internal program code>"를 받았습니다. 연관된 텍스트 및 토큰은 "별칭 컬럼 유형 이름 확보 중 오류"입니다.	내부 프로그램 오류가 발생했습니다. IBM 고객만족센터에 문의하십시오.
SQL1822N	데이터 소스 "Excel 랩퍼"에서 예기치 않은 오류 코드 "-1505.<internal program code>"를 받았습니다. 연관된 텍스트 및 토큰은 "데이터 소스 "Excel 랩퍼"에서 수신되었습니다"입니다.	지정된 <data type>은 이 랩퍼에서 지원되지 않습니다. SQL문을 수정하고 다시 수행하십시오.
SQL1822N	데이터 소스 "Excel 랩퍼"에서 예기치 않은 오류 코드 "-1506.<internal program code>"를 받았습니다. 연관된 텍스트 및 토큰은 "별칭 컬럼 정보 확보 중 오류"입니다.	내부 프로그램 오류가 발생했습니다. IBM 고객만족센터에 문의하십시오.
SQL1822N	데이터 소스 "Excel 랩퍼"에서 예기치 않은 오류 코드 "-1507.<internal program code>"를 받았습니다. 연관된 텍스트 및 토큰은 "<option> 옵션을 제거할 수 없습니다"입니다.	지정된 옵션이 필수 옵션이므로 제거할 수 없습니다.
SQL1822N	데이터 소스 "Excel 랩퍼"에서 예기치 않은 오류 코드 "-1508.VANI"를 받았습니다. 연관된 텍스트 및 토큰은 "컬럼 이름을 변경할 수 없습니다"입니다.	컬럼 이름의 변경은 Excel 랩퍼에 의해 허용되지 않습니다.
SQL1822N	데이터 소스 "Excel 랩퍼"에서 예기치 않은 오류 코드 "-1701.<internal program code>"를 받았습니다. 연관된 텍스트 및 토큰은 "SQL 분석 중 오류"입니다.	내부 프로그램 오류가 발생했습니다. IBM 고객만족센터에 문의하십시오.
SQL1822N	데이터 소스 "Excel 랩퍼"에서 예기치 않은 오류 코드 "-1702.<internal program code>"를 받았습니다. 연관된 텍스트 및 토큰은 "NICKNAME 오브젝트 액세스 중 오류"입니다.	내부 프로그램 오류가 발생했습니다. IBM 고객만족센터에 문의하십시오.

표 16. Excel의 랩퍼가 발행한 메시지 (계속)

오류 코드	메시지	설명
SQL1822N	데이터 소스 "Excel 랩퍼"에서 예기치 않은 오류 코드 "-1703.<internal program code>"를 받았습니다. 연관된 텍스트 및 토큰은 "데이터 저장 영역 빌드 중 오류"입니다.	내부 프로그램 오류가 발생했습니다. IBM 고객만족센터에 문의하십시오.
SQL1822N	데이터 소스 "Excel 랩퍼"에서 예기치 않은 오류 코드 "-1704.<internal program code>"를 받았습니다. 연관된 텍스트 및 토큰은 "SQL을 별칭 데이터에 링크 중 오류"입니다.	내부 프로그램 오류가 발생했습니다. IBM 고객만족센터에 문의하십시오.
SQL1822N	데이터 소스 "Excel 랩퍼"에서 예기치 않은 오류 코드 "-1705.<internal program code>"를 받았습니다. 연관된 텍스트 및 토큰은 "Excel 응용프로그램 시작에 실패했습니다"입니다.	Excel 응용프로그램의 시작에 실패했습니다. Excel이 시스템에 설치되어 랩퍼의 올바른 버전으로 등록되었는지 확인하십시오. Windows NT 서비스 제어판에서 DB2 인스턴스에 대한 LOG ON AS 등록 정보를 확인하십시오. Excel 응용프로그램이 이 권한을 사용하여 액세스됩니다. 이 사용자가 해당 권한을 가지고 있는지 확인하거나 이 등록 정보를 권한 부여된 계정으로 변경한 다음, DB2를 재시작하고 SQL 조화를 다시 실행하십시오.
SQL1822N	데이터 소스 "Excel 랩퍼"에서 예기치 않은 오류 코드 "-1706.<internal program code>"를 받았습니다. 연관된 텍스트 및 토큰은 "소스 스프레드시트를 여는 중 오류"입니다.	SQL 조회에서 별칭으로 참조된 스프레드시트를 여는 중 문제점이 발생했습니다. 등록 중 CREATE NICKNAME문 실행 중 지정된 PATH에 파일이 존재하는지 확인하십시오.
SQL1822N	데이터 소스 "Excel 랩퍼"에서 예기치 않은 오류 코드 "-1707.<internal program code>"를 받았습니다. 연관된 텍스트 및 토큰은 "DL 출력 저장 영역 액세스 중 오류"입니다.	내부 프로그램 오류가 발생했습니다. IBM 고객만족센터에 문의하십시오.
SQL1822N	데이터 소스 "Excel 랩퍼"에서 예기치 않은 오류 코드 "-1708.<internal program code>"를 받았습니다. 연관된 텍스트 및 토큰은 "Excel 응용프로그램 종료에 실패했습니다"입니다.	내부 프로그램 오류가 발생했습니다. 반복된 조회 이후에도 이 오류가 지속되는 경우, IBM 고객만족센터에 문의하십시오.

표 16. Excel의 랩퍼가 발행한 메시지 (계속)

오류 코드	메시지	설명
SQL1822N	데이터 소스 "Excel 랩퍼"에서 예기치 않은 오류 코드 "-1711.<internal program code>"를 받았습니다. 연관된 텍스트 및 토큰은 "페치 중 오류, 데이터/컬럼 유형이 불일치할 수 있음"입니다.	별칭 등록 중 지정된 데이터 유형과는 다른 데이터 유형의 SQL 조회 중 데이터가 페치되었습니다. 소스 스프레드시트에서 데이터를 정정하거나 별칭에서 등록된 데이터 유형을 정정하십시오. 이렇게 하여 문제점이 정정되지 않는 경우, IBM 고객만족센터에 문의하십시오.
SQL1822N	데이터 소스 "Excel 랩퍼"에서 예기치 않은 오류 코드 "-1900.<internal program code>"를 받았습니다. 연관된 텍스트 및 토큰은 "메모리 할당 오류"입니다.	내부 프로그램 오류가 발생했습니다. IBM 고객만족센터에 문의하십시오.



---

## 제6장 별칭 대체

ALTER NICKNAME문을 사용하여 데이터 소스 또는 뷰의 연합 데이터베이스의 표현을 수정하십시오. 다음을 수행할 수 있습니다.

- 테이블 또는 뷰에서 컬럼의 지역 이름을 변경
- 이 컬럼의 지역 데이터 유형을 변경
- 이 컬럼의 옵션을 추가, 변경 또는 삭제

ALTER NICKNAME문에 대한 자세한 정보는 *DB2 SQL 참조서*를 참조하십시오.

---

### 컬럼 이름 변경

다음 예에서 SQL문은 지역 컬럼 이름을 DCODE에서 DRUGCODE로 변경합니다. 별칭 DRUGDATA1은 로컬 테이블 구조 파일인 drugdata1.txt를 나타냅니다. DCODE 컬럼은 파일의 첫 번째 필드를 참조하는 지역 컬럼 이름입니다.

```
ALTER NICKNAME DRUGDATA1
  ALTER COLUMN DCODE
  LOCAL NAME DRUGCODE
```

---

### 데이터 유형 변경

다음 예에서 SQL문은 DRUG 컬럼의 지역 데이터 유형을 CHAR(30)으로 변경합니다. DRUG 컬럼은 원래 CREATE NICKNAME문을 사용하여 CHAR(20)으로 정의됩니다. 별칭 DRUGDATA1은 로컬 테이블 구조 파일인 drugdata1.txt를 나타냅니다.

```
ALTER NICKNAME DRUGDATA1
  ALTER COLUMN DRUG
  LOCAL TYPE CHAR(30)
```

---

## 파일 경로 변경

다음 예에서 SQL문은 테이블 구조 파일, drugdata1.txt의 완전한 경로를 변경합니다. 경로는 원래 CREATE NICKNAME문을 사용하여 '/user/pat/drugdata1.txt'로 정의되었습니다. 별칭 DRUGDATA1은 로컬 테이블 구조 파일인 drugdata1.txt를 나타냅니다.

```
ALTER NICKNAME DRUGDATA1
  OPTIONS (SET FILE_PATH '/usr/kelly/data/drugdata1.txt')
```

---

## 주의사항

IBM은 다른 국가에서 이 책에 기술된 제품, 서비스 또는 기능을 제공하지 않을 수도 있습니다. 현재 사용할 수 있는 제품 및 서비스에 대한 정보는 한국 IBM 담당자에게 문의하십시오. 이 책에서 IBM 제품, 프로그램 또는 서비스를 언급했다고 해서 해당 IBM 제품, 프로그램 또는 서비스만을 사용할 수 있다는 것을 의미하지는 않습니다. IBM의 지적 재산을 침해하지 않는 한, 기능상으로 동등한 제품, 프로그램 또는 서비스를 대신 사용할 수도 있습니다. 그러나 비IBM 제품, 프로그램 또는 서비스의 운영에 대한 평가 및 검증은 사용자의 책임입니다.

IBM은 이 책에서 다루고 있는 특정 내용에 대해 특허를 보유하고 있거나 현재 특허 출원 중일 수 있습니다. 이 책을 제공한다고 해서 그러한 특허에 대한 사용권까지를 부여하는 것은 아닙니다. 사용권에 대한 의문사항은 다음으로 문의하십시오.

135-270

서울특별시 강남구 도곡동 467-12, 군인공제회관빌딩

한국 아이.비.엠 주식회사

고객만족센터

전화번호: 080-023-8080

2바이트(DBCS) 정보에 관한 사용권 문의는 한국 IBM 고객만족센터에 문의하거나 다음 주소로 서면 문의하시기 바랍니다.

IBM World Trade Asia Corporation

Licensing

2-31 Roppongi 3-chome, Minato-ku

Tokyo 106, Japan

다음 단락은 현지법과 상충하는 영국이나 기타 국가에서는 적용되지 않습니다. IBM은 타인의 권리 비침해, 상품성 및 특정 목적에의 적합성에 대한 묵시적 보증을 포함하여(단, 이에 한하지 않음) 묵시적이든 명시적이든 어떠한 종류의 보증없이 이

책을 『현상태대로』 제공합니다. 일부 국가에서는 특정 거래에서 명시적 또는 묵시적 보증의 면책사항을 허용하지 않으므로, 이 사항이 적용되지 않을 수도 있습니다.

이 정보에는 기술적으로 부정확한 내용이나 인쇄상의 오류가 있을 수 있습니다. 이 정보는 주기적으로 변경되며, 이 변경사항은 최신판에 통합됩니다. IBM은 이 책에서 설명한 제품 및/또는 프로그램을 사전 통고 없이 언제든지 개선 및/또는 변경할 수 있습니다.

이 정보에서 비IBM의 웹 사이트는 단지 편의상 제공된 것으로, 어떤 방식으로든 이들 웹 사이트를 옹호하고자 하는 것은 아닙니다. 해당 웹 사이트의 자료는 본 IBM 제품 자료의 일부가 아니므로 해당 웹 사이트의 사용으로 인한 위험은 사용자 본인이 감수해야 합니다.

IBM은 귀하의 권리를 침해하지 않는 범위 내에서 적절하다고 생각하는 방식으로 귀하가 제공한 정보를 사용하거나 배포할 수 있습니다.

(i) 독립적으로 작성된 프로그램 및 기타 프로그램(이 프로그램 포함)간의 정보 교환 (ii) 교환된 정보의 상호 이용을 목적으로 정보를 원하는 프로그램 사용권자는 다음 주소로 문의하십시오.

135-270

서울특별시 강남구 도곡동 467-12, 군인공제회관빌딩  
한국 아이.비.엠 주식회사  
고객만족센터

이러한 정보는 해당 조항 및 조건(예를 들면, 사용권 지불 포함)에 따라 사용할 수 있습니다.

이 정보에 기술된 사용권 프로그램 및 사용가능한 모든 사용권 자료는 IBM이 IBM 기본 계약, IBM 프로그램 사용권 계약(IPLA) 또는 이와 동등한 계약에 따라 제공한 것입니다.

여기에 있는 모든 성능 데이터는 제한된 환경에서 산출된 것입니다. 따라서, 다른 운영 환경에서 얻어진 결과는 상당히 다를 수 있습니다. 일부 측정치는 개발 레벨 시스템에서 작성되었을 수 있으며, 이러한 측정치가 일반적으로 사용가능한 시스템



에서도 동일하다고는 보장하지 않습니다. 더우기, 일부 측정치는 추정을 통해 추측 되었을 수도 있으므로 실제 결과는 다를 수 있습니다. 이 책의 사용자는 본인의 특정 환경에 적용할 수 있는 데이터를 검증해야 합니다.

비IBM 제품에 관한 정보는 해당 제품의 공급업체, 공개자료 또는 기타 범용 소스로부터 얻은 것입니다. IBM에서는 이러한 제품들을 테스트하지 않았으므로, 비IBM 제품과 관련된 성능의 정확성, 호환성 또는 배상 청구에 대해서는 확신할 수 없습니다. 비IBM 제품의 성능에 대한 의문사항은 해당 제품의 공급업체에 문의하십시오.

IBM이 제시하는 방향 또는 의도에 관한 모든 언급은 특별한 통지없이 변경될 수 있습니다.

이 정보에는 일상의 업무에서 사용되는 자료와 보고의 예제가 포함되어 있을 수 있습니다. 가능한 완벽하게 설명하기 위해 개인, 회사, 상표 및 제품의 이름이 예제에 들어 있습니다. 이들 이름은 모두 가공의 것이며, 실제 기업의 이름 및 주소와 유사하더라도 이는 전적으로 우연입니다.

저작권:

이 정보에는 여러 가지 운영 플랫폼에서의 프로그래밍 기법을 보여주는 원어로 된 샘플 응용프로그램이 포함되어 있을 수 있습니다. 샘플 응용프로그램의 작성 기준이 된 운영 플랫폼의 응용프로그램 프로그래밍 인터페이스에 부합하는 응용프로그램을 개발, 사용, 마케팅 또는 배포를 목적으로 이들 샘플 프로그램을 복사, 수정 및 배포할 수 있으며 IBM에 대한 지불 의무는 없습니다. 이들 예제 프로그램은 모든 조건에서 철저히 검사된 것은 아닙니다. 따라서, IBM은 이들 프로그램의 신뢰성, 서비스 가능성 또는 기능에 대해 어떠한 보증도 하지 않습니다.

각 사본이나 이들 샘플 프로그램의 일부 또는 파생본에는 다음과 같은 저작권 주의사항을 포함시켜야 합니다.

© (귀하의 회사명) (연도). 이 코드 부분은 IBM 샘플 프로그램에 나와 있습니다.

© Copyright IBM Corp. \_연도 입력\_. All rights reserved.

---

## 상표

별표(\*)로 표시된 다음의 용어는 전세계에서 IBM의 상표입니다.

ACF/VTAM	IBM
AISPO	IMS
AIX	IMS/ESA
AIX/6000	LAN DistanceMVS
AIXwindows	MVS/ESA
AnyNet	MVS/XA
APPN	Net.Data
AS/400	OS/2
BookManager CICS	OS/390
C Set++	OS/400
C/370	PowerPC
DATABASE 2	QBIC
DataHub	QMF
DataJoiner	RACF
DataPropagator	RISC System/6000
DataRefresher	RS/6000S/370
DB2	SP
DB2 ConnectDB2 Extenders	SQL/DS
DB2 OLAP Server	SQL/400
DB2 Universal Database	System/370
Distributed Relational	System/390
Database Architecture	SystemView
DRDA	VisualAge
eNetwork	VM/ESA
Extended Services	VSE/ESA
FFST	VTAM
First Failure Support Technology	WebExplorer
	WIN-OS/2

다음 용어는 기타 회사의 상표 또는 등록상표입니다.

Microsoft, Windows 및 Windows NT는 Microsoft Corporation의 상표 또는 등록상표입니다.

Java 또는 모든 Java 관련 상표와 로고 및 Solaris는 미국 또는 기타 국가에서 사용되는 Sun Microsystems, Inc.의 상표입니다.

Tivoli 및 NetView는 미국 또는 기타 국가에서 사용되는 Tivoli Systems Inc.의 상표입니다.

UNIX는 미국 또는 기타 국가에서 X/Open Company Limited가 독점적인 사용권을 가진 등록상표입니다.

두 개의 별표(\*\*)가 붙은 기타 회사 이름, 제품 이름 또는 서비스 이름은 해당 회사의 상표이거나 서비스표입니다.



---

## 관련 서적

이 관련 서적에는 DB2 Life Sciences Data Connect에서 작업하는 동안 유용한 DB2 Universal Database 책이 들어 있습니다.

- *DB2 Connect* 사용자 안내서(SA30-0993)
- *UNIX용 DB2 빠른 시작*(GA30-0984)
- *DB2 SQL 참조서*(SA30-0997)
- *DB2 관리 안내서: 계획*(SA30-0990)
- *DB2 관리 안내서: 구현*(SA30-0988)
- *DB2 관리 안내서: 성능*(SA30-0989)
- *DB2 메시지 참조서*(GA30-0986)
- *IBM DB2 Universal Database 릴리스 정보 버전 7.2/버전 7.1 FixPak 4*



# 색인

## [ 라 ]

### 랩퍼

- 정의 2
- 테이블 구조 파일 9
- 플랫폼 5
- 플랫폼별 기본 라이브러리 이름 8
- Documentum 23
- Excel 57

## [ 마 ]

### 메시지

- 테이블 구조 파일 19
- Documentum 랩퍼 51
- Excel 랩퍼 66

## [ 바 ]

### 별칭 대체

- 73

## [ 사 ]

- 생명 과학 데이터 소스 1
- 설치 5

## [ 아 ]

### 연합 데이터베이스 시스템

- 1

## [ 타 ]

### 테이블 구조 파일

- 개요 9
- 메시지 19
- 연합 시스템에 추가  
랩퍼 등록 11

### 테이블 구조 파일 (계속)

#### 연합 시스템에 추가 (계속)

- 별칭 등록 13
- 서버 등록 12
- 예 9
- 유형 10
- 제한사항 및 고려사항 17
- 최적화 18
- 파일 액세스 제어 모델 18
- DB2 Life Sciences Data Connect에  
서의 액세스 10

## C

### CREATE FUNCTION문

- Documentum 35

### CREATE NICKNAME문

- 테이블 구조 파일 13
- Documentum 30
- Excel 파일 61

### CREATE SERVER문

- 테이블 구조 파일 12
- Documentum 27
- Excel 파일 60

### CREATE USER MAPPING문

- Documentum 30

### CREATE WRAPPER문

- 테이블 구조 파일 11
- Documentum 27
- Excel 파일 59

### CreateNicknameFile 유틸리티,

- Documentum 46

## D

### DB2 Life Sciences Data Connect, 설명

- 1

### DB2\_DJ\_COMM 환경 변수 12, 27

### DiscoveryLink 3

### Documentum

- 메시지 51
- 문서에 대한 사용자 액세스 51
- 반복 속성 정의 이중화 48
- 설명 23
- 연합 시스템에 추가  
랩퍼 등록 27
- 별칭 등록 30
- 사용자 맵핑 30
- 사용자 정의 함수 등록 35
- 서버 등록 27

### CREATE FUNCTION문 35

### CREATE NICKNAME문 30

### CREATE SERVER문 27

### CREATE USER

#### MAPPING문 30

### CREATE WRAPPER문 27

### CreateNicknameFile 유틸리티 46

### 예 24

### 제한 및 고려사항 49

### CreateNicknameFile 유틸리티 46

## E

### Excel 파일

- 메시지 66
- 샘플 사용자 시나리오 63
- 설명 57

### 연합 시스템에 추가

- 랩퍼 등록 59
- 별칭 등록 61
- 서버 등록 60

### 예 58

Excel 파일 (계속)

제한 및 고려사항 65

파일 액세스 제어 모델 66



---

## IBM에 문의

기술적인 문제가 발생한 경우에는 IBM 고객만족센터에 문의하기 전에 문제점 해결 안내서에서 제안한 조치를 검토하고 실행해 보십시오. 이것은 IBM 고객만족센터로 하여금 사용자를 보다 더 잘 지원할 수 있도록 사용자가 모을 수 있는 정보를 제공합니다.

DB2 Universal Database 제품에 대한 정보나 주문은 지방 사무소의 IBM 담당자나 공인 IBM 소프트웨어 재판매업자에게 문의하십시오.

미국에 사시는 분은 다음 번호 중 하나를 선택하여 전화하십시오.

- 고객 지원을 받으려면, 1-800-237-5511.
- 사용가능한 서비스 옵션을 알려면, 1-888-426-4343.

---

## 제품 정보

미국에 사시는 분은 다음 번호 중 하나를 선택하여 전화하십시오.

- 제품 주문이나 일반 정보를 얻으려면 1-800-IBM-CALL(1-800-426-2255) 또는 1-800-3IBM-OS2(1-800-342-6672).
- 책에 대한 주문은, 1-800-879-2755.

**<http://www.ibm.com/software/data/>**

DB2 World Wide Web 페이지에서는 새 소식, 제품 설명, 교육 일정 등에 관한 현재 DB2 정보를 제공합니다.

**<http://www.ibm.com/software/data/db2/library/>**

DB2 제품 및 서비스 기술 라이브러리는 빈도 높은 질문(FAQ), 수정사항(Fixes), 책 및 최신 DB2 기술 정보에 대한 액세스를 제공합니다.

주: 이러한 정보는 영어로만 제공됩니다.

**<http://www.elink.ibm.com/pbl/pbl/>**

여기에서는 책을 웹 사이트에서 주문할 수 있는 방법을 제공합니다.

**<http://www.ibm.com/education/certify/>**

IBM 웹 사이트에서 기술 전문 인증 프로그램은 DB2를 포함하여 다른 IBM 제품의 기술 전문 인증 테스트 정보를 제공합니다.

**<ftp.software.ibm.com>**

anonymous로 로그인하십시오. /ps/products/db2 디렉토리에서, DB2와 많은 관련 제품에 관한 데이터, 수정사항, 도구 등을 찾을 수 있습니다.

**<comp.databases.ibm-db2>, <bit.listserv.db2-l>**

이러한 인터넷 뉴스 그룹으로 사용자는 DB2 제품에 대한 자신의 사용 경험을 토론할 수 있습니다.

**Compuserve에서, GO IBMDB2**

이 명령을 입력하여 IBM DB2 계열 포럼을 액세스하십시오. 모든 DB2 제품은 이 포럼을 통해 지원됩니다.

미국 외 지역에서 IBM에 연락하는 방법에 관한 정보는 *IBM Software Support Handbook*의 Appendix A를 참조하십시오. 이 문서에 액세스하려면, 웹 사이트 <http://www.ibm.com/support/>로 가서 페이지 맨 밑에 있는 IBM Software Support Handbook 링크를 누르십시오.

주: 일부 국가에서는 IBM-authorized 공인 딜러는 IBM 지원 센터 대신 해당 딜러 지원 부서에 연락해야 합니다.





부품 번호: CT0M8KO

Printed in Singapore

CT0M8KO

