

IBM DB2 Cube Views



# 비즈니스 모델링 시나리오

버전 8



IBM DB2 Cube Views



# 비즈니스 모델링 시나리오

버전 8

**주!**

이 정보와 이 정보가 지원하는 제품을 사용하기 전에, 33 페이지의 『주의사항』의 정보를 읽으십시오.

이 문서에는 IBM®의 소유권에 대한 정보가 들어 있습니다. 이것은 라이선스 계약에 따라 제공되었으며 저작권 법의 보호를 받습니다. 이 책에 있는 정보에는 어떠한 제품 보증도 포함되어 있지 않으며, 이 매뉴얼의 모든 사항을 있는 그대로 해석해서는 안됩니다.

IBM 책은 온라인 또는 한국 IBM 담당자를 통해 주문할 수 있습니다.

- 책을 온라인으로 주문하려면, [www.ibm.com/shop/publications/order](http://www.ibm.com/shop/publications/order)에서 IBM Publications Center로 가십시오.
- 한국 IBM 담당자를 찾으려면, [www.ibm.com/planetwide](http://www.ibm.com/planetwide)에서 IBM Directory of Worldwide Contacts로 가십시오.

미국 또는 캐나다에서 DB2 마케팅 및 판매처로부터 DB2® 책을 주문하려면, 1800-IBM-4YOU(425-4968)로 문의하십시오.

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

---

# 목차

이 책에 대하여 . . . . .	v	제 4 장 인터넷 주문 숫자 카운팅. . . . .	21
이 책의 사용자 . . . . .	v	시나리오 세부사항 . . . . .	21
제 1 장 웨어하우스에 시간 초과된 재고의 플로우 및 값 계산. . . . .	1	치수 작성 단계 . . . . .	22
시나리오 세부사항 . . . . .	1	제 5 장 판매액 수치 순위 . . . . .	25
치수 작성 단계 . . . . .	4	시나리오 세부사항 . . . . .	25
제 2 장 광고 비용과 판매액의 상관관계 . . . . .	9	치수 작성 단계 . . . . .	27
시나리오 세부사항 . . . . .	9	제 6 장 사실 테이블에 저장된 시간 데이터를 사용하여 시간 차원 작성. . . . .	29
치수 작성 단계 . . . . .	9	시나리오 세부사항 . . . . .	29
제 3 장 이익 및 상점의 이익 마진 계산 . . . . .	15	속성 및 차원 작성 단계 . . . . .	31
시나리오 세부사항 . . . . .	15	주의사항 . . . . .	33
치수 작성 단계 . . . . .	16	상표 . . . . .	35



---

## 이 책에 대하여

이 책은 DB2 Cube Views 메타데이터를 사용하는 일반 실제 비즈니스 시나리오의 모델화 방법에 관한 *DB2 Cube Views* 설치 및 사용자 안내서에 대한 보충 정보를 제공합니다.

---

## 이 책의 사용자

OLAP 메타데이터 및 DB2 Universal Database™(DB2)에 대해 작업하는 데이터베이스 관리자(DBA)인 경우 이 책을 읽으십시오. 사용자는 다음 사항을 잘 알고 있어야 합니다.

- 스타 스키마와 같은 OLAP 개념
- 큐브 모델, 사실 오브젝트, 차원, 조인, 치수 및 속성과 같은 DB2 Cube Views 메타데이터 오브젝트





---

## 제 1 장 웨어하우스에 시간 초과된 재고의 플로우 및 값 계산

소매 사업인 XYZ Retail은 재고가 판매를 위해 특정 상점으로 보내지기 전에, 웨어하우스에 이를 보관합니다. XYZ Retail은 웨어하우스에 시간 초과된 재고 상태에 대한 데이터를 유지보수하며 이 데이터를 분석하려고 합니다. 특히, 회사는 웨어하우스의 다음 두 측면을 조사하려고 합니다.

- 웨어하우스로 들어오고 나가는 상품의 플로우
- 특정 시간에 웨어하우스의 상품 값

첫 번째 측면은 상품의 플로우가 시간 초과된 데이터 조사를 수반한다는 것입니다. 두 번째 측면은 상품의 값이 특정 시점에서 웨어하우스의 스냅샷을 취한다는 것입니다.

---

### 시나리오 세부사항

XYZ Retail에는 QUANTITY\_IN, QUANTITY\_OUT, CURRENT\_QUANTITY, PRODUCT\_VALUE, PRODUCT\_ID 및 TIME\_ID와 같은 웨어하우스 관련 컬럼이 있는 사실 테이블이 있습니다. 이 데이터는 매주 테이블에 입력됩니다. 또한 데이터베이스에는 제품 테이블 및 시간 테이블도 있습니다. 예를 들어, 샘플 사실 테이블 데이터 세트가 표 1에 표시되어 있습니다.

표 1.

PRODUCT_ID	TIME_ID	QUANTITY_IN	QUANTITY_OUT	CURRENT_QUANTITY	PRODUCT_VALUE
1234	1	5	0	5	5
1234	2	20	10	15	5
1234	3	10	20	5	5

한 제품 유형이 웨어하우스로 여러 번 들어오고 나갈 수 있으므로, 세 개의 각 샘플 데이터 항목에 대해 PRODUCT\_ID 값은 동일합니다.

XYZ Retail의 DBA는 다음과 같은 세 가지의 다른 치수를 작성해야 합니다.

## 플로우 입력

상품의 플로우를 웨어하우스로 모델화합니다.

## 플로우 출력

웨어하우스의 플로우 출력을 모델화합니다.

## 현재 값

특정 시간에 상품의 값을 모델화합니다.

첫 번째 두 치수인 플로우 입력 및 플로우 출력을 작성하기 위해, DBA는 QUANTITY\_IN 및 QUANTITY\_OUT 컬럼으로 각각 맵핑되고 모든 차원에 걸쳐 데이터를 합산하는 치수를 작성합니다. 데이터가 모든 차원에 걸쳐 SUM 함수만을 사용하여 집계되므로 이는 전체 추가 치수로 알려져 있습니다. 예를 들어, 표 2에서는 1234라는 PRODUCT\_ID를 갖는 제품에 대해 3개월 동안의 QUANTITY\_IN 및 QUANTITY\_OUT 컬럼의 샘플 데이터 세트를 표시합니다. 플로우 입력 및 플로우 출력 치수는 이러한 월별 값을 합산하여 분기 동안 웨어하우스로 들어오고 나가는 전체 수량을 계산합니다.

표 2. PRUDUCT\_ID 1234의 플로우 입력 및 플로우 출력 전체 추가 치수에 대한 샘플 데이터 계산

	1월	2월	3월	1/4분기
QUANTITY_IN	5	20	10	35
QUANTITY_OUT	0	10	20	30

전체 추가 치수는 작성되는 치수 중 가장 단순하고 일반적이어서, 보다 복잡한 치수에 대한 블록을 빌드하는 데 자주 사용됩니다. 숫자 소스 데이터에 근거하는 치수의 경우, OLAP 센터는 전체 추가 치수를 디폴트로 작성합니다.

세 번째 치수인 현재 값을 작성하기 위해, DBA는 PRODUCT\_VALUE와 CURRENT\_QUANTITY를 곱하여 값을 계산하는 계산된 치수를 작성합니다. 예를 들어, PRODUCT\_ID=1234인 제품의 값이 5일 경우, 샘플 데이터의 현재 값 치수는 표 3에 표시됩니다.

표 3. PRODUCT\_ID 1234의 현재 값 치수에 대한 샘플 데이터 계산

	1월	2월	3월
CURRENT_QUANTITY	5	10	20

표 3. *PRODUCT\_ID* 1234의 현재 값 치수에 대한 샘플 데이터 계산 (계속)

	1월	2월	3월
현재 값	25	50	100

그런 다음, 이 데이터는 차원에 걸쳐 집계되어야 합니다. 그러나 특정 시점에서 이 값이 계산되고 있으므로, 시간 차원에 걸쳐 합산하는 것은 합당하지 않습니다. 대신, 집계는 제품 차원에 걸쳐 데이터를 합산하고 시간 초과된 데이터의 평균을 찾습니다. 일부 집계만 SUM 함수를 수반하므로 이는 세미 추가 치수로 알려져 있습니다.

월을 분기에 추가하는 것이 이치에 맞지 않으므로, 스냅샷 데이터를 계산하는 치수 및 월별 재고 데이터와 같은 특정 시점을 나타내는 데이터는 세미 추가 치수입니다. 전체 분기 동안 제품이 웨어하우스에 남아 있는 경우, 그 제품은 분기의 각 3개월 동안의 웨어하우스 재고의 CURRENT\_QUANTITY 스냅샷 데이터에 포함됩니다. CURRENT\_QUANTITY 데이터가 시간 초과되어 합산되는 경우, 3개월 동안 웨어하우스에 있던 제품은 세 번 계산됩니다. 표 4에서 표시한 대로, 1/4 분기의 값 25에는 웨어하우스의 활동에 대한 유의값이 없습니다. 테이블은 웨어하우스에 25개의 제품이 있지 않았으므로 25 제품의 값을 계산하는 것은 의미가 없다는 것을 보여줍니다.

표 4. *PRODUCT\_ID* 1234의 시간 차원에 대한 SUM 함수를 사용하는 CURRENT\_QUANTITY 컬럼의 샘플 데이터 계산

	1월	2월	3월	1/4분기
SUM(CURRENT_QUANTITY)	5	15	5	25

모든 차원에 걸쳐 SUM 함수를 사용하는 대신에, 시간 차원에 대해 AVG, MIN 및 MAX와 같은 기타 집계 함수를 수행할 수 있습니다. 예를 들어, 1월, 2월 및 3월에 대한 동일한 샘플 데이터 세트를 사용하여 4 페이지의 표 5에 표시된 대로 시간 차원에 대한 두 번째 집계 함수를 사용하여 해당 분기의 의미있는 값을 작성할 수 있습니다. 현재 값 치수는 해당 분기에 웨어하우스에 저장된 상품의 평균 총계 값 또는 해당 분기의 특정 시점에서의 최대 또는 최소값을 나타냅니다.

표 5. *PRODUCT\_ID* 1234의 시간 차원에 대한 *AVG*, *MAX* 및 *MIN* 함수를 사용하는 *CURRENT\_QUANTITY* 컬럼의 샘플 데이터 계산

	1월	2월	3월	1/4분기
<i>AVG</i> ( <i>CURRENT_QUANTITY</i> )	5	15	5	8.3
<i>MAX</i> ( <i>CURRENT_QUANTITY</i> )	5	15	5	15
<i>MIN</i> ( <i>CURRENT_QUANTITY</i> )	5	15	5	5

## 치수 작성 단계

다음 단계는 사용자가 OLAP 센터 사실 등록 정보 창을 사용하여 기존 사실 오브젝트에서 플로우 입력, 플로우 출력 및 현재 값 치수를 작성하는 방법을 설명합니다.

1. 사실 등록 정보 창을 열려면, OLAP 센터 오브젝트 트리에서 사실 오브젝트를 마우스 오른쪽 단추로 누르고 치수 편집을 누르십시오. 사실 등록 정보 창이 열립니다.
2. 다음과 같이 플로우 입력 치수를 작성하십시오.
  - a. 치수 페이지에서, 계산된 치수 작성을 눌러 플로우 입력 치수를 작성하십시오. SQL 표현식 빌더 창이 열립니다.
  - b. SQL 표현식 빌더 창에서, 이름 필드에 *FLOW IN*을 입력하십시오.
  - c. 플로우 입력 표현식을 작성하려면, 다음 단계를 완료하십시오.
    - 데이터 목록에서 사실 테이블 및 컬럼 폴더를 펼치십시오.
    - **QUANTITY\_IN** 컬럼을 더블 클릭하여 이를 표현식에 추가하십시오.
    - 확인을 눌러 SQL 표현식 빌더 창을 닫으십시오. 집계 페이지에서 디폴트 집계 함수인 *SUM*을 변경할 필요는 없습니다. 데이터 소스가 숫자이고 치수가 기존 치수 뿐만 아니라 컬럼도 참조하므로 *SUM* 함수는 플로우 입력 치수에 대한 기본값입니다.
3. 다음과 같이 플로우 출력 치수를 작성하십시오.
  - a. 치수 페이지에서, 계산된 치수 작성을 눌러 플로우 출력 치수를 작성하십시오. SQL 표현식 빌더 창이 열립니다.

- b. SQL 표현식 빌더 창에서, 이름 필드에 FLOW OUT을 입력하십시오.
  - c. 플로우 출력 표현식을 작성하려면, 다음 단계를 완료하십시오.
    - 데이터 목록에서 사실 테이블 및 컬럼 폴더를 펼치십시오.
    - **QUANTITY\_OUT** 컬럼을 더블 클릭하십시오.
  - d. 확인을 눌러 SQL 표현식 빌더 창을 닫으십시오. 집계 페이지에서 디폴트 집계 함수인 SUM을 변경할 필요는 없습니다. 데이터 소스가 숫자이고 치수가 기존 치수 뿐만 아니라 컬럼도 참조하므로, SUM 함수는 플로우 출력 치수에 대한 기본값입니다.
4. 다음과 같이 현재 값 치수를 작성하십시오.
- a. 치수 페이지에서, 계산된 치수 작성을 눌러 현재 값 치수를 작성하십시오. SQL 표현식 빌더 창이 열립니다.
  - b. SQL 표현식 빌더 창에서, 이름 필드에 CURRENT VALUE를 입력하십시오.
  - c. 현재 값 표현식을 작성하려면, 다음 단계를 완료하십시오.
    - 데이터 목록에서 사실 테이블 및 컬럼 폴더를 펼치십시오.
    - 데이터 목록에서 **PRODUCT\_VALUE** 컬럼을 더블 클릭하십시오.
    - 연산자 목록에서 \* 연산자를 더블 클릭하십시오.
    - 데이터 목록에서 **CURRENT\_QUANTITY** 컬럼을 더블 클릭하십시오.
- 6 페이지의 그림 1에서는 SQL 표현식 빌더 창에서 작성할 수 있는 현재 값 표현식을 표시합니다.

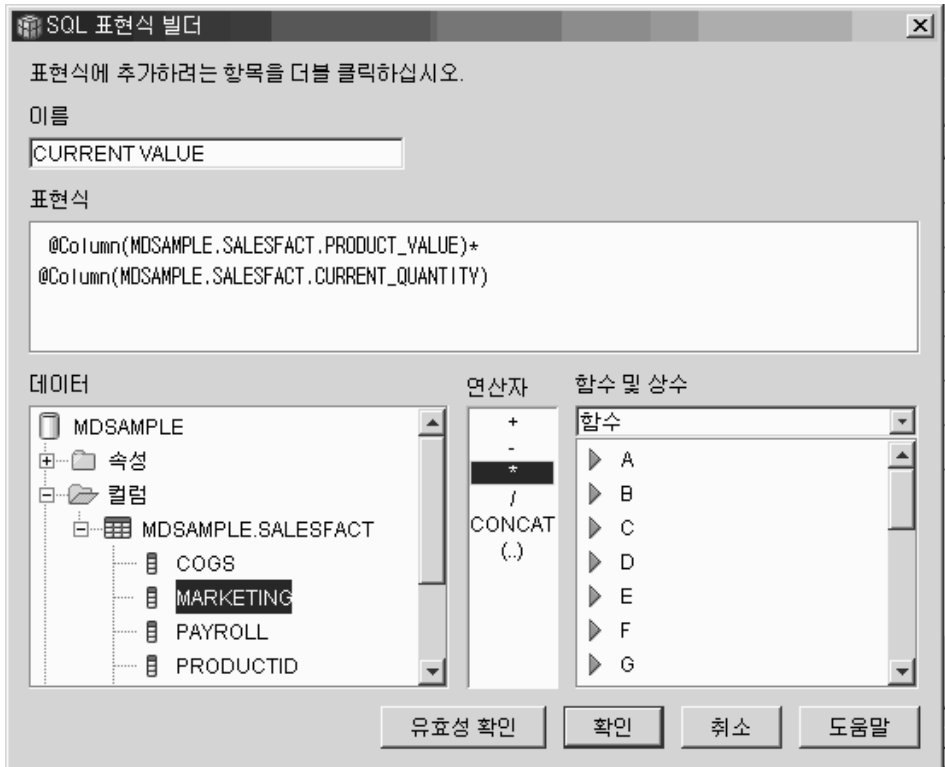



그림 1. SQL 표현식 빌더 창에서 Current Value 표현식을 완료하십시오.

- d. 확인을 눌러 SQL 표현식 빌더 창을 닫으십시오.
- e. 집계 페이지에서, 현재 값 치수에 대한 집계를 누르고 목록에서 집계 스크립트를 누르십시오. 집계 스크립트 빌더 창이 열립니다. 디폴트 집계 스크립트에는 모든 차원에 사용되는 SUM 함수가 있습니다.
- f. 필요한 경우, **Time**을 선택하고  누름 단추를 눌러 시간 차원을 아래로 이동하여 스크립트에서 마지막으로 나열되는 차원이 되도록 하십시오. 그런 다음, 시간 차원을 선택하고 컬럼 함수 목록에서 **AVG** 함수를 더블 클릭하십시오. 7 페이지의 그림 2에서 표시된 대로, 집계 스크립트가 시간을 제외한 모든 차원에 걸쳐 데이터(평균)를 합산합니다.

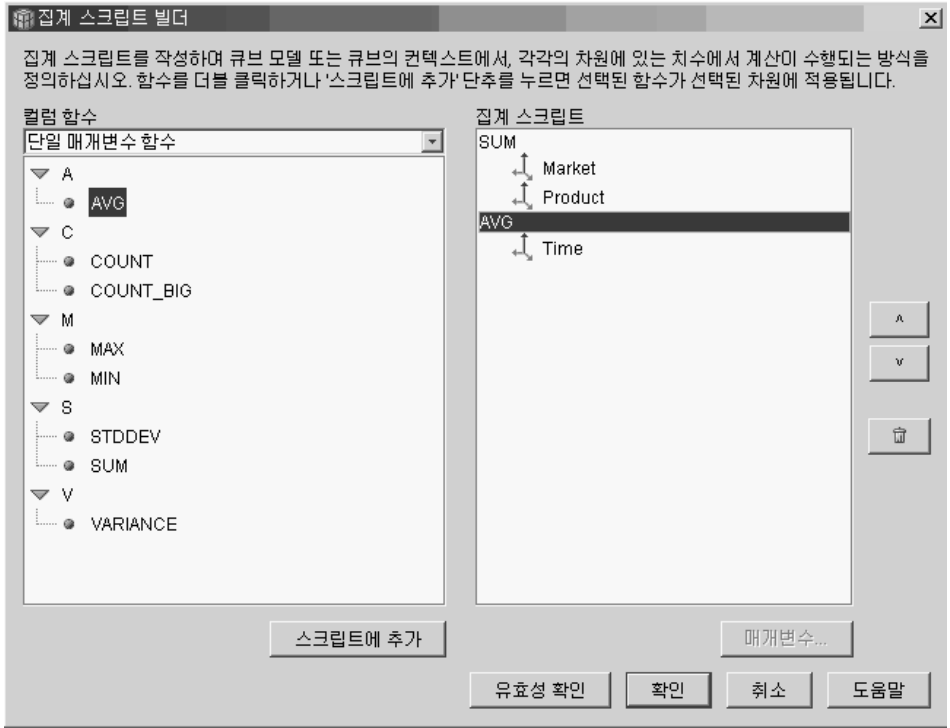


그림 2. 현재 값 치수에 대한 집계 스크립트

- g. 집계 스크립트 빌더 창에서, 유효성 확인을 눌러 집계 스크립트가 유효한 지 확인하십시오. 확인을 눌러 집계 스크립트를 저장하고 창을 닫으십시오.
5. 확인을 눌러 사실 오브젝트에 변경사항을 저장하고 사실 등록 정보 창을 닫으십시오.

이제 사용자에게 웨어하우스의 재고에 대한 세 가지의 계산된 치수가 있습니다. 이러한 치수를 사용하여 웨어하우스로 들어오고 나가는 제품 플로우의 패턴을 분석할 수 있습니다.





---

## 제 2 장 광고 비용과 판매액의 상관관계

자동차 판매점이 광고에 대한 지출 증가를 고려하고 있습니다. 올바른 결론을 내기 위해 판매점은 우선 광고 지출과 판매액 사이의 기록상의 관계를 분석하려고 합니다. 판매점은 광고의 레벨을 다양화하는 것이 판매액에 영향을 미치는지 여부 및 특히 광고의 증가와 판매액 증가가 밀접한 연관이 있는지를 판별하려고 합니다.

---

### 시나리오 세부사항

판매점의 데이터베이스에는 판매액 및 광고 비용 컬럼이 있는 사실 테이블이 있습니다. 그 데이터베이스에는 몇몇 기타 차원 테이블도 있습니다. DBA는 DB2 CORRELATION 함수를 사용하여 비용 및 판매액 사이의 상관 계산을 수행할 수 있습니다. CORRELATION 함수는 두 개의 입력 매개변수를 요구하는 다중 매개변수 함수입니다. 이 경우, DBA는 판매액 및 광고 비용 컬럼을 두 개의 입력 매개변수로 사용합니다.

DBA는 먼저 다중 매개변수 집계 함수를 집계 스크립트에 적용해야 합니다. 다중 매개변수 함수는 전체 차원에 걸쳐 적용될 수 있거나, 시간 차원을 제외한 모든 차원에 먼저 적용된 후 MAX 함수와 같은 두 번째 함수가 시간 함수에 적용될 수 있습니다. DBA가 치수에 대한 SQL 표현식을 정의하므로 이는 직접 광고 비용 컬럼으로 맵핑됩니다. 다중 매개변수 함수에서 사용되는 두 매개변수 중 SQL 표현식이 우선 순위입니다. DBA는 두 번째 매개변수를 판매액 컬럼에 직접 맵핑하는 SQL 표현식으로 정의합니다. CORRELATION 함수가 유일한 집계 함수로 정의되어 있으므로, 치수는 모든 차원에 걸쳐 광고 비용 및 판매액 결과 사이의 통계적 상관을 계산할 수 있습니다.

---

### 치수 작성 단계

다음 단계는 OLAP 센터 사실 등록 정보 창을 사용하여 기존 사실 오브젝트의 광고-판매액 상관 치수를 작성하는 방법을 설명합니다.

1. OLAP 센터 오브젝트 트리에서 사실 오브젝트를 마우스 오른쪽 단추로 누르고 치수 편집을 눌러 사실 등록 정보 창을 여십시오.
2. 계산된 치수 작성 누름 단추를 누르십시오. SQL 표현식 빌더 창이 열립니다.
3. SQL 표현식 빌더 창에서, 이름 필드에 ADVERTISING-SALES CORRELATION 을 입력하십시오.
4. 집계 스크립트에서 다중 매개변수 CORRELATION 함수의 첫 번째 매개변수로 사용되기도 하는 치수의 표현식을 정의하십시오. 표현식을 정의하려면, 데이터 목록의 치수 폴더를 펼치고 **AD COSTS** 치수를 더블 클릭하여 이를 표현식 목록에 추가하십시오. 그림 3에서는 SQL 표현식 빌더 창에서 작성하는 표현식을 표시합니다.

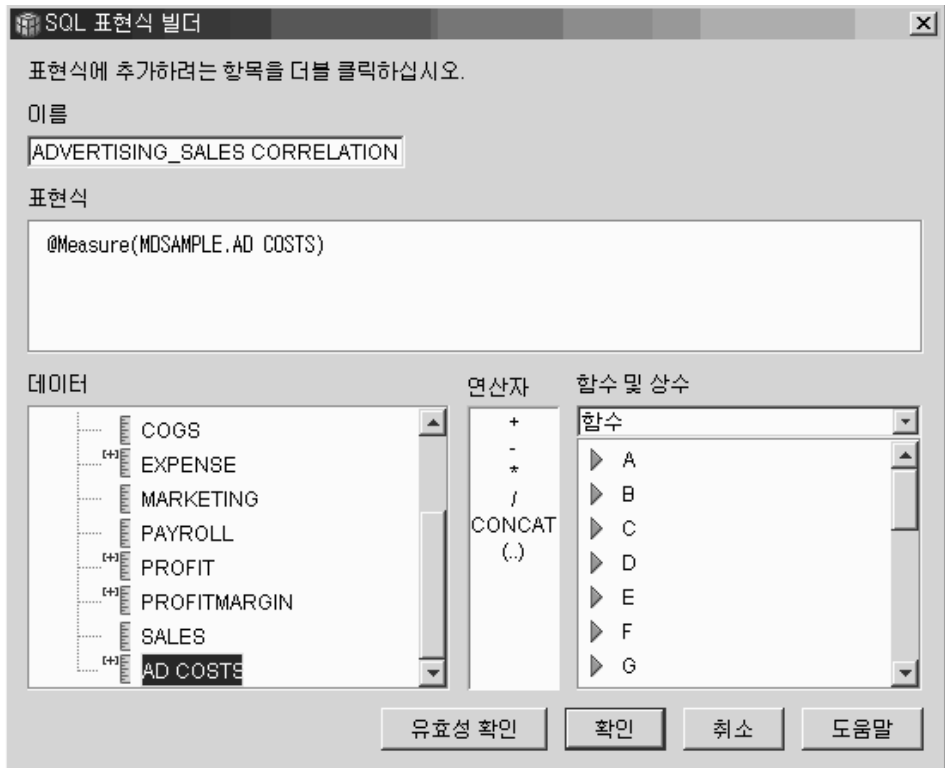


그림 3. SQL 표현식 빌더 창에서 advertising-sales correlation 상관 표현식을 완료하십시오

5. 집계 페이지에서, **ADVERTISING-SALES CORRELATION** 치수에 대한 집계 함수를 누르고 그림 4에 표시된 대로 집계 스크립트를 선택하십시오. 집계 스크립트 빌더 창이 열립니다.

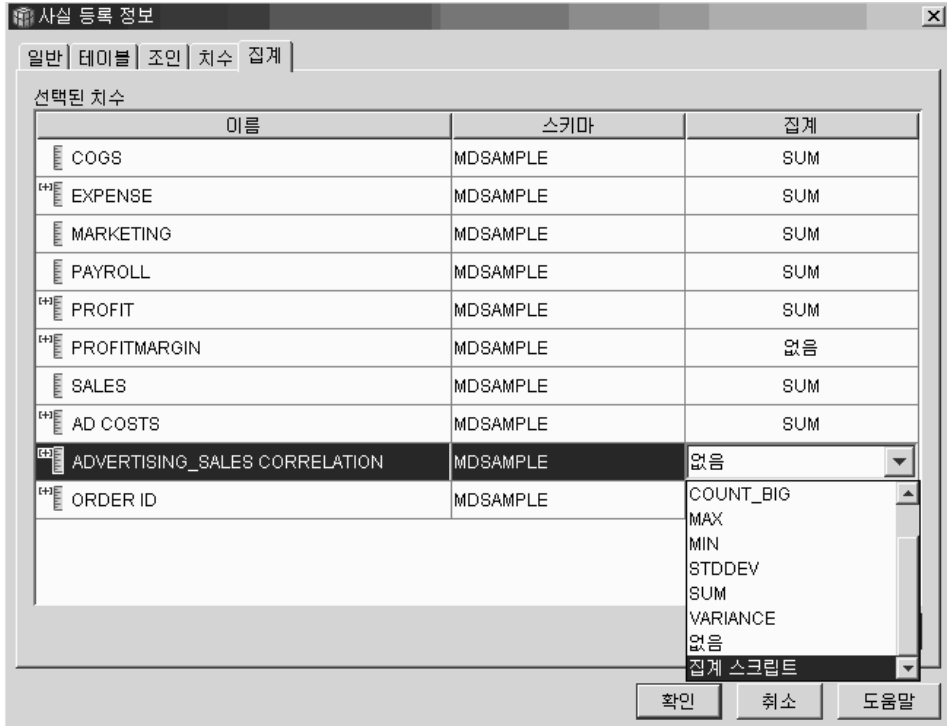


그림 4. 사실 등록 정보 창의 집계 페이지

6. 컬럼 함수 필드에서, 다중 매개변수 함수를 선택하십시오. 다중 매개변수 함수 목록에서, **CORRELATION** 함수를 선택하고 스크립트에 추가를 누르십시오. 12 페이지의 그림 5에서는 CORRELATION 함수가 스크립트의 차원 목록 위쪽에 있음을 표시합니다.

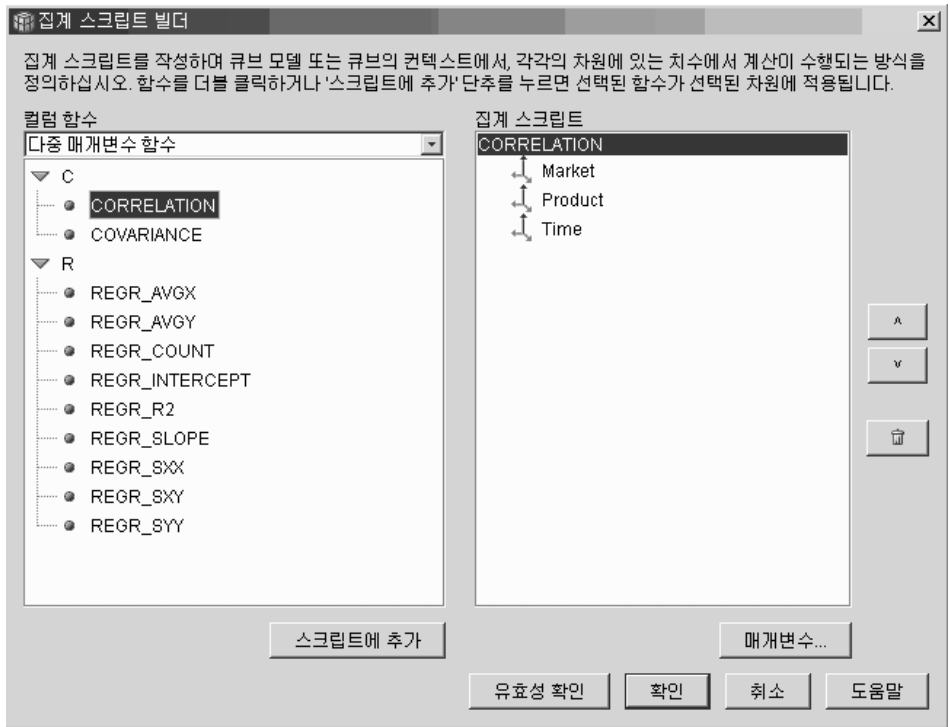


그림 5. 광고-판매액 상관 치수에 대한 집계 스크립트

7. 매개변수 누름 단추를 눌러 CORRELATION 함수에 대한 두 번째 매개변수를 지정하십시오. 함수 매개변수 창이 열립니다. 기존 치수 사용을 선택한 후 SALES를 선택하십시오. 13 페이지의 그림 6에서는 두 번째 매개변수로 지정된 판매액 치수가 있는 함수 매개변수 창을 표시합니다. 확인을 눌러 선택사항을 저장하고 함수 매개변수 창을 닫으십시오.

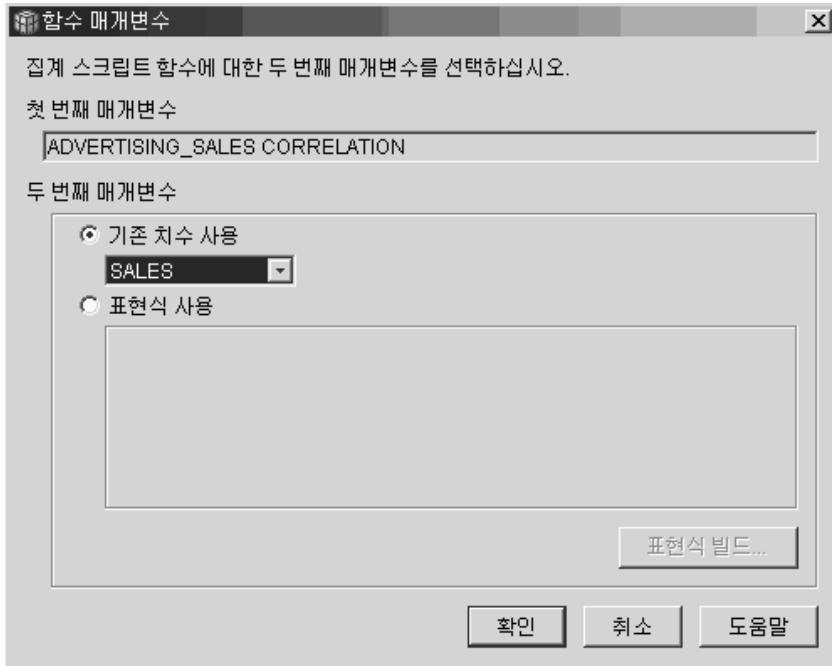


그림 6. 함수 매개변수 창에서 두 번째 매개변수로 지정된 판매액 치수

8. 집계 스크립트 빌더 창에서, 유효성 확인 누름 단추를 눌러 집계 스크립트가 유효한지 확인하십시오. 확인을 눌러 집계 스크립트를 저장하고 창을 닫으십시오.
9. 확인을 눌러 사실 오브젝트에 변경사항을 저장하고 사실 등록 정보 창을 닫으십시오.

이제 데이터베이스에 두 가지의 데이터 유형을 상관시키는 치수가 있습니다. 이 치수를 사용함으로써 기록된 경향의 결과에 근거하여 이후의 광고 지출을 결정할 수 있습니다.



---

## 제 3 장 이익 및 상점의 이익 마진 계산

장난감 상점의 총 지배인은 제품 유형 및 시기와 같은 여러 요소가 이익 및 이익 마진에 어떻게 영향을 주는지 분석하려고 합니다. 고급 분석을 완료하기 전에, 장난감 상점의 DBA는 이익 및 이익 마진 치수를 먼저 작성해야 합니다. 그런 다음, DBA는 이익 및 이익 마진 치수와 다른 인수들을 상관시키고 비교하는 추가 치수를 작성할 수 있습니다.

---

### 시나리오 세부사항

장난감 상점의 데이터베이스에는 몇몇 차원 테이블 각각에 해당되는 외부 키 컬럼 뿐만 아니라 판매액, 판매된 상품의 비용(COGS) 및 지출 컬럼이 있는 사실 테이블이 있습니다. DBA는 판매액, COGS 및 지출 컬럼에 각각 맵핑되는 판매액, COGS 및 지출 치수를 이미 작성했습니다. 이익 및 이익 마진 치수는 이러한 기존 치수로부터 전적으로 작성될 수 있습니다.

이익 치수를 작성하려면, DBA는 SQL 표현식의  $SALES - (COGS + EXPENSE)$ 를 계산하는 치수를 작성하고 모든 차원에 걸쳐 계산된 데이터를 합산합니다. 기존 치수 또는 컬럼 또는 둘의 조합을 참조하여 이익 치수를 작성할 수 있습니다.

이익 치수가 작성된 후에, DBA는 이익 마진 치수를 작성할 수 있습니다. 이익 마진 치수는 백분율( $(\text{이익}/\text{판매액}) * 100$ )로 표현되는 기존 두 치수의 비율이며 해당 집계 함수가 필요하지 않습니다. 치수는 이미 집계된 데이터의 다른 치수를 참조하기만 하므로 집계 함수가 필요하지 않습니다. DBA가 다른 치수를 참조하기만 하는 혼합 치수를 사용하여 비율을 계산할 경우, DBA는 추가 집계를 정의할 필요가 없습니다. SUM과 같은 대부분의 집계 함수는 비율에는 적용되지 않습니다. 예를 들어, 장난감 상점이 전체 4분기 동안 40%, 32%, 28%, 37%의 이익 마진을 남긴 경우, 전체 비율을 합산하여 1년에 137%의 이익 마진을 남긴다는 것은 이치에 맞지 않습니다.

---

## 치수 작성 단계

다음 단계는 사용자가 OLAP 센터 사실 등록 정보 창을 사용하여 기존 사실 오브젝트에서 이익 및 이익 마진 치수를 작성하는 방법을 설명합니다.

1. 사실 등록 정보 창을 열려면, OLAP 센터 오브젝트 트리에서 사실 오브젝트를 마우스 오른쪽 단추로 누르고 치수 편집을 누르십시오. 사실 등록 정보 창이 열립니다.
2. 다음과 같이 이익 치수를 작성하십시오.
  - a. 치수 페이지에서, 계산된 치수 작성 누름 단추를 누르십시오. SQL 표현식 빌더 창이 열립니다.
  - b. SQL 표현식 빌더 창에서, 이름 필드에 PROFIT을 입력하십시오.
  - c. 이익 표현식을 작성하려면, 치수 폴더를 데이터 목록에서 펼친 후 다음 단계를 완료하십시오.
    - 데이터 목록에서 **SALES** 치수를 더블 클릭하여 이를 표현식에 추가하십시오.
    - 연산자 목록에서 - 연산자를 더블 클릭하십시오.
    - 데이터 목록에서 **COGS** 치수를 더블 클릭하십시오.
    - 연산자 목록에서 + 연산자를 더블 클릭하십시오.
    - 데이터 목록에서 **EXPENSE** 치수를 더블 클릭하십시오.
    - 표현식 필드에서 @Measure(MDSAMPLE.COGS)+@Measure(MDSAMPLE.EXPENSE) 부분을 강조 표시한 후 연산자 목록에서 (..)을 더블 클릭하여 괄호 안 표현식의 선택 부분을 닫으십시오.

17 페이지의 그림 7에서는 SQL 표현식 빌더 창에서 작성할 수 있는 이익 표현식을 표시합니다.



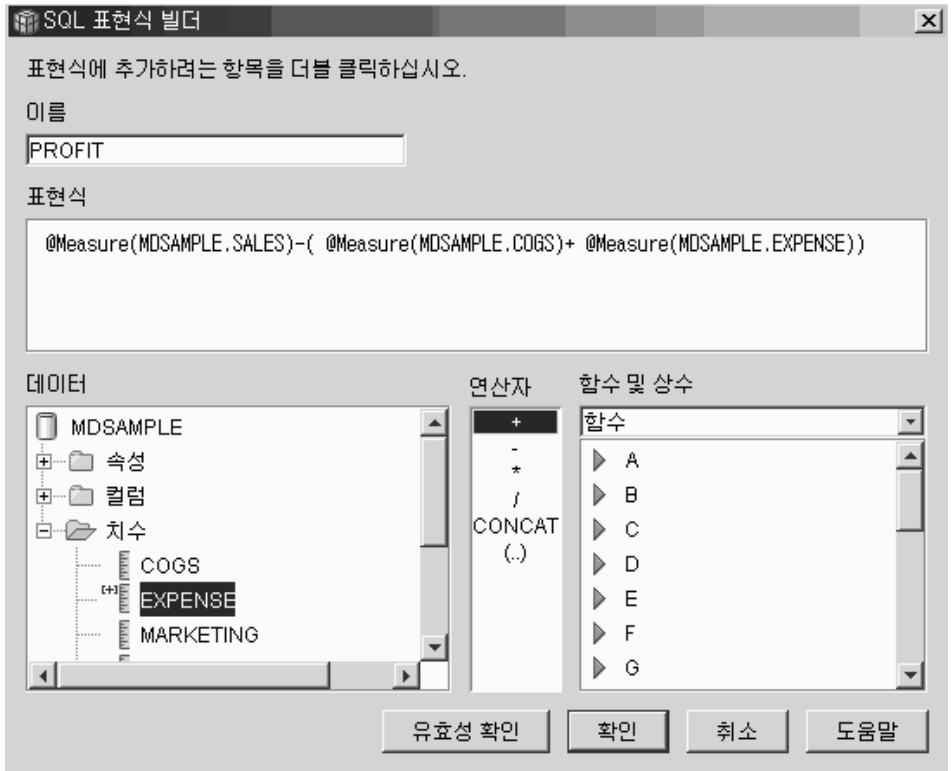


그림 7. SQL 표현식 빌더 창에서 profit 표현식을 완료하십시오.

- d. 확인을 눌러 이익 치수를 작성하고 SQL 표현식 빌더 창을 닫으십시오.
  - e. 집계 페이지에서, 이익 치수 집계를 누르고 SUM 기능을 선택하십시오. 이익 치수가 완료됩니다.
3. 다음과 같이 이익 마진 치수를 작성하십시오.
    - a. 치수 페이지에서, 계산된 치수 작성을 누르십시오. SQL 표현식 빌더 창이 열립니다.
    - b. SQL 표현식 빌더 창에서, 이름 필드에 PROFIT MARGIN을 입력하십시오.
    - c. 이익 마진 표현식을 작성하려면, 치수 폴더를 데이터 목록에서 펼친 후 다음 단계를 완료하십시오.
      - 데이터 목록에서 **PROFIT** 치수를 더블 클릭하여 이를 표현식에 추가하십시오.
      - 연산자 목록에서 / 연산자를 더블 클릭하십시오.

- 데이터 목록에서 **SALES** 치수를 더블 클릭하십시오.
- 표현식 필드에 입력하여 괄호 안의 전체 표현식을 닫으십시오.
- 표현식의 끝에 커서를 위치시키고 연산자 목록에서 \* 연산자를 더블 클릭하십시오.
- 표현식 필드에서 표현식의 끝에 100을 입력하십시오.

그림 8에서는 SQL 표현식 빌더 창에서 작성할 수 있는 이익 마진 표현식을 표시합니다.

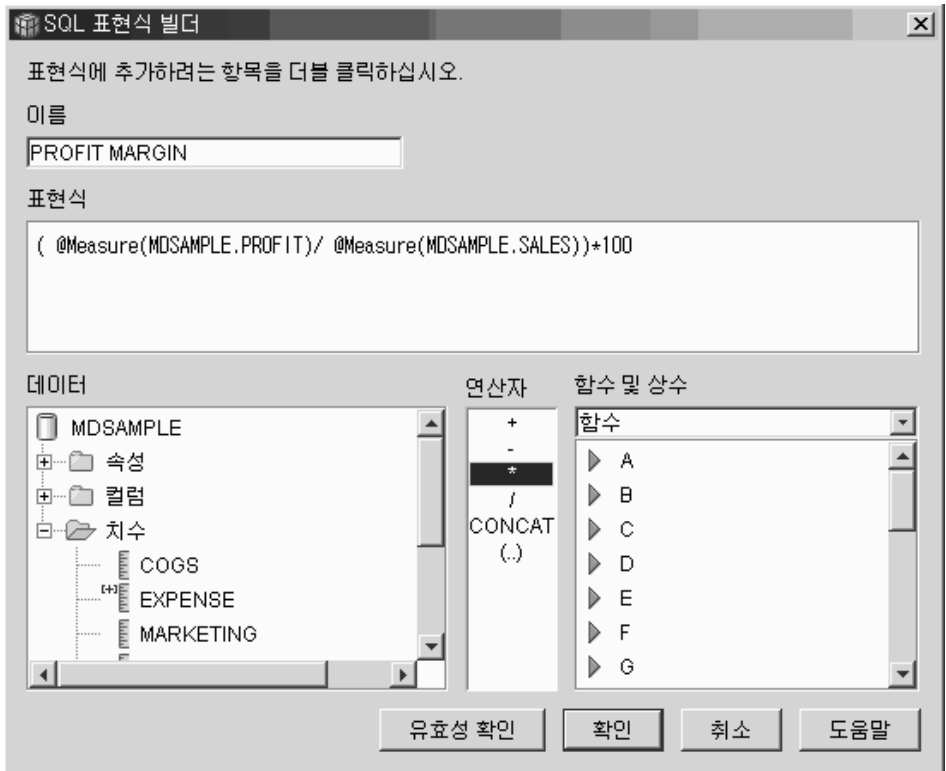


그림 8. SQL 표현식 빌더 창에서 profit margin 표현식을 완료하십시오

- d. 확인을 눌러 이익 마진 치수를 작성하고 SQL 표현식 빌더 창을 닫으십시오.

집계 페이지에서, OLAP 센터가 혼합 치수에 대한 디폴트값으로 집계 함수를 NONE으로 설정하므로 집계 함수를 변경할 필요가 없습니다.

4. 확인을 눌러 사실 등록 정보 창을 닫고 사실 오브젝트에 추가한 두 개의 새 치수를 저장하십시오.

DBA가 이러한 두 치수를 작성한 후에, 이 중요한 치수에 대해 추가 분석이 완료 될 수 있습니다.



---

## 제 4 장 인터넷 주문 숫자 카운팅

소매 회사는 몇 년 전에 인터넷 판매를 추가하여 사업을 확장했습니다. 이제 회사는 인터넷 판매의 효과를 분석하고자 합니다. 회사가 요구하는 첫 번째 계산 중 하나는 인터넷을 통해 완료된 주문의 수입입니다.

---

### 시나리오 세부사항

회사의 데이터베이스에는 ORDER\_ID, PRODUCT\_ID, QUANTITY 및 TIME\_ID 컬럼이 있는 인터넷 주문의 사실 테이블이 있습니다. PRODUCT\_ID 컬럼에는 해당 주문에서 판매된 각 제품이 있으며, QUANTITY 컬럼은 주문에서 구매된 제품의 수량을 저장합니다. 하나 이상의 제품 주문에는 주문에서 판매된 고유한 제품의 수와 같은 수의 행 항목이 있습니다. 예를 들어, 표 6에서는 세 개의 주문을 표시하는데, 여기서 주문 1에는 세 개의 제품(Product As, Product O 및 Product G가 하나씩)이 있습니다.

표 6. 사실 테이블 콘텐츠의 부분

ORDER_ID	PRODUCT_ID	QUANTITY
1	A	3
1	O	1
1	G	1
2	L	1
2	Q	2
3	P	5

DBA는 주문 계수 치수를 작성하여 ORDER\_ID 컬럼의 각 고유 항목을 카운트할 수 있습니다. 주문 계수 치수는 DISTINCT 키워드를 사용하여 SQL 표현식에서 정의되며 집계에 대한 COUNT 함수는 모든 차원에 해당됩니다. 치수의 SQL 표현식은 구별 순서 목록을 작성하며, 이는 집계 작업 동안 카운트됩니다. 치수는 어떠한 합산 작업도 포함되지 않으므로 이를 비추가 치수라고 합니다.

비치가 치수는 카운트하려는 문자 데이터 또는 기타 데이터가 있을 때에도 유용합니다. 예를 들어, 비치가 치수를 사용하여 제품을 배송하려는 곳의 우편 번호의 수를 카운트할 수 있습니다.

이 예에서, DBA는 ORDER\_ID 컬럼으로 직접 맵핑되는 주문 ID를 정의하려 했습니다. 그러나, 동일한 방법으로 ORDER\_ID 컬럼을 사용하도록 선택할 수 있습니다. 디폴트 집계는 SQL 표현식에 컬럼이 사용되는지 치수가 사용되는지에 따라 다릅니다. 그러나 두 경우 모두 『치수 작성 단계』에서 설명한 대로 디폴트 집계를 COUNT 함수로 변경해야 합니다.

---

## 치수 작성 단계

다음 단계는 OLAP 센터 사실 등록 정보 창을 사용하여 기존 사실 오브젝트의 주문 계수 치수를 작성하는 방법을 설명합니다.

1. 사실 등록 정보 창을 열려면, OLAP 센터 오브젝트 트리에서 사실 오브젝트를 마우스 오른쪽 단추로 누르고 치수 편집을 누르십시오. 사실 등록 정보 창이 열립니다.
2. 치수 페이지에서, 계산된 치수 작성 누름 단추를 누르십시오. SQL 표현식 빌더 창이 열립니다.
3. SQL 표현식 빌더 창에서, 이름 필드에 ORDER COUNT를 입력하십시오.
4. 주문 계수 표현식을 작성하려면, 치수 폴더를 데이터 목록에서 펼친 후 다음 단계를 완료하십시오.
  - 함수 및 상수 필드에서, 기타를 선택하십시오. 기타 함수 및 상수 목록에서, **DISTINCT** 키워드를 더블 클릭하십시오.
  - 데이터 목록에서 **ORDER ID** 치수를 더블 클릭하십시오.

23 페이지의 그림 9에서는 SQL 표현식 빌더 창에서 작성할 수 있는 주문 계수 표현식을 표시합니다.

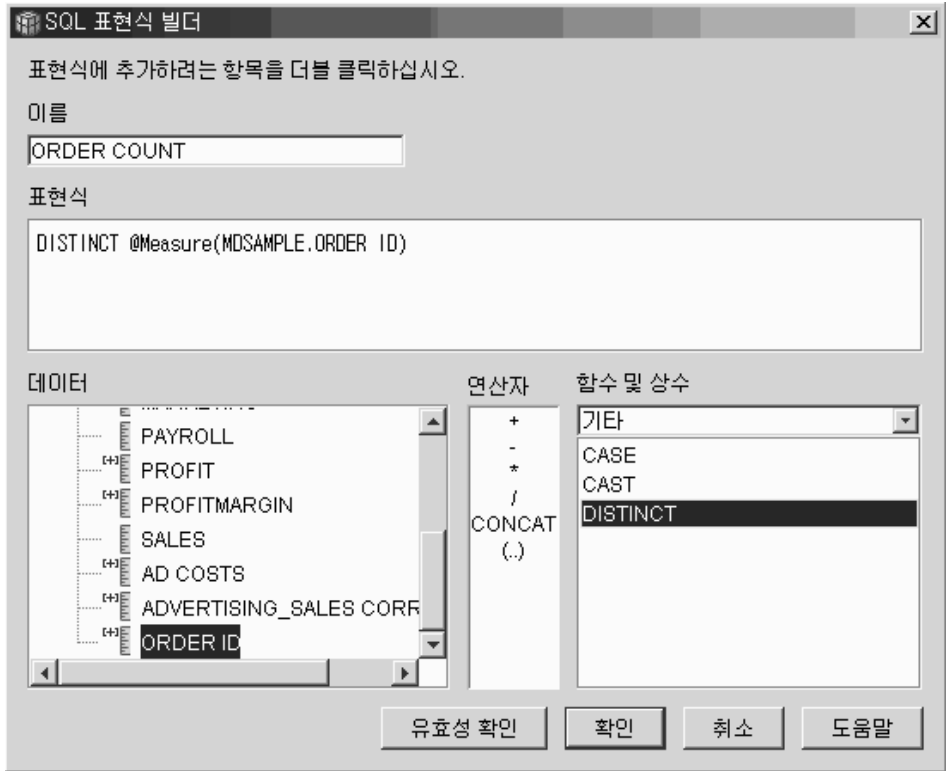


그림 9. SQL 표현식 빌더 창에서 order count 표현식을 완료하십시오

5. 확인을 눌러 SQL 표현식 빌더 창을 닫으십시오.
6. 집계 페이지에서, 주문 계수 치수에 대한 집계를 누르고 **COUNT** 함수를 선택하십시오.
7. 확인을 눌러 사실 오브젝트에 변경사항을 저장하고 사실 등록 정보 창을 닫으십시오.

이제 사용자는 구별 주문 ID 행 값의 수를 카운트하는 치수가 있습니다. 기타 치수와 함께 이 치수를 결합하여 사용자의 데이터를 보다 유용하게 분석할 수 있습니다.





---

## 제 5 장 판매액 수치 순위

사무실 공급 상점 체인은 지난 몇 년에 걸쳐 급속하게 확장되었습니다. 비즈니스 실무자들은 성과가 가장 낮은 상점 일부를 닫아 비용을 줄이고 이익을 늘리는 것에 대해 고려 중입니다. 상점의 판매액 기록은 상점 폐쇄를 결정하는 중요한 인수입니다. 분석자는 판매 수치의 순위를 정하고 분석을 완료하기 위해 차원에 걸쳐 드릴 다운할 수 있기를 원합니다.

---

### 시나리오 세부사항

사무실 공급 상점의 데이터베이스에는 기타 컬럼 및 판매액 컬럼이 있는 사실 테이블이 있습니다. 그 데이터베이스에는 몇몇 차원 테이블도 있습니다. DBA는 RANK 함수를 사용하는 판매액 순위 치수를 작성할 수 있으며, 이 함수는 DB2 Universal Database(DB2 UDB)가 제공한 OLAP 함수입니다.

DB2 Cube Views는 DB2 UDB가 제공하는 다음의 OLAP 함수를 지원합니다.

#### RANK

행을 순서화하고 각 행에 순위를 지정합니다. 순위는 바로 앞 행의 숫자에 1을 더해 정의되며 이는 순서와는 구별됩니다. 중복 행 값을 가지고 있어서 두 개 이상의 행에 대한 상대적인 순서를 판별할 수 없을 경우, 동일한 순위 번호가 지정됩니다. 중복 행 값이 있을 경우 순위 지정 결과 숫자에 갭이 있습니다. 27 페이지의 표 7에서는 일련의 샘플 행 값에 대한 RANK 함수로부터의 순위 지정 결과의 예를 표시합니다.

RANK 함수의 일반적인 구문은 다음과 같습니다.

```
RANK ( ) OVER (ORDER BY sort-key-expression expression-order)
```

여기서 *sort-key-expression*은 순위가 매겨지는 데이터 세트이며 *expression-order*는 키워드인 **ASC** 또는 **DESC**입니다. 이 키워드는 오름차순 또는 내림차순으로 *sort-key-expression*의 값을 순서화합니다. DB2 Cube Views는 *sort-key-expression*이 컬럼이나 속성이 아닌 기존 치수이

길 원합니다. 또한, DB2 Cube Views는 DB2 UDB가 이 함수와 함께 제공하는 PARTITION BY 절을 지원하지 않습니다. RANK 함수에 대한 자세한 정보는 DB2 UDB 정보 센터를 참조하십시오.

## DENSERANK

행을 순서화하고 각 행에 순위를 지정합니다. 행의 순위는 바로 앞 행의 숫자에 1을 더한 것으로 정의됩니다. 따라서 순위 지정 결과는 순차적이며 순위 숫자에 갭이 없습니다. 27 페이지의 표 7에서는 일련의 샘플 행 값에 대한 DENSERANK 함수로부터의 순위 지정 결과의 예를 표시합니다.

DENSERANK 함수의 일반적인 구문은 다음과 같습니다.

```
DENSERANK ( ) OVER (ORDER BY sort-key-expression expression-order)
```

여기서 *sort-key-expression*은 순위가 매겨지는 데이터 세트이며 *expression-order*는 키워드인 **ASC** 또는 **DESC**입니다. 이 키워드는 오름차순 또는 내림차순으로 *sort-key-expression*의 값을 순서화합니다. DB2 Cube Views는 *sort-key-expression*이 컬럼이나 속성이 아닌 기존 치수이길 원합니다. 또한, DB2 Cube Views는 DB2 UDB가 이 함수와 함께 제공하는 PARTITION BY 절을 지원하지 않습니다. DENSERANK 함수에 대한 자세한 정보는 DB2 UDB 정보 센터를 참조하십시오.

## ROWNUMBER

맨 첫 행을 1로 시작하여 순서화함으로써 행의 순차 행 번호를 계산합니다. ORDER BY 절이 지정되지 않은 경우, 행에 번호가 임의의 순서로 지정됩니다.

ROWNUMBER 함수의 일반적인 구문은 다음과 같습니다.

```
ROWNUMBER ( ) OVER  
([ORDER BY sort-key-expression expression-order])
```

여기서 *sort-key-expression*은 순위가 매겨지는 데이터 세트이며 *expression-order*는 키워드인 **ASC** 또는 **DESC**입니다. 이 키워드는 오름차순 또는 내림차순으로 *sort-key-expression*의 값을 순서화합니다. DB2 Cube Views는 컬럼이나 속성이 아닌 기존 치수가 이 함수의 데이터 소스로 사용되기를 원합니다. 또한, DB2 Cube Views는 DB2 UDB가 이

함수와 함께 제공하는 PARTITION BY 절을 지원하지 않습니다.  
ROWNUMBER 함수에 대한 자세한 정보는 DB2 UDB 정보 센터를 참조하십시오.

이러한 OLAP 함수는 SQL 표현식 빌더 함수 및 상수 목록에 나열되지 않습니다.

표 7. RANK 및 DENSERANK 함수를 사용하는 일련의 값 샘플에 대한 순위 결과

행 값	순서화	RANK 함수로부터의 순위 결과	DENSERANK 함수로부터의 순위 결과
100	1	1	1
35	2	2	2
23	3	3	3
8	4	4	4
8	4	4	5
6	5	6	6

## 치수 작성 단계

다음 단계는 OLAP 센터 사실 등록 정보 창을 사용하여 기존 사실 오브젝트의 판매액 순위 치수를 작성하는 방법을 설명합니다.

1. 사실 등록 정보 창을 열려면, OLAP 센터 오브젝트 트리에서 사실 오브젝트를 마우스 오른쪽 단추로 누르고 치수 편집을 누르십시오. 사실 등록 정보 창이 열립니다.
2. 치수 페이지에서, 계산된 치수 작성을 눌러 판매액 순위 치수를 작성하십시오. SQL 표현식 빌더 창이 열립니다.
3. SQL 표현식 빌더 창에서, 이름 필드에 SALES RANK를 입력하십시오.
4. 판매액 순위 표현식을 작성하려면, 다음 단계를 완료하십시오.
  - 표현식 필드에 다음과 같은 함수 구문을 입력하십시오.  
RANK ( ) OVER (ORDER BY measure DESC)
  - 데이터 목록에서 치수 폴더를 펼치십시오.
  - 표현식 필드의 함수 구문에 치수라는 단어를 강조표시한 후, SALES 치수를 더블 클릭하여 표현식에 SALES 치수를 추가하십시오.

최종 표현식이 그림 10에 표시됩니다.

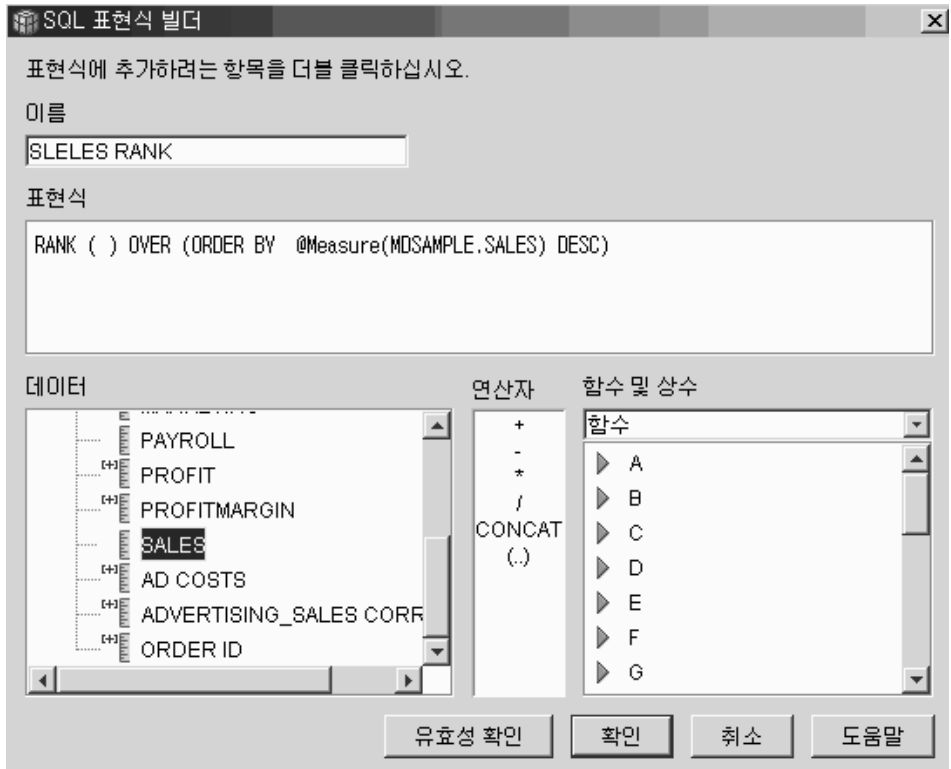


그림 10. SQL 표현식 빌더 창에서 Sales Rank 표현식을 완료하십시오.

5. 유효성 확인을 눌러 표현식이 유효한지 확인하십시오. 확인을 눌러 SQL 표현식 빌더 창을 닫으십시오.

집계 페이지에서 디폴트 집계, 없음을 변경할 필요는 없습니다. 데이터 소스가 숫자이고 기존 치수만을 참조하므로, 없음 옵션은 판매액 순위 치수에 대한 기본값입니다.

치수의 표현식에 RANK 함수를 사용하여 판매액 컬럼을 내림차순으로 순서화할 경우, 분석자는 다른 차원에 걸쳐 드릴 다운하여 특정 제품 라인에 대해 작년에 가장 저조한 판매액을 기록한 상점을 판별할 수 있거나 데이터베이스에 저장된 다른 차원의 데이터와 관련된 상점을 판별할 수 있습니다.

---

## 제 6 장 사실 테이블에 저장된 시간 데이터를 사용하여 시간 차원 작성

소매 사업인 XYZ Retail은 DB2 Cube Views에서 판매액 트랜잭션 데이터를 모델링하여 데이터를 보다 효과적으로 분석할 수 있습니다. 그러나, 데이터의 트랜잭션 특성으로 인해, 사용 가능한 유일한 시간 정보는 각 트랜잭션과 연관된 날짜입니다. 시간 차원에서 모델화되는 시간 정보는 분기별 판매 경향 분석 및 각 주별 평균 재고 값 계산과 같은 여러 일반 계산에 컨텍스트를 추가하는 데 사용됩니다.

특정일에 트랜잭션이 없는 경우 데이터에 결함이 생겨 정확하게 집계하고 표시하는 데 문제가 발생할 수 있으므로 많은 DBA는 시간 데이터를 트랜잭션 시간소인 또는 날짜로 저장하지 않습니다. 일반적으로, 시간 테이블에서 시간 데이터를 모델화하는 것이 좋습니다. 그러나 XYZ Retail의 DBA는 매일 최소 하나의 트랜잭션이 있을 것을 확신하고 데이터에 대한 현재 구조를 보존하기로 합니다.

---

### 시나리오 세부사항

XYZ Retail에는 판매액, 비용, 판매된 수량 및 날짜를 포함하는 각 트랜잭션에 대한 측정 가능 정보가 있는 사실 테이블이 있습니다. 데이터베이스에는 영역 차원 테이블 및 제품 차원 테이블도 있습니다. 문제점은 시간 데이터가 개별 차원 테이블에 저장되어 있지 않고 사실 테이블에 포함되어 있다는 것입니다. DBA는 사실 오브젝트의 날짜 데이터에 근거하여 차원 오브젝트를 작성해야 합니다.

사실 테이블의 날짜 데이터 단일 컬럼에 근거하여 시간 차원을 작성하려면 다음과 같은 두 가지 고유한 요구사항을 수행해야 합니다.

- 유효한 큐브 모델의 모든 차원 오브젝트는 사실 오브젝트로 조인되어야 하며 시간 차원 오브젝트 및 사실 오브젝트는 동일한 사실 테이블에 근거를 두어야 하므로, 사실 테이블을 자체적으로 조인하는 자체 조인을 사용하여 시간 차원 오브젝트를 사실 오브젝트로 조인해야 합니다.
- DBA는 주, 월, 분기 및 년과 같은 의미있는 레벨로 날짜 데이터를 집계하는 계산된 속성을 빌드해야 합니다.

자체 조인은 테이블을 자체적으로 조인하는 조인의 한 유형이며, 이 경우의 테이블은 사실 테이블입니다. 자체 조인은 하나 이상의 컬럼을 조인해야 하며 이로 인해 사실 테이블의 모든 행을 고유하게 식별할 수 있습니다. 기본 키를 선택하는 것이 가장 좋습니다. 그러나, 기본 키가 정의되어 있지 않은 경우, 사실 테이블을 차원 테이블과 조인하는 데 사용할 컬럼 세트를 선택하십시오. 큐브 모델을 최적화하려면, 기본 키가 정의되어 있어야 합니다. 조인 카디널리티(cardinality)는 1:1이어야 하며 조인 유형은 내부이어야 합니다.

그림 11에서는 사실 오브젝트, 사실 테이블에 근거한 차원 및 사실-차원 조인이 동일한 사실 테이블로 맵핑되는 방법을 표시합니다.

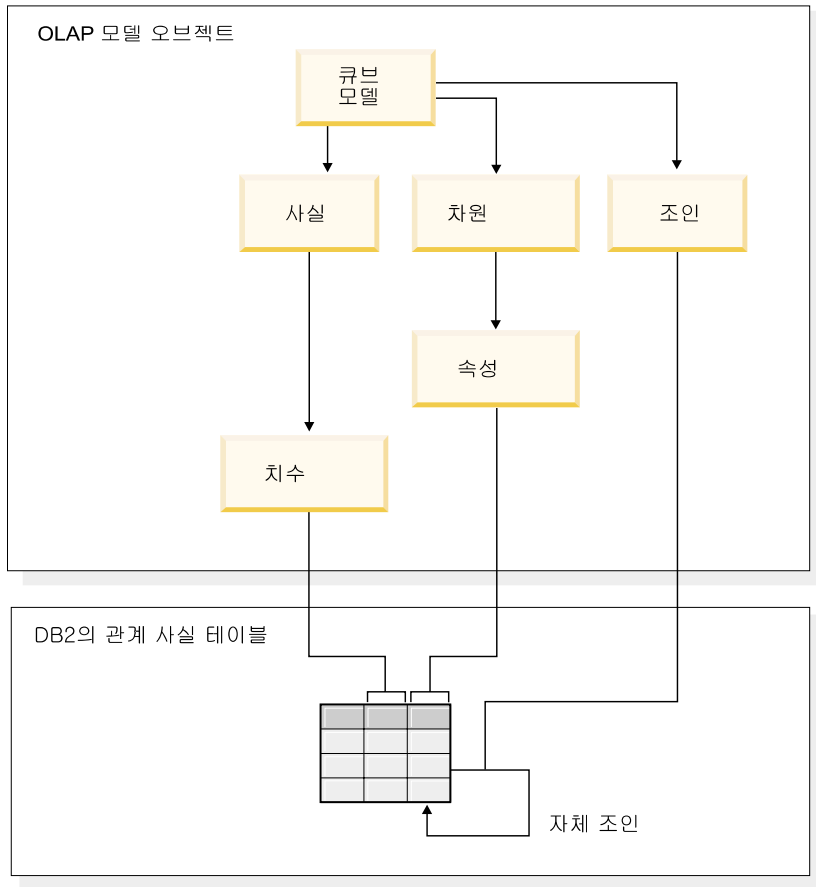


그림 11. 자체 조인이 테이블을 자체적으로 조인하는 방법

---

## 속성 및 차원 작성 단계

다음 단계는 사용자가 OLAP 센터 차원 마법사를 사용하여 사실 테이블에 근거하여 계산된 속성 및 시간 차원을 작성하는 방법을 설명합니다.

1. 차원 마법사를 열려면, OLAP 센터 오브젝트 트리의 큐브 모델을 마우스 오른쪽 단추로 누른 후 **차원 작성**을 누르십시오. 차원 마법사가 열립니다.
2. 이름 페이지에서, 이름 필드에 Time을 입력하십시오. 선택적으로 비즈니스 이름을 변경하고 주석을 입력할 수도 있습니다. 다음을 누르십시오.
3. 큐브 모델의 사실 테이블을 선택하십시오. 다음을 누르십시오. 차원에 하나의 테이블만 있으므로 차원 조인을 지정할 필요는 없습니다. 차원 조인 페이지에서, 다음을 누르십시오.
4. 차원 속성 페이지에서, 시간소인 컬럼을 선택하십시오.
5. 선택사항: 시간소인 데이터를 보다 큰 청크(예:월, 분기 및 년)로 집계하는 추가로 계산된 속성을 작성하십시오. 계산된 속성을 작성하려면, 계산된 속성 작성 누름 단추를 눌러 SQL 표현식 빌더를 열고, 소스 시간소인 컬럼을 월, 분기 및 년으로 계산하는 각 속성의 표현식을 정의하십시오. 계산된 각각의 속성을 정의한 후, 유효성 확인 누름 단추를 눌러 표현식이 유효한지 확인한 다음, 확인을 눌러 SQL 표현식 빌더를 닫고 차원 마법사로 리턴하십시오. 선택한 후 다음을 누르고 원하는 속성 모두를 작성하십시오.
6. 차원 입력 페이지에서, 시간을 선택하십시오. 다음을 누르십시오.
7. 사실 차원 조인 페이지에서, 조인 작성을 누르십시오. 열린 조인 마법사에서, 자체 조인을 작성하십시오. 이름을 입력하고 다음을 누르십시오. 기본 키와 같은 사실 테이블의 모든 행을 왼쪽 및 오른쪽 속성 둘 다에 대해 고유하게 정의하는 컬럼 또는 컬럼 세트를 선택하십시오. 한 번에 한 쌍을 선택하고 추가를 눌러 조인에 속성 쌍을 추가하십시오. 내부 조인 유형 및 1:1 카디널리티(cardinality)를 선택하십시오. 필요한 속성 쌍을 추가한 후, 완료를 누르십시오. 조인 마법사가 닫힙니다.
8. 사실 차원 조인 페이지에서, 완료를 누르십시오.

큐브 모델에서 정의한 시간 차원을 사용하여, XYZ Retail은 해당 데이터 분석에 의미있는 새 레벨을 추가할 수 있습니다. 재고를 포함한 시간 관련 분석을 수행할 수 있습니다.



---

## 주의사항

이 정보는 미국에서 제공되는 제품 및 서비스용으로 작성된 것입니다.

IBM은 다른 국가에서는 이 자료에 기술된 제품, 서비스 또는 기능을 제공하지 않을 수도 있습니다. 현재 사용할 수 있는 제품 및 서비스에 대한 정보는 한국 IBM 담당자에게 문의하십시오. 이 책에서는 IBM 제품, 프로그램 또는 서비스를 언급하는 것이 해당 IBM 제품, 프로그램 또는 서비스만을 사용할 수 있다는 것을 의미하지 않습니다. IBM의 지적 재산권을 침해하지 않는 한, 기능상 동등한 제품, 프로그램 또는 서비스를 대신 사용할 수 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 특허에 대한 라이선스까지 부여한 것은 아닙니다. 라이선스에 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 라이선스 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, 변경된 사항은 최신판에 통합됩니다. IBM은 이 책에서 설명한 제품 및(또는) 프로그램을 사전 통지없이 언제든지 개선 및(또는) 변경할 수 있습니다.

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

(1) 독립적으로 작성된 프로그램과 기타 프로그램(본 프로그램 포함)간의 정보 교환 및 교환된 정보의 상호 이용을 목적으로 정보를 원하는 프로그램 라이선스 사용자는 다음 주소로 문의하십시오.

135-270

서울특별시 강남구 도곡동 467-12, 군인공제회관빌딩

한국 아이.비.엠 주식회사

고객만족센터

이러한 정보는 해당 조항 및 조건(예를 들어, 사용료 지불 등)에 따라 사용할 수 있습니다.

이 정보에 기술된 라이선스가 있는 프로그램 및 이 프로그램에 대해 사용 가능한 모든 라이선스가 있는 자료는 IBM이 IBM 기본 계약, IBM 프로그램 라이선스 계약(IPLA) 또는 이와 동등한 계약에 따라 제공한 것입니다.

비IBM 제품에 관한 정보는 해당 제품의 공급업체, 공개 자료 또는 기타 범용 소스로부터 얻은 것입니다. IBM에서는 이러한 비IBM 제품을 테스트하지 않았으므로, 이들 제품과 관련된 성능의 정확성, 호환성 또는 기타 주장에 대해서는 확신할 수 없습니다. 비IBM 제품의 성능에 대한 의문사항은 해당 제품의 공급업체에 문의하십시오.

이 정보에는 일상의 비즈니스 운영에서 사용되는 자료 및 보고서에 대한 예제가 들어 있습니다. 이들 예제에는 개념을 가능한 완벽하게 설명하기 위해 개인, 회사,

상표 및 제품의 이름이 사용될 수 있습니다. 이들 이름은 모두 가공의 것이며 실제 기업의 이름 및 주소와 유사하더라도 이는 전적으로 우연입니다.

본 문서에 포함된 모든 성능 데이터는 제한된 환경에서 산출된 것입니다. 따라서 다른 운영 환경에서 얻어진 결과는 상당히 다를 수 있습니다. 일부 성능은 개발 레벨 상태의 시스템에서 측정되었을 수 있으므로 이러한 측정치가 일반적으로 사용되고 있는 시스템에서도 동일하게 나타날 것이라고는 보증할 수 없습니다. 또한, 일부 성능은 추정치일 수도 있으므로 실제 결과는 다를 수 있습니다. 이 문서의 사용자는 해당 데이터를 사용자의 특정 환경에서 검증해야 합니다.

비IBM 제품에 관한 정보는 해당 제품의 공급업체, 공개 자료 또는 기타 범용 소스로부터 얻은 것입니다. IBM에서는 이러한 비IBM 제품을 테스트하지 않았으므로, 이들 제품과 관련된 성능의 정확성, 호환성 또는 기타 주장에 대해서는 확신할 수 없습니다. 비IBM 제품의 성능에 대한 의문사항은 해당 제품의 공급업체에 문의하십시오.

이 정보를 소프트카피로 보는 경우에는 사진과 컬러 삽화가 제대로 나타나지 않을 수도 있습니다.

---

## 상표

다음 용어는 미국 또는 기타 국가에서 사용되는 IBM Corporation의 상표입니다.

DB2  
DB2 Universal Database  
IBM  
Office Connect  
Redbooks

다음 용어는 다음 회사의 상표 또는 등록상표입니다.

Microsoft, Windows, Windows NT, Windows 2000, Windows XP 및 Microsoft Excel은 Microsoft Corporation의 등록상표입니다.

Java 또는 모든 자바 기반 상표 및 로고와 Solaris는 미국 및 기타 국가에서 사용되는 Sun Microsystems, Inc.의 상표입니다.

UNIX은 미국 또는 기타 국가에서 사용되는 등록상표이며 X/Open Company Limited의 독점으로 라이선스를 가지고 있습니다.

Linux는 Linus Torvalds의 등록상표입니다. Red Hat 및 모든 Red Hat 기반 상표 및 로고는 미국 및 기타 국가에서 사용되는 Red Hat, Inc.의 상표 또는 등록상표입니다.

기타 회사, 제품 또는 서비스 이름은 해당되는 상표 또는 서비스표입니다.









프로그램 번호: 5724-E15

SA30-2121-00

