

SAP BusinessObjects Intelligence Suite  
문서 버전: 4.1 Support Package 7 – 2015/12/11

## **Web Intelligence**에서 함수, 수식 및 계산 사용

# 내용

<b>1</b>	문서 이력: <b>Web Intelligence</b> 함수, 수식 및 계산.....	<b>5</b>
<b>2</b>	가이드 정보.....	<b>6</b>
<b>3</b>	표준 및 사용자 지정 계산 사용.....	<b>7</b>
3.1	표준 계산식.....	7
3.2	수식을 사용하여 사용자 지정 계산식 작성.....	7
	변수를 사용하여 수식 단순화.....	8
3.3	함수 사용.....	8
	셀에 함수 포함.....	8
	함수 구문.....	9
	함수 예.....	9
	함수 및 수식 연산자.....	14
<b>4</b>	계산 컨텍스트 이해.....	<b>16</b>
4.1	입력 컨텍스트.....	16
4.2	출력 컨텍스트.....	17
4.3	기본 계산 컨텍스트.....	19
	세로 테이블의 기본 컨텍스트.....	20
	가로 테이블의 기본 컨텍스트.....	21
	크로ست랩 테이블의 기본 컨텍스트.....	21
	섹션의 기본 컨텍스트.....	22
	나누기의 기본 컨텍스트.....	23
4.4	확장 구문을 사용하여 기본 계산 컨텍스트 수정.....	24
	확장 구문 연산자.....	24
<b>5</b>	스마트 계수를 사용하여 값 계산.....	<b>28</b>
5.1	그룹화 집합 및 스마트 계수.....	28
	그룹화 집합 관리.....	29
5.2	스마트 계수 및 분석 범위.....	29
5.3	스마트 계수 및 SQL.....	30
	그룹화 집합 및 UNION 연산자.....	30
5.4	스마트 계수 및 수식.....	32
	수식이 포함된 스마트 계수 및 차원.....	32
	수식의 스마트 계수.....	32
5.5	스마트 계수 및 필터.....	33
	스마트 측정 및 필터에 관한 제한.....	33

	차원의 스마트 계수 및 필터.....	33
	스마트 계수 필터링.....	34
	스마트 계수 및 드릴 필터.....	35
	스마트 계수 및 중첩된 OR 필터.....	35
<b>6</b>	<b>함수, 연산자 및 키워드.....</b>	<b>36</b>
6.1	함수.....	36
	사용자 지정 서식.....	36
	집계 함수.....	39
	문자 함수.....	78
	날짜 및 시간 함수.....	100
	데이터 공급자 함수.....	117
	문서 함수.....	134
	논리 함수.....	144
	숫자 함수.....	154
	집합 함수.....	178
	기타 함수.....	199
6.2	함수 및 수식 연산자.....	220
	수학 연산자.....	220
	조건부 연산자.....	221
	논리 연산자.....	221
	함수별 연산자.....	225
	확장 구문 연산자.....	233
	집합 연산자.....	236
6.3	확장 구문 키워드.....	237
	Block 키워드.....	237
	Body 키워드.....	239
	Break 키워드.....	239
	Report 키워드.....	240
	Section 키워드.....	241
6.4	값 반올림 및 자르기.....	242
6.5	계층구조에서 멤버 및 멤버 집합 참조.....	243
<b>7</b>	<b>수식 문제 해결.....</b>	<b>245</b>
7.1	자동 수식 재작성 메커니즘.....	245
7.2	수식 오류 및 정보 메시지.....	246
	#COMPUTATION.....	246
	#CONTEXT.....	246
	#DATASYNC.....	247
	#DIV/0.....	247
	#ERROR.....	247



	#EXTERNAL. . . . .	248
	#INCOMPATIBLE. . . . .	248
	#MIX. . . . .	248
	#MULTIVALUE. . . . .	248
	#N/A. . . . .	248
	#OVERFLOW. . . . .	249
	#PARTIALRESULT. . . . .	249
	#RANK. . . . .	249
	#RECURSIVE. . . . .	249
	#REFRESH. . . . .	250
	#SECURITY. . . . .	250
	#SYNTAX. . . . .	250
	#TOREFRESH. . . . .	250
	#UNAVAILABLE. . . . .	251
<b>8</b>	<b>함수를 사용하여 값 비교. . . . .</b>	<b>252</b>
8.1	Previous 함수를 사용하여 값 비교. . . . .	252
8.2	RelativeValue 함수를 사용하여 값 비교. . . . .	252
	분리 차원 및 RelativeValue 함수. . . . .	253
	분리 차원 및 섹션. . . . .	255
	차원 분리 순서. . . . .	256
	분리 차원 및 정렬. . . . .	258
	크로스탭에서 RelativeValue 사용. . . . .	259

# 1 문서 이력: Web Intelligence 함수, 수식 및 계산

다음 표에 중요한 문서 변경 사항이 간략하게 나와 있습니다.

버전	날짜	설명
SAP BusinessObjects Web Intelligence 4.1 SP07	2015년 11월	다음 단원이 가이드에 추가되었습니다. <ul style="list-style-type: none"><li>주(week)에 대한 사용자 지정 서식을 생성하는 문자열 <a href="#">사용자 지정 서식 [페이지 36]</a></li><li>날짜/시간 값에 시간대 표시 가능 <a href="#">사용자 지정 서식 [페이지 36]</a></li><li>Concatenation() 함수 동작 업데이트됨 <a href="#">Concatenation [페이지 80]</a></li><li>RelativeDate() 함수 동작 업데이트됨 <a href="#">RelativeDate [페이지 111]</a></li><li>4.1 SP3부터 제공되는 자동 수식 재작성 메커니즘에 대한 내용이 기록되었습니다. <a href="#">자동 수식 재작성 메커니즘 [페이지 245]</a></li></ul>
SAP BusinessObjects Web Intelligence 4.1 SP05	2014년 11월	format_string 구문의 예가 가이드에 추가되었습니다. 이 예는 다음에서 볼 수 있습니다: <a href="#">FormatDate 함수 관련 Format_string 예제 [페이지 83]</a> .
SAP BusinessObjects Web Intelligence 4.1	2013년 5월	아랍어 로캘의 RTL(Right-to-Left: 오른쪽에서 왼쪽) 지원이 포함되었으며, 이는 특히 RTL 로캘의 다음 문자 기능에 있어 영향을 미칩니다. <ul style="list-style-type: none"><li>Left</li><li>LeftPad</li><li>LeftTrim</li><li>Right</li><li>RightPad</li><li>RightTrim</li></ul>
SAP BusinessObjects Web Intelligence 4.0 기능 팩 3	2012년 2월	기능 팩 3
SAP BusinessObjects Web Intelligence 4.0 SP02	2011년 6월	지원 패키지 2. Web Intelligence로 다시 리브랜딩됨
SAP BusinessObjects Interactive Analysis 4.0 SP01	2011년 3월	지원 패키지 1
SAP BusinessObjects Interactive Analysis 4.0	2010년 11월	처음으로 Interactive Analysis로 리브랜딩. 이 문서의 첫 번째 릴리스

---

## 2 가이드 정보

*Web Intelligence*에서 함수, 수식 및 계산식 사용 가이드는 데이터 분석을 수행할 때 사용할 수 있는 고급 계산 기능에 대한 자세한 정보와 사용 가능한 함수 및 연산자에 대한 구문 참조를 제공합니다.

## 3 표준 및 사용자 지정 계산 사용

표준 계산 함수를 사용하여 데이터에 대한 빠른 계산식을 만들 수 있습니다.

표준 계산식만으로 충분하지 않으면 수식 언어를 사용하여 사용자 지정 계산식을 만들 수도 있습니다.

### 3.1 표준 계산식

표준 계산 함수를 사용하여 데이터에 대한 빠른 계산식을 만들 수 있습니다.

다음 표준 계산식을 사용할 수 있습니다.

계산식	설명
Sum	선택한 데이터의 합계를 계산합니다.
Count	계수 개체의 행을 모두 계산하거나 차원 또는 설명 개체의 개별 행을 계산합니다.
Average	데이터의 평균을 계산합니다.
Minimum	선택한 데이터의 최소값을 표시합니다.
Maximum	선택한 데이터의 최대값을 표시합니다.
Percentage	선택한 데이터를 전체의 백분율로 표시합니다. 백분율 결과는 테이블의 추가 열이나 행에 표시됩니다.  <b>i 노트</b> 백분율은 선택한 계수를 테이블이나 나누기에 있는 해당 계수의 총 결과와 비교하여 계산됩니다. 한 계수를 다른 계수와 비교한 백분율을 계산하려면 사용자 지정 계산식을 만들어야 합니다.
Default	기본 집계 함수를 표준 계수에 적용하거나 데이터베이스 집계 함수를 스마트 계수에 적용합니다.

표준 계산식을 테이블 열에 적용하면 계산 결과가 바닥글에 표시됩니다. 바닥글은 계산식마다 한 개씩 추가됩니다.

### 3.2 수식을 사용하여 사용자 지정 계산식 작성

사용자 지정 계산식을 사용하면 기본 개체 및 표준 계산식 이외에 추가 계산식을 보고서에 삽입할 수 있습니다.

수식을 작성하여 사용자 지정 계산식을 추가합니다. 수식은 기본 보고서 변수, 함수, 연산자 및 계산 컨텍스트로 구성될 수 있습니다.

사용자 지정 계산식은 보고서 개체, 함수 및 연산자로 구성할 수 있는 수식입니다. 수식에는 사용자가 선택할 경우 명시적으로 지정할 수 있는 계산 컨텍스트가 있습니다.



예

#### 판매당 평균 수익 표시

판매 수익과 판매 개수 개체가 포함된 보고서에 개당 수익을 추가하려고 합니다.  $\frac{[\text{판매 수익}]}{[\text{판매량}]}$  계산식은 판매 수익을 판매량으로 나눠 개당 수익을 계산합니다.

### 3.2.1 변수를 사용하여 수식 단순화

수식이 복잡한 경우 변수를 사용하여 단순화할 수 있습니다.

변수를 사용하면 복잡한 수식을 의미 있는 부분으로 나누어 알아 보기 쉽게 만들고 오류 발생 가능성을 줄이면서 수식을 만들 수 있습니다.

다른 보고서 개체와 동일한 방식으로 이전에 만든 변수를 수식에서 사용할 수 있습니다. 변수는 "Variables" 폴더 아래 수식 편집기에 나타납니다.

이 변수 이름을 수식에 입력하거나 다른 보고서 개체와 마찬가지로 변수를 수식 도구 모음에 끌어 놓을 수 있습니다.

### 3.3 함수 사용

사용자 지정 계산식은 보고서 개체만으로 구성될 수도 있습니다(예:  $\frac{[\text{판매 수익}]}{[\text{판매량}]}$ ). 계산식에는 함수 및 보고서 개체도 포함될 수 있습니다.

함수는 입력값으로 0개 또는 여러 개의 값을 받아들이고 이 값에 따라 출력값을 반환합니다. 예를 들어, Sum 함수는 계수의 모든 값을 더해 결과를 출력합니다. 수식  $\text{Sum}([\text{판매 수익}])$ 은 총 판매 수익을 출력합니다. 이 경우 함수 입력값은 판매 수익 계수이고 출력값은 모든 판매 계수의 합계입니다.

#### 관련 정보

[함수 및 수식 연산자 \[페이지 14\]](#)

[함수 \[페이지 36\]](#)

### 3.3.1 셀에 함수 포함

보고서 셀의 텍스트는 항상 '='로 시작합니다.

문자 텍스트는 따옴표 안에 나타나지만 수식은 따옴표 없이 나타납니다. 예를 들어, 수식  $\text{Average}([\text{수익}])$ 는 셀에서  $=\text{Average}([\text{수익}])$ 로 나타나고, 텍스트 "평균 수익?"은 "평균 수익?"으로 나타납니다.



셀에 텍스트만 사용하거나 '+' 연산자를 사용하여 수식과 텍스트를 함께 사용할 수도 있습니다. 셀에 평균 수익을 표시하고 그 앞에 "평균 수익"이라는 텍스트를 표시하려면 셀 텍스트로 ="평균 수익: " + Average([수익])를 입력합니다.

텍스트 문자열 끝에 공백을 넣어야 텍스트와 값이 붙어서 표시되지 않습니다.

### 3.3.2 함수 구문

함수를 선택하면 [수식 편집기](#)에 함수 구문이 표시됩니다.

함수를 사용하려면 함수의 이름, 함수에 필요한 입력 값의 수 및 이러한 입력 값의 데이터 형식에 대해 알고 있어야 합니다. 함수가 출력하는 데이터의 형식도 알고 있어야 합니다.

예를 들어, Sum 함수는 판매 수익을 표시하는 계수 같은 숫자 개체를 입력으로 사용하여 계수 개체의 값 전체에 대한 합계를 나타내는 숫자 데이터를 출력합니다.

다음은 Abs 함수의 구문입니다.

```
num Abs(number)
```

이 구문은 Abs 함수가 입력 값으로 단일 숫자를 사용하며 출력 값으로 숫자를 반환한다는 것을 보여 줍니다.

### 3.3.3 함수 예

이 항목에서는 수식에 사용되는 함수에 대한 예를 제공합니다.

#### 예

##### UserResponse 함수를 사용하여 프롬프트 입력 표시

보고서에 연도, 분기 및 판매 수익이 나타나 있다고 가정합니다. 표시되지는 않더라도 주 개체도 보고서 데이터에 나타납니다. 사용자가 보고서를 실행하면 프롬프트가 표시되며 사용자는 여기서 주를 선택해야 합니다. 사용자가 선택한 주를 보고서 제목에 표시하려고 합니다. 데이터 공급자 이름이 "eFashion"이고 프롬프트 텍스트가 "Choose a State"일 경우 이 제목의 수식은 다음과 같습니다.

```
"Quarterly Revenues for " + UserResponse([Query 1];"Enter values for State:")
```

사용자가 데이터 공급자를 새로 고칠 때 Illinois를 주로 선택한 경우 보고서는 다음과 같습니다.

### Quarterly Revenue for Illinois

Year	Quarter	Sales revenue
2001	Q1	\$256,454
	Q2	\$241,458
	Q3	\$107,006
	Q4	\$133,306
<b>2001</b>	<b>합계:</b>	<b>\$738,223.80</b>

Year	Quarter	Sales revenue
2002	Q1	\$334,297
	Q2	\$254,722
	Q3	\$230,573
<b>2002</b>	<b>합계:</b>	<b>\$1,150,658.80</b>

Year	Quarter	Sales revenue
2003	Q1	\$255,658
	Q2	\$354,724
	Q3	\$273,186
	Q4	\$250,517
<b>2003</b>	<b>합계:</b>	<b>\$1,134,085.40</b>



예

#### Percentage 함수를 사용하여 백분율 계산

Percentage 함수는 백분율을 계산합니다. 이 함수는 특정 값의 백분율을 해당 컨텍스트를 기준으로 계산합니다. 예를 들어, 다음 표에는 연도 및 분기별 수익이 나와 있습니다. 백분율 열에는 Percentage ([판매 수익]) 수식이 포함 되어 있습니다.

Year	Quarter	Sales revenue	Percentage
2001	Q1	\$2660700	0.07
2001	Q2	\$2279003	0.06
2001	Q3	\$1367841	0.04
2001	Q4	\$1788580	0.05
2002	Q1	\$3326172	0.09
2002	Q2	\$2840651	0.08
2002	Q3	\$2879303	0.08
2002	Q4	\$4186120	0.12
2003	Q1	\$3742989	0.1
2003	Q2	\$4006718	0.11
2003	Q3	\$3953395	0.11
2003	Q4	\$3356041	0.09
		<b>합계:</b>	<b>1</b>

이 경우 함수는 각 수익을 총 수익에 대한 백분율로 계산합니다. 주변 컨텍스트는 총 수익이고, 이 값은 해당 테이블에서 연도 및 분기별로 데이터를 구분하지 않았을 때 의미가 있는 유일한 수익 값입니다.

보고서가 연도별 섹션으로 분할되어 있는 경우 표를 포괄하는 컨텍스트는 섹션의 총 수익이 됩니다.

2001			
Year	Quarter	Sales revenue	Percentage
2001	Q1	\$2660700	0.33
2001	Q2	\$2279003	0.28
2001	Q3	\$1367841	0.17
2001	Q4	\$1788580	0.22
		<b>합계:</b>	<b>1</b>

백분율 셀이 표 바깥쪽에서 섹션 안에 배치되어 있는 경우 컨텍스트는 총 수익이 됩니다. 이 경우 Percentage 함수는 섹션의 총 수익을 전체 총 수익에 대한 백분율로 계산합니다.

2001	0.22
------	------

Year	Quarter	Sales revenue
2001	Q1	\$2660700
2001	Q2	\$2279003
2001	Q3	\$1367841
2001	Q4	\$1788580

2002	0.36
------	------

Year	Quarter	Sales revenue
2002	Q1	\$3326172
2002	Q2	\$2840651
2002	Q3	\$2879303
2002	Q4	\$4186120



예

#### Sum 함수를 사용하여 백분율 계산

Percentage 함수보다 Sum 함수를 사용하면 백분율을 계산하는 데 사용되는 컨텍스트를 더 세밀하게 제어할 수 있습니다. 값 집합에 포함된 값 하나를 값 집합의 합계로 나누면 전체 값에 대한 해당 값의 백분율을 구할 수 있습니다. 예를 들어, 수식  $\text{[판매 수익]}/\text{Sum}(\text{[판매 수익]})$ 을 사용하면 총 수익에 대한 백분율로 판매 수익을 계산할 수 있습니다.

아래 표에서 총 백분율 열의 수식은 다음과 같습니다.

$\text{[판매 수익]}/(\text{Sum}(\text{[판매 수익] In Report}))$

또한, 연간 백분율 열의 수식은 다음과 같습니다.

$\text{[판매 수익]}/(\text{Sum}(\text{[판매 수익] In Section}))$

2001

Year	Quarter	Sales revenue	Percentage of Total	Percentage of Year
2001	Q1	\$2660700	0.07	0.33
2001	Q2	\$2279003	0.06	0.28
2001	Q3	\$1367841	0.04	0.17
2001	Q4	\$1788580	0.05	0.22

이 두 수식은 확장 구문 키워드 Report와 Section을 사용하여 Sum 함수가 전체 연도의 총 수익과 연도별 수익을 각각 계산하도록 합니다.

## 관련 정보

[확장 구문을 사용하여 기본 계산 컨텍스트 수정 \[페이지 24\]](#)

### 3.3.3.1 변수로 분산 수식 단순화

분산은 통계 용어입니다. 값 집합의 분산은 평균에 대한 이러한 값의 분포를 측정합니다.

Var 함수는 분산을 한 단계로 계산하지만 분산을 직접 계산해 보면 변수를 사용하여 복잡한 수식을 단순화하는 방법을 쉽게 파악할 수 있습니다. 분산을 직접 계산하려면 다음 작업을 수행해야 합니다.

- 판매된 품목의 평균 수를 계산합니다.
- 판매된 품목의 각 수와 평균 수 사이의 차이를 계산한 후 이 값을 제공합니다.
- 차이 제곱을 모두 더합니다.
- 이 합계를 값의 개수 - 1로 나눕니다.

분기별로 판매된 품목의 수를 보여 주는 보고서에 분산을 포함시키고자 합니다. 변수를 사용하여 단순화하지 않을 경우 다음과 같이 복잡한 수식이 됩니다.

```
Sum((( [판매 수량] - Average([판매 수량] ForEach [분기]) In Report) * ([판매 수량] - Average([판매 수량] ForEach [분기]) In Report)) In [분기]) / (Count ([판매 수량] ForEach [분기]) - 1)
```

## 분산 수식 만들기

분산 수식을 만드는 데는 여러 단계가 필요합니다. 이러한 각 단계를 변수에 캡슐화할 수 있습니다. 만드는 변수는 다음과 같습니다.

- 판매된 품목의 평균 수
- 관찰 수(즉, 판매된 품목 수의 개별 값에 대한 수)
- 관찰 및 평균 사이의 차이 제곱
- 이러한 차이 합계를 관찰 수-1로 나눔

분산 수식은 다음과 같습니다.

변수	수식
평균 판매량	Average([판매량] In ([분기])) In Report
관찰 수	Count([판매량] In ([분기])) In Report

변수	수식
차이 제공	Power(([판매량] - [평균 판매량]);2)
분산	Sum([차이 제공] ln ([분기]))/[관찰 수] - 1)

최종 수식은 다음과 같이 됩니다.

Sum ([차이 제공])/[관찰 수] - 1)

이 수식이 훨씬 이해하기 쉽습니다. 이와 같이 단순화된 버전의 수식을 사용하면 세부적인 사항에 대해 혼란스러워 않고 이 수식이 어떤 계산을 하는지 상위 수준에서 쉽게 파악할 수 있습니다. 그러면 상위 수준의 수식에서 참조되는 변수의 수식을 검사하여 해당 구성 요소를 이해할 수 있습니다.

예를 들어, 수식에서는 차이 제공 변수를 참조하고 이 변수는 다시 판매된 평균 변수를 참조합니다. 차이 제공과 판매된 평균의 수식을 검사함으로써 수식으로 드릴다운하여 수식에서 어떠한 계산이 이루어지는지 자세히 파악할 수 있습니다.

### 3.3.4 함수 및 수식 연산자

연산자는 수식의 다양한 구성 요소를 연결합니다.

수식에는 수학, 조건부, 논리, 함수별 또는 확장 구문 연산자가 포함될 수 있습니다.

#### 3.3.4.1 수학 연산자

수학 연산자는 일상적인 산술과 비슷합니다.

수식에서 수학 연산을 수행하는 데는 더하기(+), 빼기(-), 곱하기(\*), 나누기(/) 연산자를 사용할 수 있습니다. 수식 [판매 수익] - [판매 비용]에는 수학 연산자인 빼기(-)가 포함되어 있습니다.

##### i 노트

'+' 연산자는 문자열과 함께 사용할 경우 문자열 연결 연산자가 됩니다. 즉, 문자열을 결합합니다. 예를 들어, 수식 "John" + " Smith"는 "John Smith"를 반환합니다.

#### 3.3.4.2 조건부 연산자

조건부 연산자는 값 사이의 비교 유형을 결정합니다.

연산자	설명
=	같음

연산자	설명
>	보다 큼
<	보다 작음
>=	크거나 같음
<=	작거나 같음
<>	같지 않음

다음과 같이 If 함수와 함께 조건부 연산자를 사용합니다.

```
If [수익]>10000 Then "높음" Else "낮음"
```

이 경우 수익이 10000 이상인 모든 행에 대해 "High"가 반환되고 다른 모든 행에 대해서는 "Low"가 반환됩니다.

### 3.3.4.3 논리 연산자

논리 연산자에는 And, Or, Not, Between 및 InList가 있습니다.

논리 연산자는 True 또는 False를 반환하는 부울 식에서 사용됩니다.

### 3.3.4.4 컨텍스트 연산자

컨텍스트 연산자는 확장 계산 구문을 구성하는 요소입니다.

확장 구문을 사용하면 계수나 수식 계산 시 참조할 차원을 정의할 수 있습니다.

### 3.3.4.5 함수별 연산자

일부 함수는 특정 연산자를 인수로 사용할 수 있습니다.

예를 들어, Previous 함수는 Self 연산자를 사용할 수 있습니다.

모든 함수는 ) 및 (을 사용하여 함수 인수를 묶습니다. 여러 개의 매개 변수를 허용하는 함수는 ;을 사용하여 매개 변수를 구분합니다.

## 4 계산 컨텍스트 이해

계산 컨텍스트는 계산할 때 결과를 생성하기 위해 고려하는 데이터입니다.

즉, 계수에 지정된 값은 계수를 계산하는 데 사용되는 차원에 따라 결정됩니다.

보고서에는 다음 두 종류의 개체가 있습니다.

- 차원은 값을 생성하는 비즈니스 데이터를 나타냅니다. 아울렛 매장, 연도 및 지역은 차원 데이터의 예입니다. 예를 들어, 아울렛 매장, 연도 또는 지역이 수익을 발생시킬 수 있습니다. 매장별 수익, 연도별 수익, 지역별 수익 등을 예로 들 수 있습니다.
- 계수는 차원 데이터로 생성된 숫자 데이터입니다. 수익과 판매량은 계수의 예입니다. 특정 매장에서 발생한 판매량을 예로 들 수 있습니다.

계수는 차원 데이터가 결합되어 생성될 수도 있습니다. 2005년도에 특정 매장에서 발생한 판매량을 예로 들 수 있습니다.

계수의 계산 컨텍스트에는 두 가지 구성 요소가 있습니다.

- 계수 값을 확인하는 차원 또는 차원 목록
- 계수 값을 확인하는 차원 데이터의 부분

계산 컨텍스트에는 두 가지 구성 요소가 있습니다.

- 입력 컨텍스트
- 출력 컨텍스트

### 관련 정보

[입력 컨텍스트 \[페이지 16\]](#)

[출력 컨텍스트 \[페이지 17\]](#)

### 4.1 입력 컨텍스트

계수 또는 수익의 입력 컨텍스트는 계산에 사용되는 차원 목록입니다.

입력 컨텍스트의 차원 목록은 값을 출력하는 함수의 괄호 안에 표시됩니다. 1차원만 포함되어 있더라도 차원 목록은 괄호로 묶어야 하고 세미콜론으로 차원을 구분해야 합니다.



예

#### 입력 컨텍스트 지정

연도 섹션이 있는 보고서와 고객 및 수익 열이 있는 각 섹션의 블록에서 입력 컨텍스트는 다음과 같습니다.



부분 보고서	입력 컨텍스트
섹션 머리글과 블록 바닥글	연도
블록의 행	연도, 고객

즉, 섹션 머리글과 블록 바닥글에는 연도별 집계 수익이 표시되고 블록의 각 행에는 연도 및 고객별 집계 수익이 표시됩니다(해당 연도에 고객별로 발생한 수익).

수식에 명시적으로 지정한 경우 입력 컨텍스트는 다음과 같습니다.

```
Sum ([수익] In ([연도]))
```

```
Sum ([수익] In ([연도];[고객]))
```

즉, 입력 컨텍스트의 차원은 입력 컨텍스트를 지정하고 있는 함수(이 경우 Sum)의 괄호 안에 표시됩니다.

## 4.2 출력 컨텍스트

출력 컨텍스트를 사용하면 나누기가 포함된 블록의 바닥글에 표시되는 것처럼 수식 값이 출력됩니다.



### 출력 컨텍스트 지정

다음 보고서는 연도에 대해 나누기를 적용한 연도별 및 분기별 수익과 연도별로 계산된 최저 수익을 보여 줍니다.

Year	Quarter	Sales revenue
	Q1	\$2660699.50
	Q2	\$2279003.00
	Q3	\$1367840.70
	Q4	\$1788580.40
<b>2001</b>		
	<b>최소:</b>	<b>\$1367840.70</b>
Year	Quarter	Sales revenue
	Q1	\$3326172.20
	Q2	\$2840650.80
	Q3	\$2879303.00
	Q4	\$4186120.00
<b>2002</b>		
	<b>최소:</b>	<b>\$2840650.80</b>
Year	Quarter	Sales revenue
	Q1	\$3742988.90
	Q2	\$4006717.50
	Q3	\$3953395.30
	Q4	\$3356041.10
<b>2003</b>		
	<b>최소:</b>	<b>\$3356041.10</b>

나누기가 없는 블록의 연도별 최저 수익을 표시하려면 수식에 출력 컨텍스트를 지정하면 됩니다. 이 경우 수식은 다음과 같습니다.

```
Min ([판매 수익]) In ([연도])
```

즉, 출력 컨텍스트는 출력 컨텍스트를 지정하고 있는 함수의 괄호 다음에 나타납니다. 이 경우 출력 컨텍스트는 연도별 최저 수익을 계산합니다.

이 수식이 들어 있는 열을 블록에 추가하면 결과는 다음과 같이 표시됩니다.

Year	Quarter	Sales revenue	Min by Year
2001	Q1	\$2660699.50	\$1367840.70
2001	Q2	\$2279003.00	\$1367840.70
2001	Q3	\$1367840.70	\$1367840.70
2001	Q4	\$1788580.40	\$1367840.70
2002	Q1	\$3326172.20	\$2840650.80
2002	Q2	\$2840650.80	\$2840650.80
2002	Q3	\$2879303.00	\$2840650.80
2002	Q4	\$4186120.00	\$2840650.80
2003	Q1	\$3742988.90	\$3356041.10
2003	Q2	\$4006717.50	\$3356041.10
2003	Q3	\$3953395.30	\$3356041.10
2003	Q4	\$3356041.10	\$3356041.10

연도별 최저값 열에는 앞의 보고서에서 나누기 바닥글에 나타나는 최저 수익이 들어 있는 것을 볼 수 있습니다.

이 예제에서는 입력 컨텍스트가 블록의 기본 컨텍스트인 (연도, 분기)이므로 입력 컨텍스트가 지정되어 있지 않습니다. 즉, 출력 컨텍스트를 통해 연도별 및 분기별로 출력할 수익을 확인할 수 있습니다. 입력 및 출력 수식이 모두 명시적으로 지정된 수식은 다음과 같이 나타냅니다.

```
Min ([판매 수익] In([연도];[분기])) In ([연도])
```

이 수식은 연도별 및 분기별로 수익을 계산한 다음 각 연도에 발생한 수익 중 가장 작은 값을 출력하게 됩니다.

연도별 최저값 열에 출력 컨텍스트를 지정하지 않으면 출력 값은 판매 수익 열의 값과 같게 됩니다. 이유 블록의 기본 컨텍스트에 해당 블록의 차원이 포함되어 있기 때문입니다. 즉, 각 연도/분기 조합의 수익 값은 하나만 있으므로 연도별 분기별 최저 수익이 연도별 분기별 수익과 같게 됩니다.

## 4.3 기본 계산 컨텍스트

계수는 보고서의 위치에 따라 기본 계산 컨텍스트를 갖습니다.

즉, 계수가 반환하는 값은 계수가 연결된 차원에 따라 다릅니다. 이러한 차원의 조합은 계산 컨텍스트를 나타냅니다.

사용자는 확장 구문을 사용하여 기본 컨텍스트를 변경할 수 있습니다. 즉, 계수를 생성하는 데 사용되는 차원 세트를 결정할 수 있습니다. 이는 계산 컨텍스트를 정의함을 의미합니다.



### 예 보고서의 기본 컨텍스트

이 예에서는 간단한 보고서 계수의 기본 계산 컨텍스트에 대해 설명합니다. 이 보고서는 고객별로 발생한 수익을 연도별 섹션으로 구분하여 표시합니다.

2005

합계: 8000

고객	수익
Harris	1000
Jones	3000
Walsh	4000
합계:	8000

보고서 합계: 8000

아래 표에는 이 보고서의 계수 계산 컨텍스트가 표시됩니다.

계수	값	컨텍스트
보고서 합계	20000	보고서의 모든 수익 합계
섹션 머리글 합계	8000	연도
고객 합계	1000, 3000, 4000	연도/고객
블록 바닥글 합계	8000	연도

## 관련 정보

- 계산 컨텍스트 이해 [페이지 16]
- 확장 구문을 사용하여 기본 계산 컨텍스트 수정 [페이지 24]

## 4.3.1 세로 테이블의 기본 컨텍스트

세로 테이블은 맨 위와 맨 아래에 각각 머리글과 바닥글이 그 사이에 데이터가 있는 표준 보고서 테이블입니다.

세로 테이블의 기본 컨텍스트는 다음과 같습니다.

계산식이 포함된 위치	입력 컨텍스트	출력 컨텍스트
머리글	블록 본문을 생성하는 데 사용된 차원과 계수	모든 데이터가 집계된 다음 계산 함수가 단일 값을 반환합니다.
블록 본문	현재 행을 생성하는 데 사용된 차원과 계수	입력 컨텍스트와 동일합니다.
바닥글	블록 본문을 생성하는 데 사용된 차원과 계수	모든 데이터가 집계된 다음 계산 함수가 단일 값을 반환합니다.

## 4.3.2 가로 테이블의 기본 컨텍스트

가로 테이블의 기본 컨텍스트는 세로 테이블의 기본 컨텍스트와 같습니다.

가로 테이블은 방향만 다를 뿐 세로 테이블과 비슷합니다.

가로 테이블의 모양은 BI 실행 패드 기본 설정에서 선택한 기본 설정 보기 로캘에 따라 달라집니다. 일부 로캘의 경우 영어 로캘과 같이, LTR(왼쪽에서 오른쪽) 인터페이스 위치를 사용하며 다른 로캘에서는 아랍어 로캘과 같이 RTL(오른쪽에서 왼쪽) 인터페이스 위치를 사용합니다.

LTR 로캘에서 머리글과 바닥글은 각각 맨 왼쪽과 오른쪽에 나타나고 그 사이에 데이터가 있습니다. RTL 로캘에서 머리글과 바닥글은 각각 맨 오른쪽과 왼쪽에 나타나고 그 사이에 데이터가 있습니다.

## 4.3.3 크로스탭 테이블의 기본 컨텍스트

크로스탭에서는 차원의 교집합에 계수가 나타나는 매트릭스 형태로 데이터가 표시됩니다.

크로스탭의 기본 컨텍스트는 다음과 같습니다.

계산식이 포함된 위치	입력 컨텍스트	출력 컨텍스트
머리글	블록 본문을 생성하는 데 사용된 차원과 계수	모든 데이터가 집계된 다음 계산 함수가 단일 값을 반환합니다.
블록 본문	블록 본문을 생성하는 데 사용된 차원과 계수	입력 컨텍스트와 동일합니다.
바닥글	블록 본문을 생성하는 데 사용된 차원과 계수	모든 데이터가 집계된 다음 계산 함수가 단일 값을 반환합니다.
VBody 바닥글	현재 열을 생성하는 데 사용된 차원과 계수	모든 데이터가 집계된 다음 계산 함수가 단일 값을 반환합니다.
HBody 바닥글	현재 행을 생성하는 데 사용된 차원과 계수	모든 데이터가 집계된 다음 계산 함수가 단일 값을 반환합니다.
VFooter	바닥글과 동일	모든 데이터가 집계된 다음 계산 함수가 단일 값을 반환합니다.
HFooter	바닥글과 동일	모든 데이터가 집계된 다음 계산 함수가 단일 값을 반환합니다.



예

### 크로스탭의 기본 컨텍스트

다음 보고서에서는 크로스탭의 기본 컨텍스트를 보여 줍니다.

		2003	2003	2003	2003	4,046,260
		Q1	Q2	Q3	Q4	4,046,260
California	2,992,679	729,745	789,398	775,766	697,770	2,992,679
DC	1,053,581	279,008	263,098	271,645	239,831	1,053,581
합계:	4,046,260	1,008,753	1,052,495	1,047,412	937,601	4,046,260

## 4.3.4 섹션의 기본 컨텍스트

섹션은 머리글, 본문 및 바닥글로 구성됩니다.

섹션의 기본 컨텍스트는 다음과 같습니다.

계산식이 포함된 위치	입력 컨텍스트	출력 컨텍스트
본문	데이터를 섹션 데이터로 제한하기 위해 필터링된 보고서의 차원과 계수	모든 데이터가 집계된 다음 계산 함수가 단일 값을 반환합니다.



예

### 섹션의 기본 컨텍스트

다음 보고서에서는 섹션의 기본 컨텍스트를 보여 줍니다.

2001		8,096,123.6
Quarter	Sales revenue	Section
Q1	\$2,660,700	8,096,123.6
Q2	\$2,279,003	8,096,123.6
Q3	\$1,367,841	8,096,123.6
Q4	\$1,788,580	8,096,123.6
합계:	8,096,123.6	
2002		13,232,246.0
Quarter	Sales revenue	Section
Q1	\$3,326,172	13,232,246.0
Q2	\$2,840,651	13,232,246.0
Q3	\$2,879,303	13,232,246.0
Q4	\$4,186,120	13,232,246.0
합계:	13,232,246.0	
2003		15,059,142.8
Quarter	Sales revenue	Section
Q1	\$3,742,989	15,059,142.8
Q2	\$4,006,718	15,059,142.8
Q3	\$3,953,395	15,059,142.8
Q4	\$3,356,041	15,059,142.8
합계:	15,059,142.8	

### 4.3.5 나누기의 기본 컨텍스트

나누기는 머리글, 본문 및 바닥글로 구성됩니다.

나누기의 기본 컨텍스트는 다음과 같습니다.

계산식이 포함된 위치	입력 컨텍스트	출력 컨텍스트
머리글	나누기의 현재 인스턴스	모든 데이터가 집계된 다음 계산 함수가 단일 값을 반환합니다.
바닥글	나누기의 현재 인스턴스	모든 데이터가 집계된 다음 계산 함수가 단일 값을 반환합니다.



예

### 나누기의 기본 컨텍스트

다음 보고서에서는 나누기의 기본 컨텍스트를 보여 줍니다.

Year	Quarter	\$8096124
	Q1	\$2660700
	Q2	\$2279003
	Q3	\$1367841
	Q4	\$1788580
2001		
	합계:	\$8096124

  

Year	Quarter	\$13232246
	Q1	\$3326172
	Q2	\$2840651
	Q3	\$2879303
	Q4	\$4186120
2002		
	합계:	\$13232246

## 4.4 확장 구문을 사용하여 기본 계산 컨텍스트 수정

확장 구문은 사용자가 수식이나 계수에 추가할 수 있는 컨텍스트 연산자를 사용하여 계산 컨텍스트를 지정합니다.

계수 또는 수식 컨텍스트는 입력 컨텍스트와 출력 컨텍스트로 구성됩니다.

### 관련 정보

[확장 구문 키워드 \[페이지 237\]](#)

[확장 구문 연산자 \[페이지 24\]](#)

### 4.4.1 확장 구문 연산자

컨텍스트 연산자를 사용하여 입력 및 출력 컨텍스트를 명시적으로 지정합니다.



다음 표에는 컨텍스트 연산자가 나열되어 있습니다.

연산자	설명
In	컨텍스트에서 사용할 명시적인 차원 목록을 지정합니다.
ForEach	기본 컨텍스트에 차원을 추가합니다.
ForAll	기본 컨텍스트에서 차원을 제거합니다.

ForAll 및 ForEach 연산자는 차원이 여러 개인 기본 컨텍스트가 있을 때 유용합니다. 대개는 In을 사용하여 명시적으로 목록을 지정하는 것보다 ForAll과 ForEach를 사용하여 컨텍스트를 "추가"하거나 "제거"하는 것이 더 쉽습니다.

#### 4.4.1.1 In 컨텍스트 연산자

In 컨텍스트 연산자는 컨텍스트에 차원을 명시적으로 지정합니다.



##### In을 사용하여 컨텍스트에 차원 지정

이 예제의 보고서는 연도와 판매 수익을 보여 줍니다. 데이터 공급자에는 분기 개체도 들어 있지만 블록에는 이 차원이 포함되어 있지 않습니다. 대신 각 연도의 분기별 최고 수익을 나타내기 위한 추가 열을 포함하려고 합니다. 이 보고서는 다음과 같습니다.

연도	매출	최대 분기별 수익
2001	\$8,096,123.60	\$2,660,699.50
2002	\$13,232,246.00	\$4,186,120.00
2003	\$15,059,142.80	\$4,006,717.50

분기별 최고 수익 열의 값은 이 블록을 분기 차원이 포함된 블록과 함께 검사하여 얻은 값임을 알 수 있습니다.

연도	분기	매출
2001	Q1	\$2,660,699.50
2001	Q2	\$2,279,003.00
2001	Q3	\$1,367,841.00
2001	Q4	\$1,788,580.00
	최대값:	\$2,660,699.50

연도	분기	매출
	Q1	\$3,326,172.00
	Q2	\$2,840,651.00
	Q3	\$2,879,303.00
	Q4	\$4,186,120.00
	최대값:	\$4,186,120.00

연도	분기	매출
	Q1	\$3,742,989.00
	Q2	\$4,006,717.50
	Q3	\$3,953,395.00
	Q4	\$3,356,041.00
	최대값:	\$4,006,717.50

분기별 최고 수익 열에는 각 연도의 분기별로 가장 높은 수익이 표시됩니다. 예를 들어, 2002년에는 4분기의 수익이 가장 높았으므로 분기별 최고 수익에는 2002년을 나타내는 행의 4분기 수익이 표시됩니다.

In 연산자를 사용할 경우 분기별 최고 수익의 수식은 다음과 같습니다.

```
Max ([판매 수익] In ([연도];[분기])) In ([연도])
```

이 수식은 각 (연도, 분기) 조합의 최고 판매 수익을 계산한 다음 이 값을 연도별로 출력합니다.

#### i 노트

블록의 기본 출력 컨텍스트는 연도이므로 이 수식에 출력 컨텍스트를 명시적으로 지정할 필요는 없습니다.

### 4.4.1.2 ForEach 컨텍스트 연산자

ForEach 연산자는 컨텍스트에 차원을 추가합니다.



#### ForEach 연산자를 사용하여 컨텍스트에 차원 추가

다음 표에서는 분기 차원이 포함되어 있지만 분기 차원이 블록에 포함되지 않은 보고서의 각 분기에 대한 최고 수익을 보여 줍니다.

연도	매출	최대 분기별 수익
2001	8096123.60	2660699.50
2002	13232246.00	4186120.00
2003	15059142.80	4006717.50

ForEach 연산자가 포함되지 않은 분기별 최고 수익 열에 대한 수식을 만들 수 있습니다.

```
Max ([판매 수익] In ([연도];[분기])) In ([연도])
```

ForEach 컨텍스트 연산자를 사용하여 다음 수식으로 동일한 결과를 얻을 수 있습니다.

```
Max ([판매 수익] ForEach ([분기])) In ([연도])
```

이유 연도 차원이 블록의 기본 입력 컨텍스트이기 때문입니다. ForEach 연산자를 사용하여 컨텍스트에 분기 차원을 추가하고 입력 컨텍스트를 ([연도];[분기])로 지정합니다.

### 4.4.1.3 ForAll 컨텍스트 연산자

ForAll 컨텍스트 연산자는 컨텍스트에서 차원을 제거합니다.



#### ForAll을 사용하여 컨텍스트에서 차원 제거

연도, 분기 및 판매 수익이 표시된 보고서에 다음 블록에 표시된 것과 같이 각 연도의 총 수익을 표시하는 열을 추가하려고 합니다.

Year	Quarter	Sales revenue	Yearly Total
2001	Q1	\$2660700	\$8096124
2001	Q2	\$2279003	\$8096124
2001	Q3	\$1367841	\$8096124
2001	Q4	\$1788580	\$8096124
2002	Q1	\$3326172	\$13232246
2002	Q2	\$2840651	\$13232246
2002	Q3	\$2879303	\$13232246
2002	Q4	\$4186120	\$13232246
2003	Q1	\$3742989	\$15059143
2003	Q2	\$4006718	\$15059143
2003	Q3	\$3953395	\$15059143
2003	Q4	\$3356041	\$15059143

연도별로 총 수익을 구하려면 입력 컨텍스트가 (연도)여야 하는데 기본 입력 컨텍스트는 (연도; 분기)입니다. 따라서 다음과 같이 수식에 ForAll ([분기])를 지정하여 입력 컨텍스트에서 분기를 제거할 수 있습니다.

```
Sum([판매 수익] ForAll ([분기]))
```

In 연산자를 사용하여 분기를 제거할 수도 있습니다. 이 경우 수식은 다음과 같습니다.

```
Sum([판매 수익] In ([연도]))
```

이 버전의 수식에서는 분기를 제거하고 연도만 남겨 두는 것이 아니라 명시적으로 연도를 컨텍스트로 지정합니다.

## 5 스마트 계수를 사용하여 값 계산

스마트 계수는 유니버스의 기반인 데이터베이스(관계형 또는 OLAP)에 의해 값이 계산되는 계수입니다.

데이터베이스가 반환하는 상세 값을 기반으로 계산되는 기본 계수와는 다릅니다. 스마트 계수에서 반환된 데이터는 SAP BusinessObjects Business Intelligence 플랫폼의 Web Intelligence 구성 요소에서 기본적으로 지원하지 않는 방법을 통해 집계됩니다.

스마트 계수가 포함된 쿼리는 보고서에 필요한 모든 계산 컨텍스트에서 계수를 계산합니다. 보고서를 변경할 경우 이러한 컨텍스트가 변경될 수 있습니다. 따라서 필요한 컨텍스트가 변경된 후 데이터를 새로 고칠 때마다 쿼리가 변경됩니다.

이러한 보고서를 편집할 때 변경 사항을 반영하기 위해 보고서 새로 고침이 필요하다는 #TOREFRESH 메시지가 자동으로 보고서에 삽입됩니다. 문서 속성 대화 상자에서 문서 자동 새로 고침 옵션을 선택하여 보고서를 자동으로 업데이트하도록 선택할 수 있습니다.

### i 노트

계수 위임은 정적이며 디자인 시 보고서 정의를 기반으로 정의됩니다. 일부 경우("if [choice]= 1 then [dimension 1] else [dimension 2]"를 기반으로 하는 수식) 차원 컨텍스트가 런타임 시 가변적입니다. 이 경우 계수 계산을 위임할 수 없어 빈 값이 반환됩니다.

스마트 계수는 데이터베이스의 지원 없이 모든 컨텍스트에서 계산될 수 있는 기본 집계 함수(Max, Min, Count, Sum, Average) 집합을 지원하는 기본 계수와 다른 방식으로 작동합니다. 예를 들어, [국가], [지역] 차원과 수익의 합계를 계산하는 [수익] 계수가 포함된 쿼리를 작성할 경우 처음에는 블록에 국가, 지역 및 수익이 표시됩니다. 블록에서 지역을 제거할 경우 국가의 모든 지역에 대한 수익을 집계하면 데이터를 새로 고치지 않고도 각 국가에 대한 총 수익을 계산할 수 있습니다. 이 경우 스마트 계수를 사용하려면 데이터를 새로 고쳐야 합니다.

계산 컨텍스트는 생성된 쿼리에서 그룹화 집합으로 표시됩니다.

### 5.1 그룹화 집합 및 스마트 계수

그룹화 집합은 계수의 결과를 생성하는 차원 집합입니다.

스마트 계수의 데이터를 반환하는 생성된 SQL에는 보고서에 있는 해당 계수의 모든 집계에 대한 그룹화 집합이 포함됩니다.

### 예

#### 쿼리의 그룹화 집합

쿼리에 [국가], [지역], [도시] 차원과 [수익] 스마트 계수가 있습니다. 이러한 개체는 다음 그룹화 집합을 가정하여 모든 가능한 컨텍스트에서 수익을 계산합니다.

- 전체 스마트 계수 값
- (국가, 지역, 도시)별 스마트 계수 값
- (국가, 도시)별 스마트 계수 값

- (도시)별 스마트 계수 값
- (지역, 도시)별 스마트 계수 값
- (지역)별 스마트 계수 값
- (국가, 지역)별 스마트 계수 값
- (국가)별 스마트 계수 값

데이터베이스가 UNION을 지원할 경우 각 그룹화 집합은 생성된 SQL에서 UNION 절에 표시됩니다.

그룹화 집합은 보고서에 필요한 계산 컨텍스트(보고서 구조가 변경될 경우 변경될 수 있음)에 따라 업데이트됩니다.

### 5.1.1 그룹화 집합 관리

스마트 계수가 포함된 쿼리를 처음 작성하고 실행할 경우 생성된 SQL에는 쿼리 개체에 포함된 가장 자세한 수준에서 스마트 계수를 계산하는 데 필요한 그룹화 집합이 포함됩니다.

예를 들어, [국가], [지역], [도시] 차원과 [수익] 스마트 계수가 포함된 쿼리를 작성할 경우 생성된 SQL에 (국가, 지역, 도시) 그룹화 집합이 나타납니다. SQL에는 항상 가장 자세한 그룹화 집합이 나타납니다. 다른 그룹화 집합은 보고서가 변경될 경우 추가 및 제거됩니다.

블록에서 [도시] 차원을 제거할 경우 수익 값을 반환하려면 (국가, 지역) 그룹화 집합이 필요합니다. 이 그룹화 집합은 쿼리 SQL에서 사용할 수 없으므로 [수익] 셀에 #TOREFRESH가 표시됩니다. 데이터를 새로 고치면 #TOREFRESH가 수익 값으로 바뀝니다.

그런 다음 블록에서 [도시] 차원을 바꾸면 (국가, 지역) 그룹화 집합이 더 이상 필요 없습니다. 다음 번에 데이터를 새로 고칠 때 해당 그룹화 집합이 쿼리 SQL에서 제거되며 관련 값이 삭제됩니다.

보고서 데이터를 새로 고칠 때마다 보고서에 필요한 계산 컨텍스트에 따라 그룹화 집합이 포함되거나 삭제됩니다.

스마트 계수 값을 표시할 수 없는 경우도 있습니다. 이 경우 계수 셀에 #UNAVAILABLE이 표시됩니다.

## 5.2 스마트 계수 및 분석 범위

분석 범위가 포함된 쿼리를 작성할 경우 초기 그룹화 집합에는 범위 개체 대신 결과 개체가 포함됩니다.

쿼리는 결과 개체와 범위 개체의 조합에서 가능한 모든 그룹화 집합을 생성하는 것이 아닙니다.



#### 예 분석 범위가 포함된 쿼리 및 스마트 계수

쿼리에 [국가]와 [수익] 결과 개체가 있습니다. 분석 범위에는 [지역]과 [도시] 차원이 있습니다. 쿼리를 실행하면 SQL에 (국가) 그룹화 집합이 포함되며 블록에 [국가]와 [수익]이 표시됩니다.

## 5.3 스마트 계수 및 SQL

### 5.3.1 그룹화 집합 및 UNION 연산자

일부 데이터베이스는 GROUPING SETS 연산자를 사용하여 그룹화 집합을 명시적으로 지원합니다.

스마트 계수가 포함된 쿼리를 작성할 경우 생성된 SQL에 여러 결과 집합과 UNION 연산자가 사용되어 GROUPING SETS의 결과가 시뮬레이션됩니다.



예

#### UNION 연산자를 사용하여 가져온 그룹화 집합

이 예제에서는 [국가], [지역], [도시] 차원과 [수익] 스마트 계수가 포함된 쿼리에 대해 설명합니다.

#### i 노트

이 예제에서는 스마트 계수가 합계를 계산합니다. 실제로는 Web Intelligence에서 합계가 기본적으로 지원되므로 이 집계에 대해 스마트 계수가 필요하지 않습니다.

쿼리를 처음 실행하면 그룹화 집합은 (국가, 지역, 도시)입니다. 전체 SQL 쿼리는 이 그룹화 집합을 반환하므로 SQL의 UNION 연산자가 필요 없습니다.

테이블에서 [도시] 차원을 제거할 경우 수익(처음에는 #TOREFRESH로 표시됨)을 표시하려면 (국가, 지역) 그룹화 집합이 필요합니다. 데이터를 새로 고치고 나면 SQL은 다음과 같습니다.

```
SELECT
  SELECT
    0 AS GID,
    country.country_name,
    region.region_name,
    NULL,
    sum(city.revenue)
  FROM
    country,
    region,
    city
  WHERE
    ( country.country_id=region.country_id )
    AND ( region.region_id=city.region_id )
  GROUP BY
    country.country_name,
    region.region_name
  UNION
  SELECT
    1 AS GID,
    country.country_name,
    region.region_name,
    city.city_name,
    sum(city.revenue)
  FROM
    country,
    region,
    city
  WHERE
    ( country.country_id=region.country_id )
    AND ( region.region_id=city.region_id )
  GROUP BY
    country.country_name,
```

```
region.region_name,  
city.city_name
```

각 그룹화 집합은 SELECT 문으로 표현되며 고유한 ID(GID 열)를 사용합니다. UNION 연산자가 포함된 쿼리의 각 SELECT 문에는 동일한 개수의 열을 사용해야 하므로 전체 차원 집합을 포함하지 않는 그룹화 집합에는 빈 열 (SELECT '')이 포함됩니다.

[국가] 및 [수익]이 포함된 새 블록을 보고서에 추가할 경우 (국가) 그룹화 집합이 필요합니다. 이제 생성된 SQL에는 다음과 같은 세 가지 그룹화 집합이 포함됩니다.

```
SELECT  
  0 AS GID,  
  country.country_name,  
  region.region_name,  
  NULL,  
  sum(city.revenue)  
FROM  
  country,  
  region,  
  city  
WHERE  
  ( country.country_id=region.country_id )  
  AND ( region.region_id=city.region_id )  
GROUP BY  
  country.country_name,  
  region.region_name  
UNION  
SELECT  
  1 AS GID,  
  country.country_name,  
  NULL,  
  NULL,  
  sum(city.revenue)  
FROM  
  country,  
  city,  
  region  
WHERE  
  ( country.country_id=region.country_id )  
  AND ( region.region_id=city.region_id )  
GROUP BY  
  country.country_name  
UNION  
SELECT  
  2 AS GID,  
  country.country_name,  
  region.region_name,  
  city.city_name,  
  sum(city.revenue)  
FROM  
  country,  
  region,  
  city  
WHERE  
  ( country.country_id=region.country_id )  
  AND ( region.region_id=city.region_id )  
GROUP BY  
  country.country_name,  
  region.region_name,  
  city.city_name
```

## 5.4 스마트 계수 및 수식

### 5.4.1 수식이 포함된 스마트 계수 및 차원

수식 또는 변수가 스마트 계수의 계산 컨텍스트에서 차원으로 표시되고 수식이 계수에 필요한 그룹화 집합을 결정할 경우 스마트 계수의 값을 표시할 수 있습니다.

예를 들어 스마트 계수와 차원이 이제 다음에 대한 값을 반환합니다.

- 하이퍼링크 마법사로 만든 URL
- 차원에 대한 단순 연결(또는 공백 제거).
- FormatDate가 [날짜]에 사용된 경우

#### i 노트

ForEach, ForAll, In, Where, Rank, Previous, RelativeValue, RelativeDate, TimeDim 함수 및 ([selection] =1) then [dim1] else [dim2] 수식에 Min, Max, Last 또는 First가 사용된 경우 집계 함수에 대해서도 #UNAVAILABLE 메시지가 반환됩니다

### 5.4.2 수식의 스마트 계수

수식 위치에 의해 포함된 컨텍스트와는 다른 계산 컨텍스트가 수식에 필요한 경우에도 수식에 스마트 계수가 포함되어 있으면 값이 반환될 수 있습니다.

예를 들어, 보고서에 다음과 같은 블록이 있습니다.

국가	지역	수익
미국	북부	10000
미국	남부	15000
미국	동부	14000
미국	서부	12000

수식을 사용하여 이 테이블에 추가로 열을 포함할 경우

```
[수익] ForAll ([지역])
```

열의 초기 값은 #TOREFRESH입니다. 계산에서 지역을 제외하는 수식에 그룹화 집합(국가)이 필요하기 때문입니다. 데이터를 새로 고치면(국가) 그룹화 집합이 쿼리에 추가되고 계수의 값이 표시됩니다.

#### 관련 정보

[ForAll 컨텍스트 연산자 \[페이지 27\]](#)



## 5.5 스마트 계수 및 필터

### 5.5.1 스마트 측정 및 필터에 관한 제한

테이블 또는 상위 컨텍스트(보고서 필터)에 필터가 없을 때 테이블 본문에서 스마트 측정을 평가할 수 있습니다.

다음 표에서는 필터가 있을 때 스마트 측정을 평가하는 방법을 설명합니다.

보고서에 필터가 있을 때의 스마트 측정 평가 방법

다음에 대한 필터가 있을 때	스마트 측정 평가 방법
계수	스마트 측정이 올바르게 평가되지만, 일부 행은 테이블에서 제거됩니다.
이미 테이블 측에 속한 차원	스마트 측정이 올바르게 평가되지만, 일부 행은 테이블에서 제거됩니다. 필터링 후 집계가 없기 때문에 스마트 측정을 평가할 수 있습니다.
테이블의 측에 속하지 않고 필터 피연산자가 단일 값일 때(필터가 한 값/행을 반환함)의 차원.	스마트 측정이 올바르게 평가됩니다. 필터링 후 집계 없기 때문에 스마트 측정을 평가할 수 있습니다.
테이블의 측에 속하지 않고 필터 피연산자가 다중 값인 경우(필터가 다수의 값/행을 반환할 수 있음)의 차원.	이 경우에는 집계 전에 필터링이 완료되고 테이블의 한 행에 대해 집계 필요하기 때문에 스마트 측정을 평가할 수 없습니다(#UNAVAILABLE이 표시됨).

### 5.5.2 차원의 스마트 계수 및 필터

스마트 계수 값이 종속되는 차원에 다중 값 필터를 적용했지만 차원이 계수의 계산 컨텍스트에 명시적으로 표시되지 않을 경우 스마트 계수가 값을 반환할 수 없으므로 셀에 #UNAVAILABLE이 표시됩니다.

이 상황은 입력 컨트롤에서 보고서 필터를 가져오는 경우에도 적용됩니다.

보고서에서 계수를 필터링하고 집계해야 하지만 보고서 수준 필터가 적용된 후에는 스마트 계수를 집계할 수 없으므로 #UNAVAILABLE이 표시되는 것입니다. 생성된 SQL에 쿼리 필터를 추가하여 계수를 계산할 수는 있지만 이 방법을 사용하면 동일한 쿼리를 기반으로 하는 다른 보고서가 영향을 받을 수 있다는 위험이 따릅니다.

#### i 노트

다중 값 필터는 보다 큼, 목록에 있음 또는 보다 작음 등의 연산자를 사용하여 다중 값을 필터링합니다. 같음 등의 단일 값 필터는 #UNAVAILABLE 오류를 발생시키지 않고 적용할 수 있습니다.

#### i 노트

집계가 필요하지 않을 경우 자격이 계수인 변수로 수식을 정의하고 사용된 차원이 해당 변수를 가진 블록에 포함되도록 하면 됩니다. 표시 효과를 높이기 위해 해당 열을 숨길 수도 있습니다.



예

### 차원의 스마트 계수 및 필터

쿼리에 국가 및 제품 차원과 수익 스마트 계수가 포함되어 있습니다. 국가와 수익이 블록이 표시됩니다. 제품 값을 "드레스" 또는 "재킷"으로 제한하는 보고서 필터를 적용할 경우 수익 셀에 #UNAVAILABLE이 표시됩니다.

국가	Revenue
프랑스	#UNAVAILABLE
미국	#UNAVAILABLE
합계:	#UNAVAILABLE

제품을 "재킷"으로만 제한할 경우 값이 표시됩니다.

국가	수익
미국	971,444
합계:	971,444

### i 노트

필터가 테이블에 있기 때문에 국가에 대한 다중 값 필터가 결과를 반환합니다.

## 5.5.3 스마트 계수 필터링

테이블 바닥글의 값은 테이블에서 사용자에게 표시되는 항목의 집계여야 합니다.

테이블에서 사용자에게 표시되는 항목이 로컬에서 필터링된 경우 로컬에서 필터링된 항목의 위임된 집계가 반환되지 않습니다.



예

### 스마트 계수 필터링

국가	OrderAmountDel
브라질	28,833.36
중국	51,384.33
프랑스	68,630.22
미국	3,529,511.14
합계:	3,678,359.05
합계:	3,678,359.05

다음 표의 데이터가 OrderAmountDel > 60,000으로 필터링된 경우

테이블 컨텍스트(국가별)에서 OrderAmountDel이 60,000보다 큰 행이 테이블에 표시됩니다.

바닥글의 합계는 표시되는 행의 합계를 계산합니다.

계산이 백엔드에 집계를 위임했지만 Web Intelligence에서 로컬 필터링을 수행했기 때문에 집계를 위임할 수 없어 집계  
의 경우 #UNAVAILABLE이 반환됩니다.

국가	OrderAmountDel
프랑스	68,630.22
미국	3,529,511.14
합계:	#UNAVAILABLE
합계:	3,598,141.36

## 5.5.4 스마트 계수 및 드릴 필터

드릴 필터는 단일 필터입니다.

드릴 도구 모음을 사용하여 직접 드릴할 수 있습니다.

## 5.5.5 스마트 계수 및 중첩된 OR 필터

필터링된 차원 중 최소 한 개가 블록에 나타나지 않는 중첩된 OR 필터는 블록의 스마트 계수에 대해 #UNAVAILABLE 오  
류를 생성합니다.

이 오류는 일부 로컬 처리(예: 필터링, 일부 특정 Web Intelligence 수식) 후 스마트 계수를 로컬에서 집계해야 하는데 이  
계수가 위임되지 않았기 때문에 발생합니다.

## 6 함수, 연산자 및 키워드

### 6.1 함수

수식 함수는 몇 개의 범주로 분류됩니다.

#### i 노트

중국어, 일본어, 헝가리어, 한국어, 폴란드어, 터키어, 태국어, 러시아어에서는 함수, 키워드, 연산자가 번역되지 않습니다. 인터페이스에서 함수가 영어로 표시됩니다.

범주	설명
집계	값의 집합을 더하거나 평균을 구하여 데이터를 집계합니다.
문자	문자열을 조작합니다.
날짜 및 시간	날짜 또는 시간 데이터를 반환합니다.
문서	문서에 대한 데이터를 반환합니다.
데이터 공급자	문서의 데이터 공급자에 대한 데이터를 반환합니다.
논리	TRUE 또는 FALSE를 반환합니다.
숫자	숫자 데이터를 반환합니다.
기타	위의 범주 어디에도 속하지 않은 함수입니다.
집합	계층구조에서 멤버 집합을 반환합니다.

#### 6.1.1 사용자 지정 서식

테이블에서 사용자 지정 서식 유형을 사용하여 모든 셀에 대해 사용자 지정된 서식을 정의할 수 있습니다.

Web Intelligence 함수에서는 아래에 표시된 요일/날짜, 달력 및 하루 중 시간 문자 정의가 적용됩니다.

#### i 노트

Web Intelligence HTML 인터페이스의 테이블에서는 사용자 지정 서식을 숫자와 날짜에 적용할 수 없습니다.

다음 표에는 사용자 지정 서식을 만드는 데 사용할 수 있는 문자열이 나열되어 있습니다.

문자	표시	예
#	해당 자릿수. 숫자의 자릿수가 형식을 지정하는 데 사용된 # 문자의 수보다 적으면 앞에 0을 추가하지 않습니다.	'12345'에 #, ##0 형식을 지정하면 '12,345'(로캘에서 그룹화 구분 기호로 침표가 정의된 경우) 또는 '12 345'(로캘에서 그룹화 구분 기호로 공백이 정의된 경우)와 같이 표시됩니다.
0	해당 자릿수. 숫자의 자릿수가 형식을 지정하는 데 사용된 0 문자의 수보다 적으면 숫자 앞에 0이 추가됩니다.	'123'에 #0,000 형식을 지정하면 '0,123'으로 표시됩니다.
,	현재 로캘에 정의된 그룹화 구분 기호	'1234567'에 #, ##0 형식을 지정하면 '1,234,567'(로캘에서 그룹화 구분 기호로 침표가 정의된 경우) 또는 '1 234 567'(로캘에서 그룹화 구분 기호로 공백이 정의된 경우)과 같이 표시됩니다.
.	현재 로캘에 정의된 소수 구분 기호	'12.34'에 #.#0 형식을 지정하면 '12.34'(현재 로캘에 소수 구분 기호가 마침표로 정의된 경우) 또는 '12,34'(현재 로캘에 소수 구분 기호가 침표로 정의된 경우)로 표시됩니다.
[%]%	결과 다음에 백분율 기호(%)를 표시하고 결과에 100을 곱합니다.	0.50은 50%가 됩니다.
%	결과 다음에 % 기호를 표시하지만 결과에 100을 곱하지는 않습니다.	0.50은 0.50%가 됩니다.
	단어 잘림 방지 공백()	'1234567'에 # ##0 형식을 지정하면 '1234 567'과 같이 표시됩니다.
1, 2, 3, a, b, c, \$, £, €(등)	영숫자	<p>'705.15'에 \$#. #0 형식을 지정하면 '\$705.15'가 표시되고 #.#0 € 형식을 지정할 경우 '705,15 €'가 표시됩니다.</p> <div> <b>i 노트</b>  영숫자는 작은 따옴표로 구분해 주어야만 형식 지정 문자로 잘못 해석되는 것을 피할 수 있습니다. 예를 들어 ## 표시는 '123 4'가 되고 '# #'의 경우 '# 1234'가 됩니다. </div>
[Red], [Blue], [Green], [Yellow], [Gray], [White], [Dark Red], [Dark Blue], [Dark Green]	지정된 색상의 값	'150'에 #, ##0 [Red] 형식을 지정하면 '150'이 빨간색 텍스트로, #, ##0 [Blue] 형식을 지정하면 '150'이 파란색 텍스트로 표시됩니다.
Day/date characters	(day, date)	
d	앞에 0이 없는 날짜. 날짜가 두 자리 미만일 경우 앞에 0이 없는 날짜가 표시됩니다.	월의 첫 번째 날에 d 형식을 지정하면 '1'이 표시됩니다.
dd	앞에 0이 있는 날짜. 날짜가 두 자리 미만일 경우 앞에 0이 있는 날짜가 표시됩니다.	월의 첫 번째 날에 dd 형식을 지정하면 '01'이 표시됩니다.

문자	표시	예
ddd	요일의 약어 선택한 로케에서 요일 이름의 첫 글자를 대문자로 사용할 경우 첫 글자가 대문자로 표시됩니다.	ddd 형식을 사용하면 영어의 경우 'Monday'가 'Mon'으로, 프랑스어의 경우 lundi가 lun으로 표시됩니다.
Dddd	모든 로케에 대해 요일 이름의 첫 글자를 대문자로 표시합니다.	Dddd 형식을 사용하면 영어의 경우 'Monday'가 'Mon'으로, 프랑스어의 경우 lundi가 Lun으로 표시됩니다.
dddd	전체 요일 이름. 선택한 로케에서 요일 이름의 첫 글자를 대문자로 사용할 경우 첫 글자가 대문자로 표시됩니다.	dddd 형식을 사용하면 영어의 경우 'Monday'가 'Monday'로, 프랑스어의 경우 lundi로 표시됩니다.
DDDD	대문자로 표시된 전체 요일 이름.	DDDD 형식을 사용하면 영어의 경우 'Monday'가 'MONDAY'로, 프랑스어의 경우 LUNDI로 표시됩니다.
dddd dd	요일 다음에 공백이 나오고 그 다음에 날짜가 표시됩니다.	'월요일'에 dddd 형식을 지정하면 '월요일 01'이 표시됩니다.
Calendar characters	(week, month, year)	
M	앞에 0이 없는 월. 월의 숫자가 두 자리 미만일 경우 앞에 0이 없는 숫자가 표시됩니다.	'1월'에 M 형식을 지정하면 '1'이 표시됩니다.
MM	앞에 0이 있는 월. 월의 숫자가 두 자리 미만일 경우 앞에 0이 있는 숫자가 표시됩니다.	'1월'에 MM 형식을 지정하면 '01'이 표시됩니다.
mmm	월의 약어 선택한 로케에서 첫 글자를 대문자로 사용할 경우 첫 글자가 대문자로 표시됩니다.	mmm 형식을 사용하면 영어의 경우 'January'가 Jan으로, 프랑스어의 경우 'jan'으로 표시됩니다.
Mmmm	월의 약어 모든 로케에 대해 첫 글자가 대문자로 표시됩니다.	mmm 형식을 사용하면 영어의 경우 'January'가 Jan으로, 프랑스어의 경우 'Jan'으로 표시됩니다.
mmmm	전체 월 이름 선택한 로케에서 첫 글자를 대문자로 사용할 경우 첫 글자가 대문자로 표시됩니다.	mmmm 형식을 사용하면 영어의 경우 'January'가 January로, 프랑스어의 경우 janvier로 표시됩니다.
MMMM	모두 대문자로 표시된 전체 월 이름.	MMMM 형식을 사용하면 영어의 경우 'January'가 JANUARY로, 프랑스어의 경우 JANVIER로 표시됩니다.
ww	연중 주 번호.	2015년 1월 9일의 경우, ww 형식을 지정하면 2015년의 7번째 주이므로 '02'로 표시됩니다.
w	연중 주 번호(선행 0 생략).	2015년 1월 9일의 경우, w 형식을 지정하면 2015년의 7번째 주이므로 '2'로 표시됩니다.
W	월의 주 번호.	2015년 1월 9일의 경우, W 형식을 지정하면 1월의 2번째 주이므로 '2'로 표시됩니다.
yy	연도의 마지막 두 자리	'2003'에 yy 형식을 지정하면 '03'으로 표시됩니다.
yyyy	연도의 네 자리 숫자 모두	'2003'에 yyyy 형식을 지정하면 '2003'으로 표시됩니다.

문자	표시	예
Time of day characters	(hours, minutes, seconds, am/pm)	
hh:mm:ss a	앞에 0이 없는 시간과 앞에 0이 있는 분 및 초. "a" 문자는 시간 다음에 오는 AM 또는 PM을 나타냅니다(사용 가능한 경우).	'21:05:03'에 hh:mm:ss a 형식을 지정하면 로캘이 영어일 때 '9:05:03 PM'이 표시됩니다.
H	24시간제(0부터 시작)에 따른 시간입니다. 한 자릿수 시간의 경우 앞에 0이 붙지 않습니다.	'21:00'에 H 형식을 지정하면 '21'이 표시됩니다. 가능한 값은 0-23입니다.
HH	24시간제(0부터 시작)에 따른 시간.	'21:00'에 HH 형식을 지정하면 '21'이 표시됩니다. 가능한 값은 00-23입니다.
k	24시간제(1부터 시작)에 따른 시간입니다. 한 자릿수 시간의 경우 앞에 0이 붙지 않습니다.	'21:00'에 k 형식을 지정하면 '21'이 표시됩니다. 가능한 값은 1-24입니다.
kk	24시간제(01부터 시작)에 따른 시간.	'21:00'에 kk 형식을 지정하면 '21'이 표시됩니다. 가능한 값은 01-24입니다.
hh	12시간제에 따른 시간	'21:00'에 hh 형식을 지정하면 '09'로 표시됩니다.
HH:mm	한 자릿수 시간 앞에 0이 있는 시간 및 분	'07:15:00'에 HH:mm 형식을 지정하면 '07:15'와 같이 표시됩니다.
HH:mm:ss	한 자릿수 시간 앞에 0이 있는 시간, 분, 초	'오전 7:15'에 HH:mm:ss 형식을 지정하면 '오전 7:15:00'으로 표시됩니다.
mm:ss	한 자릿수 시간 앞에 0이 있는 분, 초	'오전 7:15:03'에 mm:ss 형식을 지정하면 '오후 3:03'으로 표시됩니다.
z	날짜/시간 값에 대한 시간대 정보는 다음과 같습니다. GMT+/-HH:mm	

## 6.1.2 집계 함수

### 6.1.2.1 Aggregate

#### 설명

지정된 멤버 집합에 대한 계수의 기본 집계를 반환합니다.

#### 함수 그룹

#### 집계

## 구문

```
num Aggregate (measure[;member_set])
```

## 입력

매개 변수	설명	유형	필수
measure	임의의 계수	계수	예
member_set	집계 계산에 사용되는 멤버 집합	멤버 집합	아니요

## 참고

- Aggregate와 함께 확장 구문 컨텍스트 연산자를 사용할 수 있습니다.
- member\_set를 포함하면 Aggregate에서는 멤버 집합의 모든 멤버에 대한 계수의 집계 값을 반환합니다.
- member\_set에는 세미콜론(;)으로 구분된 여러 집합을 포함시킬 수 있습니다.
- 멤버 집합 목록은 {}로 묶어야 합니다.
- 멤버 집합 표현식에 정확한 멤버나 노드가 지정되지 않은 경우, 참조된 계층구조가 테이블에 있어야 합니다. 그러면 멤버 집합 표현식이 테이블에 있는 계층구조의 현재 멤버를 참조합니다. 계층구조가 테이블에 없는 경우 함수가 #MULTIVALUE 메시지를 반환합니다.
- 필요한 집계를 쿼리에서 사용할 수 없는 경우 위임된 계수 집계는 #TOREFRESH를 반환합니다. 새로운 수준의 집계를 얻으려면 문서를 새로 고쳐야 합니다. 예를 들어, 필터 표시줄을 사용할 때 사용자가 "모든 값"을 선택하기에 앞서 어떤 값을 선택하는 경우와 그 반대로 선택한 값에 앞서 "모든 값"을 선택하는 경우 이 값이 반환됩니다.

## 예

[판매 수익] 계수의 기본 집계는 Sum이고 [캘리포니아]가 [지역] 계층구조(국가 > 주 > 시)의 멤버인 경우 `Aggregate([판매 수익];{Descendants([지역]&[미국].[캘리포니아];1)})`은 캘리포니아에 있는 모든 시의 총 판매 수익을 반환합니다.

## 관련 정보

[계층구조에서 멤버 및 멤버 집합 참조 \[페이지 243\]](#)



## 6.1.2.2 집계 함수에서의 멤버 선택

### 설명

특정 집계 함수의 경우, 블록 안에 계층구조가 있을 때 집계 상황에 맞게 멤버 선택을 정의할 수 있습니다.

### 함수 그룹

### 집계

### 구문

```
=AggregationFunction([my object];{memberselection})
```

### 입력

매개 변수	설명	형식	필수
AggregationFunction	다음 중 하나여야 합니다. <ul style="list-style-type: none"><li>Aggregate</li><li>Average</li><li>Coun</li><li>Ma</li><li>Mi</li><li>Sum</li></ul>	Aggregate 함수	예
내 개체	차원 또는 계수	차원 또는 계수	예
memberselection	지정된 멤버, 또는 Set 함수를 사용하여 계산된 멤버 집합. memberselection은 중괄호로 둘러싸야 합니다. 멤버 집합의 각 구성 멤버는 세미콜론으로 구분됩니다. <pre>{[member one];[member two];CalculatedMemberSet() }</pre> CalculatedMemberSet에는 다음 Set 함수 중 하나를 사용: <ul style="list-style-type: none"><li>Ancestor</li><li>Descendant</li></ul>	멤버, 또는 Set 함수를 사용하여 계산된 멤버 집합	예

매개 변수	설명	형식	필수
	<ul style="list-style-type: none"> <li>• Lag</li> <li>• Children</li> <li>• Parent</li> <li>• Siblings</li> </ul>		

## 설명

Set 함수는 개체, 수준, 멤버를 매개 변수로 사용합니다. 함수에 개체만 지정했고 해당 개체가 블록 내에 존재하는 계층 구조 개체라면, 함수는 현재의 멤버를 사용합니다. 특정 하나의 멤버를 지정하려면 다음 구문을 사용합니다.

```
[HierarchicalObject].[RootMember].[ChildMember].[ChildMember]
```

Microsoft 및 Essbase .UNIX 소스의 경우 다음 구문을 통해 수준을 지정할 수 있습니다.

```
[HierarchicalObject].[LevelName]
```

## 예

다음 예는 모두 영어 버전의 데이터 소스에서 가져왔습니다.

### 예

다음은 2002년과 2001년의 인터넷 매출 차이를 알아내는 예입니다.

```
=Sum([Internet Sales].[Internet Sales Amount];{[Calendar].[Date.Calendar]&[All Periods].[CY 2002]}) + Sum([Internet Sales].[Internet Sales Amount];{[Calendar].[Date.Calendar]&[All Periods].[CY 2001]})
```

다음은 두 멤버를 지정하는 예입니다.

```
=Sum([Internet Sales].[Internet Sales Amount];{[Calendar].[Date.Calendar]&[All Periods].[CY 2002];[Calendar].[Date.Calendar]&[All Periods].[CY 2001]})
```

Date.Calendar	Internet Sales Amount	{CY 2001;CY 2002}
 All Periods	29,358,677.22	9,796,717.18
 CY 2001	3,266,373.66	
 CY 2002	6,530,343.53	
 CY 2003	9,791,060.3	
 CY 2004	9,770,899.74	

## 예

다음은 제품 계층구조가 있을 때 바이크에 관련된 모든 제품의 인터넷 매출을 알아내는 예입니다. 단, 제품 중 두 개는 다른 카테고리에 속해 있습니다.

```
=Sum([Query 3].[Internet Sales].[Internet Sales Amount];{[Product Model  
Categories]&[All Products].[Accessories].[Bike Racks];[Product Model  
Categories]&[All Products].[Accessories].[Bike Stands];[Product Model  
Categories]&[All Products].[Bikes]})
```

Bikes Amount	28,397,095.65
[-] All Products	29,358,677.22
[-] Accessories	700,759.96
+ Bike Racks	39,360
+ Bike Stands	39,591
+ Bottles and Cages	56,798.19
+ Cleaners	7,218.6
+ Fenders	46,619.58
+ Helmets	225,335.6
+ Hydration Packs	40,307.67
+ Tires and Tubes	245,529.32
[-] Bikes	28,318,144.65
+ Mountain Bikes	9,952,759.56
+ Road Bikes	14,520,584.04
+ Touring Bikes	3,844,801.05
[-] Clothing	339,772.61
+ Caps	19,688.1
+ Gloves	35,020.7
+ Jerseys	172,950.68
+ Shorts	71,319.81
+ Socks	5,106.32
+ Vests	35,687

#### 예

다음은 북미 지역 국가들 간의 인터넷 매출액을 비교는 예입니다. 먼저 캐나다와 미국을 비교한 다음 전세계 다른 국가들과 비교합니다.

먼저 북미 국가들의 총계를 얻는데 이 예에서는 캐나다와 미국에만 관심을 둡니다.

```
=Sum([Query 2].[Internet Sales].[Internet Sales Amount];{[Customer Geography]&[All Customers].[Canada];[Customer Geography]&[All Customers].[United States]})
```

Customer Geography		Internet Sales Amount
[-] All Customers	11,367,634.37	29,358,677.22
+ Australia	11,367,634.37	9,061,000.58
+ Canada	11,367,634.37	1,977,844.86
+ France	11,367,634.37	2,644,017.71
+ Germany	11,367,634.37	2,894,312.34
+ United Kingdom	11,367,634.37	3,391,712.21
+ United States	11,367,634.37	9,389,789.51

이어서 북미와 다른 모든 국가를 비교합니다.

```
=([Query 2].[Internet Sales].[Internet Sales Amount] / Sum([Query 2].[Internet Sales].[Internet Sales Amount];{[Customer Geography]&[All Customers].[Canada];[Customer Geography]&[All Customers].[United States]}))
```

Customer Geography		Internet Sales Amount
[-] All Customers	258.27%	29,358,677.22
+ Australia	79.71%	9,061,000.58
+ Canada	17.40%	1,977,844.86
+ France	23.26%	2,644,017.71
+ Germany	25.46%	2,894,312.34
+ United Kingdom	29.84%	3,391,712.21
+ United States	82.60%	9,389,789.51

전세계 고객의 총계가 북미의 2.5배이고 오스트레일리아는 북미의 80% 수준임을 볼 수 있습니다.

## 관련 정보

[Aggregate \[페이지 39\]](#)

### 6.1.2.3 Average

#### 설명

계수의 평균 값을 반환합니다.

#### 함수 그룹

#### 집계

#### 구문

```
num Average(measure[;member_set][;IncludeEmpty])
```

#### 입력

매개 변수	설명	유형	필수
measure	임의의 계수	계수	예
member_set	멤버의 집합	멤버 집합	아니요
IncludeEmpty	빈 행을 계산에 포함시킵니다.	키워드	아니요 (빈 행은 기본적으로 제외됨)

#### 참고

- Average 함수와 함께 확장 구문 컨텍스트 연산자를 사용할 수 있습니다.
- member\_set를 포함하면 Average 함수는 멤버 집합의 모든 멤버에 대한 계수의 평균 값을 반환합니다.
- member\_set에는 세미콜론(;)으로 구분된 여러 집합을 포함시킬 수 있습니다.
- 멤버 집합 목록은 {}로 묶어야 합니다.
- 멤버 집합 표현식에 정확한 멤버나 노드가 지정되지 않은 경우, 참조된 계층구조가 테이블에 있어야 합니다. 그러면 멤버 집합 표현식에 테이블에 있는 계층구조의 현재 멤버를 참조합니다. 계층구조가 테이블에 없는 경우 함수가 #MULTIVALUE 메시지를 반환합니다.
- 필요한 집계를 쿼리에서 사용할 수 없는 경우 위임된 계수 집계는 #TOREFRESH를 반환합니다. 새로운 수준의 집계를 얻으려면 문서를 새로 고쳐야 합니다. 예를 들어, 필터 표시줄을 사용할 때 사용자가 "모든 값"을 선택하기에 앞서 어떤 값을 선택하는 경우와 그 반대로 선택한 값에 앞서 "모든 값"을 선택하는 경우 이 값이 반환됩니다.

- 그룹에 대해 제공된 위임된 계수의 경우 로컬 집계(그룹화된 값의 계수 값 집계)가 필요하므로 #UNAVAILABLE을 반환합니다.  
위임된 계수의 로컬 집계를 "if then else" 수식 또는 그룹 값에 대해 강제로 수행하더라도 #MULTIVALUE 메시지가 계속 반환됩니다.

## 예제

[판매 수익] 계수에 41569, 30500, 40000 및 50138 값이 있는 경우 `Average([판매 수익])`은 40552를 반환합니다.

[캘리포니아]가 [지역] 계층구조(국가 > 주 > 시)의 멤버인 경우 `Average([판매 수익]; {[지역]&[미국].[캘리포니아].children})`은 캘리포니아에 있는 모든 시의 평균 판매 수익을 반환합니다.

## 관련 정보

[계층구조에서 멤버 및 멤버 집합 참조 \[페이지 243\]](#)

[IncludeEmpty 연산자 \[페이지 228\]](#)

## 6.1.2.4 Count

### 설명

값 집합에 있는 값 수를 반환합니다.

### 함수 그룹

### 집계

### 구문

```
integer Count(aggregated_data[;member_set][;IncludeEmpty][;Distinct|All])
```

## 입력

매개 변수	설명	유형	필수
aggregated_data	임의의 차원, 계수, 계층구조, 수준 또는 멤버 집합	차원, 계수, 계층구조, 멤버 집합	예
member_set	개수 계산에 사용되는 멤버 집합입니다.	멤버 집합	아니요
IncludeEmpty	빈 값을 계산에 포함시킵니다.	키워드	아니요
Distinct All	고유 값만(차원의 기본값) 또는 모든 값(계수의 기본값)을 계산에 포함시킵니다.	키워드	아니요

## 참고

- Count 함수와 함께 확장 구문 컨텍스트 연산자를 사용할 수 있습니다.
- IncludeEmpty를 두 번째 인수로 지정하면 함수가 빈(null) 값을 계산에 포함시킵니다.
- Distinct|All 매개 변수를 지정하지 않으면 차원의 기본값은 Distinct이고 계수의 기본값은 All이 됩니다.
- member\_set를 포함시키면 Count가 개수를 member\_set에 있는 값 수로 제한합니다.
- member\_set에는 세미콜론(;)으로 구분된 여러 집합을 포함시킬 수 있습니다.
- 멤버 집합 목록은 {}로 묶어야 합니다.
- 멤버 집합 표현식에 정확한 멤버나 노드가 지정되지 않은 경우, 참조된 계층구조가 테이블에 있어야 합니다. 그러면 멤버 집합 표현식에 테이블에 있는 계층구조의 현재 멤버를 참조합니다. 계층구조가 테이블에 없는 경우 함수가 #MULTIVALUE 메시지를 반환합니다.
- 필요한 집계를 쿼리에서 사용할 수 없는 경우 위임된 계수 집계는 #TOREFRESH를 반환합니다. 새로운 수준의 집계를 얻으려면 문서를 새로 고쳐야 합니다. 예를 들어, 필터 표시줄을 사용할 때 사용자가 **모든 값**을 선택하기에 앞서 어떤 값을 선택하는 경우와 그 반대로 하나의 값을 선택하기에 앞서 **모든 값**을 선택하는 경우가 해당됩니다.
- 그룹에 대해 제공된 위임된 계수의 경우 로컬 집계(그룹화된 값의 계수 값 집계)가 필요하므로 #UNAVAILABLE을 반환합니다.  
위임된 계수의 로컬 집계를 "if then else" 수식 또는 그룹 값에 대해 강제로 수행하더라도 #MULTIVALUE 메시지가 계속 반환됩니다.

## 예제

Count("Test")는 1을 반환합니다.

Count([도시]; Distinct)는 도시 목록에 서로 다른 도시가 5개 있는 경우 중복으로 인해 목록에 5개보다 많은 행이 있더라도 5를 반환합니다.

Count([도시]; All)는 도시 목록에 도시가 10개 있는 경우 일부가 중복되더라도 10을 반환합니다.

Count([도시]; IncludeEmpty)는 도시 목록에 5개의 도시와 1개의 빈 행이 있는 경우 6을 반환합니다.

Count([제품]; {[지역]&[주]})는 [지역] 계층구조의 [주] 수준에 있는 총 제품 수를 반환합니다.



## 관련 정보

[IncludeEmpty 연산자 \[페이지 228\]](#)

[Distinct/All 연산자 \[페이지 227\]](#)

### 6.1.2.5 First

#### 설명

데이터 집합의 첫 번째 값을 반환합니다.

#### 함수 그룹

#### 집계

#### 구문

```
input_type First (dimension|measure)
```

#### 입력

매개 변수	설명	유형	필수
dimension measure	임의의 차원 또는 계수	차원 또는 계수	예

#### 참고

- 나누기 바닥글에 있는 경우 `First`는 나누기에 있는 첫 번째 값을 반환합니다.
- 섹션 바닥글에 있는 경우 `First`는 섹션에 있는 첫 번째 값을 반환합니다.

#### 예제

테이블 바닥글에 있는 경우 `First([수익])`은 테이블에 있는 [수익]의 첫 번째 값을 반환합니다.

## 6.1.2.6 Interpolation

### 설명

보간에 의해 빈 계수 값을 계산합니다.

### 함수 그룹

### 집계

### 구문

```
num Interpolation(measure[;PointToPoint|Linear] [;NotOnBreak| (reset_dims)] [;Row|Col])
```

### 입력

매개 변수	설명	유형	필수
measure	임의의 계수	계수	예
PointToPoint Linear	보간 방식: <ul style="list-style-type: none"><li>PointToPoint - 점 간 보간</li><li>Linear - 최소 제공 보간을 사용하는 선형 회귀 분석</li></ul>	키워드	아니요 (PointToPoint가 기본값임)
NotOnBreak  reset_dims	<ul style="list-style-type: none"><li>NotOnBreak - 함수가 블록 및 섹션 나누기에서 계산을 다시 설정하지 않도록 합니다.</li><li>reset_dims - 보간을 다시 설정하는 데 사용되는 차원 목록입니다.</li></ul>	키워드  차원 목록	아니요
Row Col	계산 방향을 설정합니다.	키워드	(Row가 기본값임)

## 참고

- `Interpolation`은 누락 값이 들어 있는 계수에 대해 선 그래프를 만들 때 특히 유용합니다. 이 함수를 사용하면 그래프에 끊어진 선 및 점이 아닌 연속된 선이 그려집니다.
- 최소 제곱 보간법을 이용한 선형 회귀 분석을 사용하면 사용 가능한 모든 계수 값을 최대한 가깝게 통과하는  $f(x) = ax + b$  형식의 1차 방정식을 계산하여 누락된 값을 계산합니다.
- 점 간 보간법은 누락된 값의 두 인접 값을 통과하는  $f(x) = ax + b$  형식의 1차 방정식을 계산하여 누락된 값을 계산합니다.
- 계수의 정렬 순서는 `Interpolation`에서 반환되는 값에 영향을 미칩니다.
- `Interpolation`을 포함하는 수식에는 정렬이나 순위를 적용할 수 없습니다.
- 값 목록에 값이 하나만 있는 경우 `Interpolation`은 이 값을 사용하여 모든 누락 값을 제공합니다.
- 보간 처리된 계수에 적용된 필터는 필터가 어떤 값에 영향을 미치는가에 따라 `Interpolation`에서 반환된 값을 변경할 수 있습니다.

## 예제

`Interpolation([값])`은 기본 점 간 보간 방법을 사용하여 다음 누락 값을 제공합니다.

요일	값	<code>Interpolation([값])</code>
월요일	12	12
화요일	14	14
수요일		15
목요일	16	16
금요일		17
토요일		18
일요일	19	19

## 관련 정보

[Linear 연산자 \[페이지 229\]](#)

[PointToPoint 연산자 \[페이지 230\]](#)

## 6.1.2.7 Last

### 설명

차원 또는 계수의 마지막 값을 반환합니다.

## 함수 그룹

### 집계

## 구문

```
input_type Last(dimension|measure)
```

## 입력

매개 변수	설명	유형	필수
dimension measure	임의의 차원 또는 계수	차원 또는 계수	예

## 참고

- 나누기 바닥글에 있는 경우 Last는 나누기에 있는 마지막 값을 반환합니다.
- 섹션 바닥글에 있는 경우 Last는 섹션에 있는 마지막 값을 반환합니다.

## 예제

테이블 바닥글에 있는 경우 First([수익])은 테이블에 있는 [수익]의 첫 번째 값을 반환합니다.

## 6.1.2.8 Max

### 설명

값 집합에서 가장 큰 값을 반환합니다.

## 함수 그룹

### 집계

## 구문

```
input_type Max(aggreated_data[;member_set])
```

## 입력

매개 변수	설명	유형	필수
aggreated_data	임의의 차원, 계수, 계층구조, 수준 또는 멤버 집합	차원, 계수, 계층구조, 수준 또는 멤버 집합	예
member_set	멤버의 집합	멤버 집합	아니요

## 참고

- Max 함수와 함께 확장 구문 컨텍스트 연산자를 사용할 수 있습니다.
- member\_set를 포함시키면 Max 함수는 멤버 집합에 있는 모든 멤버에 대해 집계된 데이터의 최대값을 반환합니다.
- member\_set에는 세미콜론(;)으로 구분된 여러 집합을 포함시킬 수 있습니다.
- 멤버 집합 목록은 {}로 묶어야 합니다.
- 멤버 집합 표현식에 정확한 멤버나 노드가 지정되지 않은 경우, 참조된 계층구조가 테이블에 있어야 합니다. 그러면 멤버 집합 표현식이 테이블에 있는 계층구조의 현재 멤버를 참조합니다. 계층구조가 테이블에 없는 경우 함수가 #MULTIVALUE 메시지를 반환합니다.
- 필요한 집계를 쿼리에서 사용할 수 없는 경우 위임된 계수 집계는 #TOREFRESH를 반환합니다. 새로운 수준의 집계를 얻으려면 문서를 새로 고쳐야 합니다. 예를 들어, 필터 표시줄을 사용할 때 사용자가 "모든 값"을 선택하기에 앞서 어떤 값을 선택하는 경우와 그 반대로 선택한 값에 앞서 "모든 값"을 선택하는 경우 이 값이 반환됩니다.
- 그룹에 대해 제공된 위임된 계수의 경우 로컬 집계(그룹화된 값의 계수 값 집계)가 필요하므로 #UNAVAILABLE을 반환합니다.  
위임된 계수의 로컬 집계를 "if then else" 수식 또는 그룹 값에 대해 강제로 수행하더라도 #MULTIVALUE 메시지가 계속 반환됩니다.

## 예

[판매 수익] 계수의 값이 3000, 60034 및 901234인 경우 Max([판매 수익])은 901234를 반환합니다.

[City] 차원에 "Aberdeen" 및 "London" 값이 있는 경우 Max([City])은 "London"을 반환합니다.

[미국]이 [지역] 계층구조(국가 > 주 > 시)의 멤버인 경우 Max([판매 수익]; {[지역].[미국].Children})은 미국에서 판매 수익이 가장 높은 주를 반환합니다.

## 6.1.2.9 Median

### 설명

계수의 중앙값(중간값)을 반환합니다.

### 함수 그룹

### 집계

### 구문

```
num Median(measure)
```

### 입력

매개 변수	설명	유형	필수
measure	임의의 계수	계수	예

### 참고

숫자 집합에 짝수 개의 값이 있는 경우 Median은 중간에 있는 두 값의 평균을 사용합니다.

### 예제

`Median([수익])`은 [수익] 값이 835420, 971444 및 1479660인 경우 971,444를 반환합니다.

## 6.1.2.10 Min

### 설명

값 집합에서 가장 작은 값을 반환합니다.

## 함수 그룹

### 집계

### 구문

```
input_type Min(aggregated_data[;member_set])
```

### 입력

매개 변수	설명	유형	필수
aggregated_data	임의의 차원, 계수, 계층구조, 수준 또는 멤버 집합	차원, 계수, 계층구조, 수준 또는 멤버 집합	예
member_set	멤버의 집합	멤버 집합	아니요

### 참고

- Min 함수와 함께 확장 구문 컨텍스트 연산자를 사용할 수 있습니다.
- member\_set를 포함시키면 Min 함수는 멤버 집합에 있는 모든 멤버에 대해 집계된 데이터의 최소값을 반환합니다.
- member\_set에는 세미콜론(;)으로 구분된 여러 집합을 포함시킬 수 있습니다.
- 멤버 집합 목록은 {}로 묶어야 합니다.
- 멤버 집합 표현식에 정확한 멤버나 노드가 지정되지 않은 경우, 참조된 계층구조가 테이블에 있어야 합니다. 그러면 멤버 집합 표현식이 테이블에 있는 계층구조의 현재 멤버를 참조합니다. 계층구조가 테이블에 없는 경우 함수가 #MULTIVALUE 메시지를 반환합니다.
- 필요한 집계를 쿼리에서 사용할 수 없는 경우 위임된 계수 집계는 #TOREFRESH를 반환합니다. 새로운 수준의 집계를 얻으려면 문서를 새로 고쳐야 합니다. 예를 들어, 필터 표시줄을 사용할 때 사용자가 "모든 값"을 선택하기에 앞서 어떤 값을 선택하는 경우와 그 반대로 선택한 값에 앞서 "모든 값"을 선택하는 경우 이 값이 반환됩니다.
- 그룹에 대해 제공된 위임된 계수의 경우 로컬 집계(그룹화된 값의 계수 값 집계)가 필요하므로 #UNAVAILABLE을 반환합니다.  
위임된 계수의 로컬 집계를 "if then else" 수식 또는 그룹 값에 대해 강제로 수행하더라도 #MULTIVALUE 메시지가 계속 반환됩니다.

### 예제

[판매 수익] 계수에 값 3000, 60034 및 901234가 있는 경우 Min([판매 수익])은 3000을 반환합니다.

[City] 차원에 "Aberdeen" 및 "London" 값이 있는 경우 Min([City])는 "Aberdeen"을 반환합니다.

`Min([판매 수익]; {[지역]&[미국].children})`은 [미국]이 [국가]>[주]>[시] 수준을 사용하는 [지역] 계층구조의 멤버인 경우 미국에서 판매 수익이 가장 낮은 주를 반환합니다.

## 6.1.2.11 Mode

### 설명

데이터 집합에서 가장 자주 사용된 값을 반환합니다.

### 함수 그룹

### 집계

### 구문

```
input_type Mode(dimension|measure)
```

### 입력

매개 변수	설명	유형	필수
dimension measure	임의의 차원 또는 계수	계수	예

### 참고

- 데이터 집합에 다른 모든 값보다 자주 나오는 값이 포함되어 있지 않은 경우 `Mode`는 `null`을 반환합니다.

### 예제

`[수익]`에 100, 200, 300, 200 값이 있는 경우, `Mode([수익])`는 200을 반환합니다.

`Mode([국가])`는 가장 자주 나오는 [국가] 값을 반환합니다.



## 6.1.2.12 Percentage

### 설명

계수 값을 포함 컨텍스트의 백분율로 표시합니다.

### 함수 그룹

### 집계

### 구문

```
num Percentage (measure [ ; Break ] [ ; Row | Col ] )
```

### 입력

매개 변수	설명	유형	필수
measure	임의의 계수	계수	예
Break	테이블 나누기를 처리합니다.	키워드	아니요
Row Col	계산 방향을 설정합니다.	키워드	아니요

### 예제

다음 표에서 백분율 열에는 Percentage ([판매 수익])이라는 수식이 있습니다.

연도	판매 수익	백분율
2001	1000	10
2002	5000	50
2003	4000	40
합계:	10000	100

기본적으로 포함 컨텍스트는 표의 계수 합계입니다. 선택 사항인 Break 인수를 사용하면 함수에서 표의 구분을 고려하도록 만들 수 있습니다. 이 경우 기본 포함 컨텍스트는 표 섹션이 됩니다.

다음 표에서 백분율 열에는 Percentage ([판매 수익]; Break)라는 수식이 있습니다.

연도	분기	판매 수익	백분율
2001	1분기	1000	10
	2분기	2000	20
	3분기	5000	50
	4분기	2000	20
2001	합계:	10000	100

연도	분기	판매 수익	백분율
2002	1분기	2000	20
	2분기	2000	20
	3분기	5000	50
	4분기	1000	10
2002	합계:	10000	100

열 또는 행 사이에 Percentage 함수를 사용할 수 있고 선택 사항인 ROW|COL 인수를 사용하여 이러한 사용 방식을 명시적으로 지정할 수 있습니다. 예를 들어, 다음 크로스탭에서 % 열에는 Percentage ([판매 수익];ROW) 라는 수식이 있습니다.

	1분기	%	2분기	%	3분기	%	4분기	%
2001	1000	10	2000	20	5000	50	2000	20
2002	2000	20	2000	20	5000	50	1000	10

## 6.1.2.13 Percentile

### 설명

계수의 n번째 백분위수를 반환합니다.

### 함수 그룹

### 숫자

### 구문

```
num Percentile(measure;percentile)
```

## 입력

매개 변수	설명	유형	필수
measure	임의의 계수	계수	예
백분위수	10진수로 표현된 백분율	숫자	예

## 참고

n번째 백분위수는 집합에 있는 숫자의 n%보다 크거나 같은 숫자입니다. n%는 0.n 형태로 표현합니다.

## 예제

[계수]에 숫자 집합 (10;20;30;40;50)이 있는 경우 `Percentile([계수];0.3)`은 집합에 있는 숫자의 30%보다 크거나 같은 숫자인 22를 반환합니다.

## 6.1.2.14 Product

### 설명

계수의 값을 곱합니다.

### 함수 그룹

### 집계

### 구문

```
num Product (measure)
```

## 입력

매개 변수	설명	유형	필수
measure	임의의 계수	계수	예

## 예제

`Product([계수])`는 [계수]에 값 2, 3, 5가 있는 경우 30을 반환합니다.

## 6.1.2.15 RunningAverage

### 설명

계수의 누적 평균을 반환합니다.

### 함수 그룹

### 집계

### 구문

```
num RunningAverage(measure[;Row|Col][;IncludeEmpty][;(reset_dims)])
```

RunningAverage의 각 섹션에서 재설정하려면 다음 구문을 사용하는 것이 좋습니다.

```
num RunningAverage(measure;section)
```

## 입력

매개 변수	설명	유형	필수
measure	임의의 계수	계수	예
Row Col	계산 방향을 설정합니다.	키워드	아니요

매개 변수	설명	유형	필수
IncludeEmpty	빈 값을 계산에 포함시킵니다.	키워드	아니요
reset_dims	지정한 차원을 기준으로 계산을 재설정합니다.	차원 목록	아니요
section	섹션이 설정되는 차원입니다.	키워드	섹션 재설정 시 예

## 참고

- RunningAverage 함수와 함께 확장 구문 컨텍스트 연산자를 사용할 수 있습니다.
- Row 및 Col 연산자를 사용하여 계산 방향을 설정할 수 있습니다.
- RunningAverage 함수에서 참조하는 계수에 대해 정렬을 적용하면 계수가 정렬된 후 누적 평균이 계산됩니다.
- 재설정 차원 목록에 차원이 하나만 있는 경우에도 항상 차원을 괄호로 묶어야 합니다.
- 재설정 차원 집합을 지정하는 경우 세미콜론으로 구분해야 합니다.
- RunningAverage는 블록 나누기 또는 새 섹션 추가 후에 자동으로 평균을 다시 설정하지 않습니다.

## 예제

RunningAverage([수익])은 다음 결과를 반환합니다.

국가	휴양지	수익	누적 평균
미국	Hawaiian Club	1,479,660	835,420
미국	Bahamas Beach	971,444	1,225,552
프랑스	French Riviera	835,420	1,095,508

RunningAverage([수익];([국가]))는 다음 결과를 반환합니다.

국가	휴양지	수익	누적 평균
미국	Hawaiian Club	1,479,660	835,420
미국	Bahamas Beach	971,444	1,225,552
France	French Riviera	835,420	835,420

RunningAverage([판매 수익];([분기])) 수식을 사용하여 [분기]의 섹션에서 RunningAverage를 사용하는 예제의 경우 다음 결과가 반환됩니다.

Q1		
구/군/시	판매 수익	누적 평균
뉴욕	\$1,987,114.70	\$1,987,114.70

Houston	\$1,544,627.80	\$1,765,871.25
로스앤젤레스	\$1,129,177.60	\$1,553,640.03
<b>Q2</b>		
구/군/시	판매 수익	누적 평균
뉴욕	\$2,028,090.70	\$2,028,090.70
Houston	\$1,380,838.20	\$1,704,464.45
로스앤젤레스	\$980,405.30	\$1,463,111.40

## 관련 정보

[IncludeEmpty 연산자 \[페이지 228\]](#)

[Row/Col 연산자 \[페이지 230\]](#)

## 6.1.2.16 RunningCount

### 설명

숫자 집합의 누적 개수를 반환합니다.

### 함수 그룹

### 집계

### 구문

```
num RunningCount (dimension|measure [; Row|Col] [; IncludeEmpty] [; (reset_dims) ] )
```

RunningCount의 각 섹션에서 재설정하려면 다음 구문을 사용하는 것이 좋습니다.

```
num RunningCount (dimension|measure; section)
```

## 입력

매개 변수	설명	유형	필수
dimension measure	임의의 차원 또는 계수	차원 또는 계수	예
Row Col	계산 방향을 설정합니다.	키워드	아니요
IncludeEmpty	빈 값을 계산에 포함시킵니다.	키워드	아니요
reset_dims	지정한 차원을 기준으로 계산을 재설정합니다.	차원 목록	아니요
section	섹션이 설정되는 차원입니다.	키워드	섹션 재설정 시 예

## 참고

- RunningCount 함수와 함께 확장 구문 컨텍스트 연산자를 사용할 수 있습니다.
- Row 및 Col 연산자를 사용하여 계산 방향을 설정할 수 있습니다.
- RunningCount 함수에서 참조하는 계수에 대해 정렬을 적용하면 계수가 정렬된 후 누적 개수가 계산됩니다.
- 재설정 차원 목록에 차원이 하나만 있는 경우에도 항상 차원을 괄호로 묶어야 합니다.
- 재설정 차원 집합을 지정하는 경우 세미콜론으로 구분해야 합니다.
- RunningCount는 블록 나누기 또는 새 섹션 추가 후에 자동으로 개수를 다시 설정하지 않습니다.

## 예제

RunningCount ([판매 수익])은 다음 표에 나와 있는 결과를 반환합니다.

국가	휴양지	판매 수익	누적 개수
미국	Hawaiian Club	1,479,660	1
미국	Bahamas Beach	971,444	2
프랑스	French Riviera	835,420	3

RunningCount ([수익]; ([국가]))는 다음 표에 나열된 결과를 반환합니다.

국가	휴양지	수익	누적 개수
미국	Hawaiian Club	1,479,660	1
미국	Bahamas Beach	971,444	2
France	French Riviera	835,420	1

RunningCount ([라인]; ([주])) 수식을 사용하고 [판매 수익]에서 입력 컨트롤을 통해 30,000달러를 초과하는 수익으로 목록을 제한하며 [주]의 섹션에서 RunningCount를 사용하는 예제의 경우 다음 결과가 반환됩니다.

주 1		
-----	--	--

라인	판매 수익	누적 개수
스웨트 티셔츠	\$186,191	1
블라우스	\$139,082	2
드레스	\$70,931	3
주 2		
라인	판매 수익	누적 개수
액세서리	\$344,617	1
스웨트 티셔츠	\$196,976	2
블라우스	\$105,597	3
드레스	\$76,290	4
스웨터	\$68,364	5

주 1에는 수익이 30,000달러를 초과한 세 개 라인이 있으며 주 2에는 수익이 30,000달러를 초과한 다섯 개 제품 라인이 있습니다.

## 관련 정보

[IncludeEmpty 연산자 \[페이지 228\]](#)

[Row/Col 연산자 \[페이지 230\]](#)

[IncludeEmpty 연산자 \[페이지 228\]](#)

[IncludeEmpty 연산자 \[페이지 228\]](#)

## 6.1.2.17 RunningMax

### 설명

차원 또는 계수의 누적 최대값을 반환합니다.

### 함수 그룹

### 집계



## 구문

```
input_type RunningMax (dimension|measure[;Row|Col] [(reset_dims)])
```

RunningMax의 각 섹션에서 재설정하려면 다음 구문을 사용하는 것이 좋습니다.

```
num RunningMax (measure;section)
```

## 입력

매개 변수	설명	유형	필수
dimension measure	임의의 차원 또는 계수	차원 또는 계수	예
Row Col	계산 방향을 설정합니다.	키워드	아니요
reset_dims	지정한 차원을 기준으로 계산을 재설정합니다.	차원 목록	아니요
section	섹션이 설정되는 차원입니다.	키워드	섹션 재설정 시 예

## 참고

- RunningMax 함수와 함께 확장 구문 컨텍스트 연산자를 사용할 수 있습니다.
- Row 및 Col 연산자를 사용하여 계산 방향을 설정할 수 있습니다.
- RunningMax 함수에서 참조하는 계수에 대해 정렬을 적용하면 계수가 정렬된 후 누적 최대값이 계산됩니다.
- 재설정 차원 목록에 차원이 하나만 있는 경우에도 항상 차원을 괄호로 묶어야 합니다.
- 재설정 차원 집합을 지정하는 경우 세미콜론으로 구분해야 합니다.
- RunningMax는 블록 나누기 또는 새 섹션 추가 후에 자동으로 최대값을 다시 설정하지 않습니다.

## 예제

RunningMax([수익])은 다음 표에 나열된 결과를 반환합니다.

국가	휴양지	수익	누적 최대값
프랑스	French Riviera	835,420	835,420
미국	Bahamas Beach	971,444	971,444
미국	Hawaiian Club	1,479,660	1,479,660

RunningMax([판매 수익];([구/군/시])) 수식을 사용하여 [구/군/시]의 섹션에서 RunningMax를 사용하는 예제의 경우 다음 결과가 반환됩니다.

오스틴		
분기	판매 수익	누적 최대값
Q1	\$775,482.70	\$775,482.70
Q2	\$667,850.30	\$775,482.70
Q3	\$581,470.40	\$775,482.70
Q4	\$674,869.80	\$775,482.70
보스턴		
분기	판매 수익	누적 최대값
Q1	\$312,896.40	\$312,896.40
Q2	\$291,431.00	\$312,896.40
Q3	\$249,529.00	\$312,896.40
Q4	\$429,850.20	\$429,850.20

## 관련 정보

[IncludeEmpty 연산자 \[페이지 228\]](#)

[Row/Col 연산자 \[페이지 230\]](#)

## 6.1.2.18 RunningMin

### 설명

차원 또는 계수의 누적 최소값을 반환합니다.

### 함수 그룹

### 집계

### 구문

```
input_type RunningMin (dimension|measure; [Row|Col]; [ (reset_dims) ] )
```

RunningMin의 각 섹션에서 재설정하려면 다음 구문을 사용하는 것이 좋습니다.

```
num RunningMin(measure;section)
```

## 입력

매개 변수	설명	유형	필수
dimension detail measure	임의의 차원 또는 계수	차원 또는 계수	예
Row Col	계산 방향을 설정합니다.	키워드	아니요
reset_dims	지정한 차원을 기준으로 계산을 재설정합니다.	차원 목록	아니요
section	섹션이 설정되는 차원입니다.	키워드	섹션 재설정 시 예

## 참고

- RunningMin과 함께 확장 구문 컨텍스트 연산자를 사용할 수 있습니다.
- Row 및 Col 연산자를 사용하여 계산 방향을 설정할 수 있습니다.
- RunningMin 함수에서 참조하는 계수에 대해 정렬을 적용하면 계수가 정렬된 후 누적 최소값이 계산됩니다.
- 재설정 차원 목록에 차원이 하나만 있는 경우에도 항상 차원을 괄호로 묶어야 합니다.
- 재설정 차원 집합을 지정하는 경우 세미콜론으로 구분해야 합니다.
- RunningMin은 블록 나누기 또는 새 섹션 추가 후에 자동으로 최소값을 다시 설정하지 않습니다.

## 예제

RunningMin([판매 수익])은 다음 표에 나와 있는 결과를 반환합니다.

국가	휴양지	판매 수익	누적 최소값
프랑스	French Riviera	835,420	835,420
미국	Bahamas Beach	971,444	835,420
미국	Hawaiian Club	1,479,660	835,420

RunningMin([판매 수익];([구/군/시])) 수식을 사용하여 [구/군/시]의 섹션에서 RunningMin을 사용하는 예제의 경우 다음 결과가 반환됩니다.

오스틴		
분기	판매 수익	누적 최소값

Q1	\$775,482.70	\$775,482.70
Q2	\$667,850.30	\$667,850.30
Q3	\$581,470.40	\$581,470.40
Q4	\$674,869.80	\$581,470.40
보스턴		
분기	판매 수익	누적 최소값
Q1	\$312,896.40	\$312,896.40
Q2	\$291,431.00	\$291,431.00
Q3	\$249,529.00	\$249,529.00
Q4	\$429,850.20	\$249,529.00

## 관련 정보

[IncludeEmpty 연산자 \[페이지 228\]](#)

[Row/Col 연산자 \[페이지 230\]](#)

## 6.1.2.19 RunningProduct

### 설명

계수의 누적 평균을 반환합니다.

### 함수 그룹

### 집계

### 구문

```
num RunningProduct (measure [; Row | Col] [; (reset_dims) ])
```

## 입력

매개 변수	설명	유형	필수
measure	임의의 계수	계수	예
Row Col	계산 방향을 설정합니다.	키워드	아니요
reset_dims	지정한 차원을 기준으로 계산을 재설정합니다.	차원 목록	아니요

## 참고

- RunningProduct와 함께 확장 구문 컨텍스트 연산자를 사용할 수 있습니다.
- Row 및 Col 연산자를 사용하여 계산 방향을 설정할 수 있습니다.
- RunningProduct 함수에서 참조하는 계수에 대해 정렬을 적용하면 계수가 정렬된 후 누적 곱이 계산됩니다.
- 재설정 차원 목록에 차원이 하나만 있는 경우에도 항상 차원을 괄호로 묶어야 합니다.
- 재설정 차원 집합을 지정하는 경우 세미콜론으로 구분해야 합니다.
- RunningProduct는 블록 나누기 또는 새 섹션 추가 후에 자동으로 곱을 다시 설정하지 않습니다.

## 예제

RunningProduct([Number of guests])는 다음 표에 나열된 결과를 반환합니다.

원산지	도시	고객 수	누적 곱
일본	고베	6	6
일본	오사카	4	24
미국	시카고	241	5,784

RunningProduct([고객 수];([원산지]))은 다음 표에 나열된 결과를 반환합니다.

원산지	도시	고객 수	누적 곱
일본	고베	6	6
일본	오사카	4	24
미국	시카고	241	5784

## 관련 정보

[IncludeEmpty 연산자 \[페이지 228\]](#)

[Row/Col 연산자 \[페이지 230\]](#)

## 6.1.2.20 RunningSum

### 설명

계수의 누적 합계를 반환합니다.

### 함수 그룹

### 집계

### 구문

```
num RunningSum(measure[;Row|Col][; (reset_dims)])
```

RunningSum의 각 섹션에서 재설정하려면 다음 구문을 사용하는 것이 좋습니다.

```
num RunningSum(measure;section)
```

### 입력

매개 변수	설명	유형	필수
measure	임의의 계수	계수	예
Row Col	계산 방향을 설정합니다.	키워드	아니요
reset_dims	지정한 차원을 기준으로 계산을 재설정합니다.	차원 목록	아니요
section	섹션이 설정되는 차원입니다.	키워드	섹션 재설정 시 예

### 참고

- RunningSum과 함께 확장 구문 컨텍스트 연산자를 사용할 수 있습니다.
- Row 및 Col 연산자를 사용하여 계산 방향을 설정할 수 있습니다.
- RunningSum 함수에서 참조하는 계수에 대해 정렬을 적용하면 계수가 정렬된 후 누적 합계가 계산됩니다.
- 재설정 차원 목록에 차원이 하나만 있는 경우에도 항상 차원을 괄호로 묶어야 합니다.
- 재설정 차원 집합을 지정하는 경우 세미콜론으로 구분해야 합니다.
- RunningSum은 블록 나누기 또는 새 섹션 추가 후에 자동으로 합계를 다시 설정하지 않습니다.

## 예제

`RunningSum([수익])`은 다음 표에 나열된 결과를 반환합니다.

국가	휴양지	수익	누적 합계
프랑스	French Riviera	835,420	835,420
미국	Bahamas Beach	971,444	1,806,864
미국	Hawaiian Club	1,479,660	3,286,524

`RunningSum([수익];([국가]))`는 다음 표에 나열된 결과를 반환합니다.

국가	휴양지	수익	누적 합계
프랑스	French Riviera	835,420	835,420
미국	Bahamas Beach	971,444	971,444
미국	Hawaiian Club	1,479,660	2,451,104

`RunningSum([판매 수익];([분기]))` 수식을 사용하여 [분기]의 섹션에서 `RunningSum`을 사용하는 예제의 경우 다음 결과가 반환됩니다.

Q1		
구/군/시	판매 수익	누적 합계
뉴욕	\$1,987,114.70	\$1,987,114.70
휴스턴	\$1,544,627.80	\$3,531,742.50
로스앤젤레스	\$1,129,177.60	\$4,660,920.10
Q2		
구/군/시	판매 수익	누적 합계
뉴욕	\$2,028,090.70	\$2,028,090.70
휴스턴	\$1,380,838.20	\$3,408,928.90
로스앤젤레스	\$980,405.30	\$4,389,334.20

## 관련 정보

[IncludeEmpty 연산자 \[페이지 228\]](#)

[Row/Col 연산자 \[페이지 230\]](#)

## 6.1.2.21 StdDev

### 설명

계수의 표준 편차를 반환합니다.

### 함수 그룹

### 집계

### 구문

```
num StdDev (measure)
```

### 입력

매개 변수	설명	유형	필수
measure	임의의 계수	계수	예

### 참고

표준 편차는 숫자 집합에서의 통계적 산포의 계수이며 다음 방법으로 계산됩니다.

- 숫자 집합의 평균을 찾습니다.
- 집합의 각 숫자에서 평균을 뺀 다음 그 차이를 제공합니다.
- 이러한 차이 제곱을 모두 더합니다.
- 이 합계를 (<집합의 숫자 수> - 1)로 나눕니다.
- 결과의 제곱근을 찾습니다.

### 예제

[계수]에 값 집합 (2, 4, 6, 8)이 있는 경우 StdDev ([계수])는 2.58을 반환합니다.



관련 정보

[Var \[페이지 75\]](#)

## 6.1.2.22 StdDevP

설명

계수의 모집단 표준 편차를 반환합니다.

함수 그룹

집계

구문

```
num StdDevP (measure)
```

입력

매개 변수	설명	유형	필수
measure	임의의 계수	계수	예

참고

모집단 표준 편차는 숫자 집합에서의 통계적 산포의 계수이며 다음 방법으로 계산됩니다.

- 숫자 집합의 평균을 찾습니다.
- 집합의 각 숫자에서 평균을 뺀 다음 그 차이를 제공합니다.
- 이러한 차이 제곱을 모두 더합니다.
- 이 합계를 (<집합의 숫자 수>)로 나눕니다.
- 결과의 제곱근을 찾습니다.

StdDevP 함수와 함께 확장 구문 컨텍스트 연산자를 사용할 수 있습니다.

## 예제

[계수]에 값 집합 (2, 4, 6, 8)이 있는 경우 `StdDevP([계수])`는 2.24을 반환합니다.

### 6.1.2.23 Sum

## 설명

계수의 합계를 반환합니다.

## 함수 그룹

## 집계

## 구문

```
num Sum(measure[;member_set])
```

## 입력

매개 변수	설명	유형	필수
measure	임의의 계수	계수	예
member_set	멤버의 집합	멤버 집합	아니요

## 참고

- Sum 함수와 함께 확장 구문 컨텍스트 연산자를 사용할 수 있습니다.
- member\_set를 포함시키면 Sum 함수는 멤버 집합의 모든 멤버에 대한 계수의 합계를 반환합니다.
- member\_set에는 세미콜론(;)으로 구분된 여러 집합을 포함시킬 수 있습니다.
- 멤버 집합 목록은 {}로 묶어야 합니다.
- 멤버 집합 표현식에 정확한 멤버나 노드가 지정되지 않은 경우, 참조된 계층구조가 테이블에 있어야 합니다. 그러면 멤버 집합 표현식이 테이블에 있는 계층구조의 현재 멤버를 참조합니다. 계층구조가 테이블에 없는 경우 함수가 #MULTIVALUE 메시지를 반환합니다.

- 필요한 집계를 쿼리에서 사용할 수 없는 경우 위임된 계수 집계는 #TOREFRESH를 반환합니다. 새로운 수준의 집계를 얻으려면 문서를 새로 고쳐야 합니다. 예를 들어, 필터 표시줄을 사용할 때 사용자가 "모든 값"을 선택하기에 앞서 어떤 값을 선택하는 경우와 그 반대로 선택한 값에 앞서 "모든 값"을 선택하는 경우 이 값이 반환됩니다.
- XIR2에서 XIR3로 마이그레이션 할 경우 XIR2 쿼리에 IN 및 WHERE 절이 있는 집계 함수는 다음과 같이 괄호를 사용하여 Sum 함수에 포함되어야 합니다.  
XIR2의 수식: =Sum([Measure] In ([Dim 1];[Dim 2])) In ([Dim 1]) Where ([Dim 3]="Constant")  
XIR3 이후 부터 선언을 수정: =Sum((([Measure] In ([Dim 1];[Dim 2])) In ([Dim 1]) Where ([Dim 3]="Constant"))
- 그룹에 대해 제공된 위임된 계수의 경우 로컬 집계(그룹화된 값의 계수 값 집계)가 필요하므로 #UNAVAILABLE를 반환합니다.  
위임된 계수의 로컬 집계를 "if then else" 수식 또는 그룹 값에 대해 강제로 수행하더라도 #MULTIVALUE 메시지가 계속 반환됩니다.

## 예제

Sales Revenue 계수에 값 2000, 3000, 4000 및 1000이 있는 경우 Sum([판매 수익])은 10000을 반환합니다.

[캘리포니아]가 [지역] 계층구조(국가 > 주 > 시)의 멤버인 경우 Sum([판매 수익]; {Descendants([지역]&[미국].[캘리포니아];1)})은 캘리포니아에 있는 모든 시의 총 판매 수익을 반환합니다.

## 6.1.2.24 Var

### 설명

계수의 분산을 반환합니다.

### 함수 그룹

### 집계

### 구문

```
num Var(measure)
```

## 입력

매개 변수	설명	유형	필수
measure	임의의 계수	계수	예

## 참고

분산은 숫자 집합에서의 통계적 산포의 계수로서, 다음 방법으로 계산됩니다.

- 숫자 집합의 평균을 찾습니다.
- 집합의 각 숫자에서 평균을 뺀 다음 그 차이를 제공합니다.
- 이러한 차이 제곱을 모두 더합니다.
- 이 합계를 (<집합의 숫자 수> - 1)로 나눕니다.

분산은 표준 편차의 제곱입니다.

Var 함수와 함께 확장 구문 컨텍스트 연산자를 사용할 수 있습니다.

## 예제

계수에 값 집합 {2, 4, 6, 8}이 있는 경우 Var([계수])는 6.67을 반환합니다.

## 관련 정보

[StdDev \[페이지 72\]](#)

## 6.1.2.25 VarP

### 설명

계수의 모집단 분산을 반환합니다.

### 함수 그룹

### 집계

## 구문

```
num VarP(measure)
```

## 입력

매개 변수	설명	유형	필수
measure	임의의 계수	계수	예

## 참고

모집단 분산은 숫자 집합에서의 통계적 산포의 계수로서 다음 방법으로 계산됩니다.

- 숫자 집합의 평균을 찾습니다.
- 집합의 각 숫자에서 평균을 뺀 다음 그 차이를 제공합니다.
- 이러한 차이 제곱을 모두 더합니다.
- 이 합계를 (<집합의 숫자 수>)로 나눕니다.

모집단 분산은 모집단 표준 편차의 제곱입니다.

VarP 함수와 함께 확장 구문 컨텍스트 연산자를 사용할 수 있습니다.

## 예제

제수에 값 집합 {2, 4, 6, 8}이 있는 경우 VarP([제수])는 5를 반환합니다.

## 관련 정보

[StdDevP \[페이지 73\]](#)

## 6.1.3 문자 함수

### 6.1.3.1 Asc

#### 설명

문자의 ASCII 값을 반환합니다.

#### 함수 그룹

문자

#### 구문

```
int Asc(string)
```

#### 입력

매개 변수	설명	유형	필수
string	임의의 문자열	문자열	예

#### 참고

string에 하나 이상의 문자가 포함되어 있는 경우 문자열의 첫 번째 문자에 대한 ASCII 값을 반환합니다.

#### 예제

Asc("A")는 65를 반환합니다.

Asc("ab")는 97을 반환합니다.

Asc([국가])는 [국가] 값이 "US"인 경우 85를 반환합니다.

## 6.1.3.2 Char

### 설명

ASCII 코드에 연결된 문자를 반환합니다.

### 함수 그룹

### 문자

### 구문

```
string Char(ascii_code)
```

### 입력

매개 변수	설명	유형	필수
ascii_code	ASCII 코드	숫자	예

### 참고

number가 소수인 경우 소수 부분은 무시됩니다.

### 예제

s

Char(123)은 "{"을 반환합니다.

### 6.1.3.3 Concatenation

#### 설명

두 개의 문자열을 연결(조인)합니다. 숫자가 있는 경우, 함수는 그들을 연결하는 대신 값의 합계를 계산합니다.

#### i 노트

입력 매개 변수 중 적어도 하나가 문자열인 경우 다른 모든 입력 매개 변수는 문자열로 변환됩니다.

#### 함수 그룹

#### 문자

#### 구문

```
string Concatenation(first_string;second_string)
```

#### 입력

매개 변수	설명	유형	필수
first_string	첫 번째 문자열	문자열 또는 숫자	예
second_string	첫 번째 문자열에 추가된 문자열	문자열 또는 숫자	예

#### 참고

'+' 연산자를 사용하여 문자열을 연결할 수도 있습니다.

"First " + "Second"는 "첫 번째 두 번째"를 반환합니다.

"First " + "Second" + " Third"는 "첫 번째 두 번째 세 번째"를 반환합니다.

Concatenation을 사용하면 집계 함수에 여러 차원을 포함시킬 수 있습니다. 예를 들면, Count([Sales Person]+[Quarter]+[Resort])는 구문 Count(<Sales Person>,<Quarter>,<Resort>)에 해당하며 이는 Desktop Intelligence에서 허용됩니다.



## 예

`Concatenation("첫 번째"; "두 번째")`는 "첫 번째 두 번째"를 반환합니다.

`Concatenation("첫 번째"; Concatenation("두 번째"; "세 번째"))`는 "첫 번째 두 번째 세 번째"를 반환합니다.

`[A]`가 숫자이고 `[A] = 1`이면 `Concatenation([A]; [A])`은 "2"를 반환합니다.

`[A]`가 문자열이고 `[A] = 1`이면 `Concatenation([A]; [A])`은 "11"을 반환합니다.

`[A]`가 문자열, `[B]`가 숫자, `[A] = 1`이고 `[B] = 2`이면 `Concatenation([A]; [B])`은 "12"를 반환합니다.

## 6.1.3.4 Fill

### 설명

문자열을 `n`번 반복하는 문자열을 작성합니다.

### 함수 그룹

#### 문자

### 구문

```
string Fill(repeating_string; num_repeats)
```

### 입력

매개 변수	설명	유형	필수
repeating_string	반복되는 문자열	문자열	예
num_repeats	반복 횟수	숫자	예

### 예제

`Fill("뉴욕"; 2)`는 "뉴욕 뉴욕"을 반환합니다.

## 6.1.3.5 FormatDate

### 설명

지정된 서식에 따라 날짜의 서식을 지정합니다.

### 함수 그룹

### 문자

### 구문

```
string FormatDate(date;format_string)
```

### 입력

매개 변수	설명	유형	필수
date	서식을 지정할 날짜	날짜	예
format_string	적용할 서식	문자열	예

### 참고

- 출력 형식은 셀에 적용된 날짜 형식에 따라 달라집니다.
- 색상 서식 지정 문자열(예: [Red], [Blue] 등)은 FormatDate 함수에 적용할 수 없습니다.

### 예제

FormatDate(CurrentDate();"dd/MM/yyyy")는 오늘 날짜가 2005년 12월 15일인 경우 "15/12/2005"를 반환합니다.

## 관련 정보

[사용자 지정 서식 \[페이지 36\]](#)

### 6.1.3.5.1 FormatDate 함수 관련 Format\_string 예제

format\_string용 FormatDate 구문에서 다음 표에 나와 있는 예제를 사용할 수 있습니다.

#### i 노트

Web Intelligence Rich Client 또는 Applet 인터페이스의 [숫자 형식](#) 대화 상자에서 이들 샘플을 찾을 수 있습니다. 단, BI LaunchPad 기본 설정의 선택한 제품 로캘에 따라 표시되는 샘플이 결정됩니다. 예를 들어 [영어](#)를 선택하면 "September 21, 2004" 샘플을 사용할 수 있습니다.

샘플	구문
Tuesday, September 21, 2004	dddd, 'mmmm d', 'yyyy
September 21, 2004	mmmm d', 'yyyy
Sep 21, 2004	mmm d', 'yyyy
9/21/04	M'/'d'/'yy
Sep 21, 2004 8:45:30 PM	mmm d', 'yyyy h':mm':ss a
9/21/04 8:45 PM	M'/'d'/'yy h':mm a
9/21/2004	M'/'d'/'yyyy
09/21/2004	MM'/'d'/'yyyy
9/21/04 8:45:30 PM	M'/'d'/'yy h':mm a
8:45:30 PM	h':mm':ss a
8:45 PM	h':mm a
20:45:30	HH':mm':ss
20h45	HH'h'mm

#### ➔ 팁

구문의 실제 텍스트가 패턴 기호로 잘못 인식되지 않도록 텍스트 앞뒤에 아포스트로피를 사용하는 것이 좋습니다. 예를 들어 위 표에 나와 있는 마지막 샘플에서는 "HH'h'mm"의 'h'가 실제 텍스트입니다.

## 관련 정보

[FormatDate \[페이지 82\]](#)

[사용자 지정 서식 \[페이지 36\]](#)

## 6.1.3.6 FormatNumber

### 설명

지정된 서식에 따라 숫자 서식을 지정합니다.

### 함수 그룹

### 문자

### 구문

```
string FormatNumber(number;format_string)
```

### 입력

매개 변수	설명	유형	필수
숫자	서식을 지정할 숫자	숫자	예
format_string	적용할 서식	문자열	예

### 참고

- 출력 형식은 셀에 적용된 숫자 서식에 따라 달라집니다.
- 색상 서식 지정 문자열(예: [Red], [Blue] 등)은 FormatNumber에 적용할 수 없습니다.

### 예

FormatNumber([수익];"#,##.00")은 [수익]이 835,420인 경우 835,420.00을 반환합니다.

### 관련 정보

[사용자 지정 서식 \[페이지 36\]](#)

### 6.1.3.7 HTMLEncode

#### 설명

문자열에 HTML 인코딩 규칙을 적용합니다.

#### 함수 그룹

문자

#### 구문

```
string HTMLEncode(html)
```

#### 입력

매개 변수	설명	유형	필수
html	HTML 문자열	문자열	예

#### 예제

`HTMLEncode("http://www.sap.com")` 는 "http%3A%2F%2Fwww%2Esap%2Ecom"을 반환합니다.

### 6.1.3.8 InitCap

#### 설명

문자열의 첫 번째 문자를 대문자로 표시합니다.

#### 함수 그룹

문자

## 구문

```
string InitCap(string)
```

## 입력

매개 변수	설명	유형	필수
string	대문자로 표시할 문자열	문자열	예

## 예제

InitCap("we hold these truths to be self-evident") 는 "We hold these truths to be self-evident"를 반환합니다.

## 6.1.3.9 Left

### 설명

문자열의 가장 왼쪽에 있는 문자를 반환합니다.

#### **i** 노트

선택한 인터페이스 로컬이 아랍어(오른쪽에서 왼쪽으로 쓰기/읽기)인 경우 이 함수는 문자열의 논리적 시작부터 첫 번째 문자를 반환합니다.

## 함수 그룹

### 문자

## 구문

```
string Left(string; num_chars)
```

## 입력

매개 변수	설명	유형	필수
string	입력 문자열	문자열	예
num_chars	문자열의 시작부터 반환할 문자의 수	숫자	예

## 예제

`Left([국가]; 2)`는 [국가]가 "France"인 경우 "Fr"을 반환합니다.

### 6.1.3.10 LeftPad

## 설명

다른 문자열로 문자열의 왼쪽을 채웁니다.

#### **i** 노트

선택한 인터페이스 로캘이 아랍어(오른쪽에서 왼쪽으로 쓰기/읽기)인 경우 이 함수는 다른 문자열 문자의 논리적 시작 앞에 문자열을 채웁니다.

## 함수 그룹

## 문자

## 구문

```
string LeftPad(padded_string;length;left_string)
```

## 입력

매개 변수	설명	유형	필수
padded_string	원래 문자열	문자열	예
길이	출력 문자열의 길이	숫자	예
left_string	padded_string의 시작에 추가할 문자열	문자열	예

## 참고

- length가 left\_string과 padded\_string을 합친 길이보다 짧은 경우 left\_string이 잘립니다.
- length가 padded\_string의 길이보다 짧거나 같은 경우 padded\_string이 반환됩니다.
- length가 padded\_string과 left\_string을 합친 길이보다 긴 경우 합친 길이와 동일하게 될 때까지 left\_string이 전체 또는 부분적으로 반복됩니다.

## 예제

LeftPad("York";8;"New ")는 "New York"을 반환합니다.

LeftPad("York";6;"New ")는 "NeYork"을 반환합니다.

LeftPad("York";11;"New ")는 "New NewYork"을 반환합니다.

LeftPad("New ";2;"York")는 "New"를 반환합니다.

### 6.1.3.11 LeftTrim

## 설명

문자열의 선행 공백을 제거합니다.

#### **i** 노트

선택한 인터페이스 로캘이 아랍어(오른쪽에서 왼쪽으로 쓰기/읽기)인 경우 이 함수는 문자열의 논리적 시작부터 첫 번째 공백 문자를 제거합니다.



함수 그룹

문자

구문

```
string LeftTrim(trimmed_string)
```

입력

매개 변수	설명	유형	필수
trimmed_string	트리밍할 문자열	문자열	예

예제

LeftTrim([국가])는 [국가]가 "France"인 경우 "France"를 반환합니다.

## 6.1.3.12 Length

설명

문자열의 문자 수를 반환합니다.

함수 그룹

문자

구문

```
int Length(string)
```

## 입력

매개 변수	설명	유형	필수
string	입력 문자열	문자열	예

## 예제

`Length ([Last Name])`은 `[Last Name]`이 "Smith"인 경우 5를 반환합니다.

## 6.1.3.13 Lower

## 설명

문자열을 소문자로 변환합니다.

## 함수 그룹

문자

## 구문

```
string Lower(string)
```

## 입력

매개 변수	설명	유형	필수
string	소문자로 변환할 문자열입니다.	문자열	예

## 예제

`Lower ("New York")`은 "new york"을 반환합니다.

## 6.1.3.14 Match

### 설명

문자열이 패턴과 일치하는지 확인합니다.

### 함수 그룹

문자

### 구문

```
bool Match(test_string;pattern)
```

### 입력

매개 변수	설명	유형	필수
test_string	텍스트 패턴 일치 여부를 확인할 문자열입니다.	문자열	예
pattern	텍스트 패턴	문자열	예

### 참고

- 패턴은 와일드카드를 포함할 수 있습니다. "\*"는 문자 집합을 대체하고 "?"는 단일 문자를 대체합니다.

### 예제

Match([국가];"F\*")는 [국가]가 "France"인 경우 True를 반환합니다.

Match([국가];"?S?")는 [국가]가 "USA"인 경우 True를 반환합니다.

Match("New York";"P\*")는 False를 반환합니다.

## 6.1.3.15 Pos

### 설명

문자열에서 텍스트 패턴의 시작 위치를 반환합니다.

### 함수 그룹

### 문자

### 구문

```
int Pos(test_string;pattern)
```

### 입력

매개 변수	설명	유형	필수
test_string	텍스트 패턴을 확인할 문자열	string	예
pattern	텍스트 패턴	문자열	예

### 참고

- 패턴이 두 번 이상 발생하면 Pos는 처음으로 발생하는 위치를 반환합니다.

### 예제

Pos("New York"; "Ne")는 1을 반환합니다.

Pos("New York, New York"; "Ne")는 1을 반환합니다.

Pos("New York"; "York")는 5를 반환합니다.

## 6.1.3.16 Replace

### 설명

문자열의 일부를 다른 문자열로 대체합니다.

### 함수 그룹

문자

### 구문

```
string Replace(replace_in;replaced_string;replace_with)
```

### 입력

매개 변수	설명	유형	필수
replace_in	텍스트가 대체되는 문자열입니다.	string	예
replaced_string	대체되는 텍스트입니다.	string	예
replace_with	replaced_string을 대체하는 텍스트입니다.	string	예

### 예제

`Replace("New YORK";"ORK";"ork")`는 "New York"을 반환합니다.

## 6.1.3.17 Right

### 설명

문자열의 가장 오른쪽 문자(문자열의 끝에 있는 문자)를 반환합니다.

### i 노트

선택한 인터페이스 로캘이 아랍어(오른쪽에서 왼쪽으로 표시/읽기)이면 이 함수는 문자열의 논리적 시작에서 첫 번째 문자를 반환합니다.

## 함수 그룹

문자

## 구문

```
string Right(string; num_chars)
```

## 입력

매개 변수	설명	유형	필수
string	임의의 문자열	string	예
num_chars	오른쪽부터 반환할 문자의 수	숫자	예

## 예제

`Right([국가]; 2)`는 [국가]가 "France"인 경우 "ce"를 반환합니다.

## 6.1.3.18 RightPad

## 설명

다른 문자열로 문자열의 오른쪽을 채웁니다(원래 문자열의 시작에 문자열을 추가).

### i 노트

선택한 인터페이스 로캘이 아랍어(오른쪽에서 왼쪽으로 표시/읽기)이면 이 함수는 문자열의 논리적 시작에서 첫 번째 문자에 문자열을 추가합니다.

## 함수 그룹

### 문자

### 구문

```
string RightPad(padded_string;length;right_string)
```

### 입력

매개 변수	설명	유형	필수
padded_string	원래 문자열	문자열	예
길이	출력 문자열의 길이	숫자	예
right_string	padded_string의 끝에 추가할 문자열	문자열	예

### 참고

- length가 right\_string과 padded\_string을 합친 길이보다 짧은 경우 right\_string이 잘립니다.
- length가 padded\_string의 길이보다 짧거나 같은 경우 padded\_string이 반환됩니다.
- length가 padded\_string과 right\_string을 합친 길이보다 긴 경우 합친 길이와 동일하게 될 때까지 right\_string이 전체/부분적으로 반복됩니다.

### 예제

RightPad("New ";8;"York")는 "New York"을 반환합니다.

RightPad("New "; 6;"York")는 "New Yo"를 반환합니다.

RightPad("New ";11;"York")는 "New YorkYor"를 반환합니다.

RightPad("New ";2;"York")는 "New"를 반환합니다.

### 6.1.3.19 RightTrim

#### 설명

문자열의 후행 공백을 제거합니다.

#### i 노트

선택한 인터페이스 로캘이 아랍어(오른쪽에서 왼쪽으로 표시/읽기)이면 이 함수는 문자열의 논리적 끝에서 후행 공백을 제거합니다.

#### 함수 그룹

문자

#### 구문

```
string RightTrim(trimmed_string)
```

#### 입력

매개 변수	설명	유형	필수
trimmed_string	트리밍할 문자열	문자열	예

#### 예제

`RightTrim([국가])`는 [국가]가 "France "인 경우 "France"를 반환합니다.

### 6.1.3.20 Substr

#### 설명

문자열의 일부를 반환합니다.



## 함수 그룹

### 문자

## 구문

```
string SubStr(string;start;length)
```

## 입력

매개 변수	설명	유형	필수
string	임의의 문자열	문자열	예
start	추출할 문자열의 시작 위치	숫자	예
길이	추출할 문자열의 길이	숫자	예

## 예제

SubStr("Great Britain";1;2)는 "Great"을 반환합니다.

SubStr("Great Britain";3;2)는 "Britain"을 반환합니다.

## 6.1.3.21 Trim

### 설명

문자열의 선행 및 후행 공백을 제거합니다.

## 함수 그룹

### 문자

## 구문

```
string Trim(trimmed_string)
```

## 입력

매개 변수	설명	유형	필수
string	트리밍할 문자열	문자열	예

## 예제

Trim(" Great Britain ")은 "Great Britain"을 반환합니다.

## 6.1.3.22 Upper

### 설명

문자열을 대문자로 변환합니다.

### 함수 그룹

#### 문자

## 구문

```
string Upper(string)
```

## 입력

매개 변수	설명	유형	필수
string	변환할 문자열	문자열	예

## 예제

`Upper("New York")`는 "NEW YORK"을 반환합니다.

## 6.1.3.23 UriEncode

## 설명

문자열에 URL 인코딩 규칙을 적용합니다.

## 함수 그룹

## 문자

## 구문

```
string UriEncode(html)
```

## 입력

매개 변수	설명	유형	필수
html	인코딩할 URL	문자열	예

## 예제

`UriEncode("http://www.sap.com")`는 "http%3A%2F%2Fwww%2Esap%2Ecom"을 반환합니다.

### 6.1.3.24 첫글자대문자

#### 설명

문자열에 있는 모든 단어의 첫 글자를 대문자화합니다.

#### 함수 그룹

문자

#### 구문

```
string WordCap(string)
```

#### 입력

매개 변수	설명	유형	필수
string	대문자화할 문자열	문자열	예

#### 예제

WordCap("Sales revenue for March")는 "Sales Revenue For March"를 반환합니다.

## 6.1.4 날짜 및 시간 함수

### 6.1.4.1 CurrentDate

#### 설명

지역 설정에 따라 서식 설정된 현재 날짜를 반환합니다.

---

함수 그룹

날짜 및 시간

구문

```
date CurrentDate()
```

예제

CurrentDate()는 오늘 날짜가 2002년 9월 10일인 경우 2002년 9월 10일을 반환합니다.

## 6.1.4.2 CurrentTime

설명

지역 설정에 따라 서식 설정된 현재 시간을 반환합니다.

함수 그룹

날짜 및 시간

구문

```
time CurrentTime()
```

예제

CurrentTime은 현재 시간이 11시 15분인 경우 11:15를 반환합니다.

### 6.1.4.3 DayName

#### 설명

날짜의 요일 이름을 반환합니다.

#### 함수 그룹

날짜 및 시간

#### 구문

```
string DayName (date)
```

#### 입력

매개 변수	설명	유형	필수
date	입력 날짜	날짜	예

#### 예제

DayName ([예약 날짜]) 는 [예약 날짜]의 날짜가 2001년 12월 15일(토요일)인 경우 "토요일"을 반환합니다.

#### 참고

입력 날짜는 변수여야 합니다. DayName ("07/15/2001") 과 같이 날짜를 직접 지정할 수 없습니다.

### 6.1.4.4 DayNumberOfMonth

#### 설명

일을 월 기준 숫자 값으로 반환합니다.

함수 그룹

날짜 및 시간

구문

```
int DayNumberOfMonth (date)
```

입력

매개 변수	설명	유형	필수
date	입력 날짜	날짜	예

예제

DayNumberOfMonth ([예약 날짜]) 는 [예약 날짜]의 날짜가 2001년 12월 15일인 경우 15를 반환합니다.

## 6.1.4.5 DayNumberOfWeek

설명

일을 주 기준 숫자 값으로 반환합니다.

함수 그룹

날짜 및 시간

구문

```
int DayNumberOfWeek (date)
```

## 입력

매개 변수	설명	유형	필수
date	입력 날짜	날짜	예

## 참고

이 함수는 월요일을 한 주의 첫 번째 날로 처리합니다.

## 예

`DayNumberOfWeek([예약 날짜])`는 [예약 날짜]의 날짜가 2005년 5월 2일(월요일)인 경우 1을 반환합니다.

## 6.1.4.6 DayNumberOfYear

### 설명

일을 연 기준 숫자 값으로 반환합니다

### 함수 그룹

날짜 및 시간

## 구문

```
int DayNumberOfYear(date)
```



## 입력

매개 변수	설명	유형	필수
date	입력 날짜	날짜	예

## 예제

DayNumberOfYear ([예약 날짜]) 는 [예약 날짜]의 날짜가 2001년 12월 15일인 경우 349를 반환합니다.

## 6.1.4.7 DaysBetween

### 설명

지정된 날짜 사이의 일수를 반환합니다.

### 함수 그룹

날짜 및 시간

### 구문

```
int DaysBetween(first_date;last_date)
```

#### **i** 노트

인수에 지정된 날짜가 동일한 시간대에 있는지 확인해야 합니다. 이는 모든 날짜 작업(비교, 계산 등)에 적용됩니다.

## 입력

매개 변수	설명	유형	필수
first_date	첫 번째 날짜	날짜	예
last_date	마지막 날짜	날짜	예

## 예제

`DaysBetween([판매 날짜],[송장 날짜])`는 [판매 날짜]가 2001년 12월 15일이고 [송장 날짜]가 2001년 12월 17일인 경우 2를 반환합니다.

## 6.1.4.8 LastDayOfMonth

### 설명

월의 마지막 날짜를 반환합니다.

### 함수 그룹

날짜 및 시간

### 구문

```
date LastDayOfMonth(date)
```

### 입력

매개 변수	설명	유형	필수
date	월의 임의 날짜	날짜	예

## 예제

`LastDayOfMonth([판매 날짜])`는 [판매 날짜]가 2005년 12월 11일인 경우 2005년 12월 31일을 반환합니다.

## 6.1.4.9 LastDayOfWeek

### 설명

주의 마지막 날짜를 반환합니다.

### 함수 그룹

날짜 및 시간

### 구문

```
date LastDayOfWeek (date)
```

### 입력

매개 변수	설명	유형	필수
date	주의 임의 날짜	날짜	예

### 참고

이 함수는 월요일을 한 주의 첫 번째 날로 처리합니다.

### 예

`LastDayOfWeek ([판매 날짜])`는 [판매 날짜]가 2005년 5월 11일인 경우 2005년 5월 15일(일요일)을 반환합니다.

## 6.1.4.10 Month

### 설명

날짜의 월 이름을 반환합니다.

## 함수 그룹

날짜 및 시간

## 구문

```
string Month(date)
```

## 입력

매개 변수	설명	유형	필수
date	입력 날짜	날짜	예

## 예제

`Month([예약 날짜])`는 [예약 날짜]의 날짜가 2005년 12월 15일인 경우 "12월"을 반환합니다.

## 6.1.4.11 MonthNumberOfYear

## 설명

날짜의 달 숫자를 반환합니다.

## 함수 그룹

날짜 및 시간

## 구문

```
int MonthNumberOfYear(date)
```

## 입력

매개 변수	설명	유형	필수
date	연중 임의의 날짜	날짜	예

## 예제

MonthNumberOfYear([예약 날짜])는 [예약 날짜]의 날짜가 2005년 12월 15일인 경우 12를 반환합니다.

## 6.1.4.12 MonthsBetween

### 설명

두 날짜 사이의 개월 수를 반환합니다.

### 함수 그룹

날짜 및 시간

### 구문

```
int MonthsBetween(first_date;last_date)
```

## 입력

매개 변수	설명	유형	필수
first_date	첫 번째 날짜	날짜	예
last_date	마지막 날짜	날짜	예

## 예제

MonthsBetween([판매 날짜], [송장 날짜])는 [판매 날짜]의 날짜가 2005년 12월 2일이고 [송장 날짜]가 2006년 1월 2일인 경우 1을 반환합니다.

MonthsBetween([판매 날짜], [송장 날짜])는 [판매 날짜]가 31/03/2008일이고 [송장 날짜]가 30/04/2008일인 경우 1을 반환합니다.

MonthsBetween([판매 날짜], [송장 날짜])는 [판매 날짜]가 07/01/1993일이고 [송장 날짜]가 06/11/2002일인 경우 118을 반환합니다.

## 6.1.4.13 Quarter

### 설명

날짜의 분기 숫자를 반환합니다.

### 함수 그룹

날짜 및 시간

### 구문

```
int Quarter(date)
```

### 입력

매개 변수	설명	유형	필수
date	분기중 임의의 날짜	날짜	예

### 예제

Quarter([예약 날짜])는 [예약 날짜]의 날짜가 2005년 12월 15일인 경우 4를 반환합니다.

## 6.1.4.14 RelativeDate

### 설명

다른 날짜에 대한 상대적인 날짜를 반환합니다.

### 함수 그룹

날짜 및 시간

### 구문

```
날짜 RelativeDate (start_date; num; period)
```

### 입력

매개 변수	설명	유형	필수
start_date	시작 날짜	날짜	예
num	시작 날짜에 추가되는 기간 단위 수	숫자	예
period	시작 날짜에 추가되는 단위의 유형	미리 정의됨	선택 사항

### 참고

- num 매개 변수는 상수, 함수의 결과(수치 값), 계수 값 또는 차원 값(수치)이 될 수 있으며 정수여야 합니다.
- num 매개 변수는 음수를 사용하여 start\_date보다 이른 날짜를 반환할 수 있습니다.
- period 매개 변수 값이 생략되면 일수 (DayPeriod)가 값으로 사용됩니다.
- 월을 더하거나 뺄 때(SemesterPeriod, QuarterPeriod 및 MonthPeriod에 대해) 반환된 월에 일자가 없으면 반환된 월의 마지막 날이 사용되어야 합니다.
- period 매개 변수에 사용할 수 있는 값은 MillisecondPeriod, SecondPeriod, MinutePeriod, HourPeriod, DayPeriod, WeekPeriod, MonthPeriod, QuarterPeriod, SemesterPeriod, YearPeriod입니다.

예

`RelativeDate([예약 날짜]; 2)`는 [예약 날짜]가 2005년 12월 15일인 경우 2005년 12월 17일을 반환합니다.

`RelativeDate([예약 날짜]; -3)`는 [예약 날짜]가 2007년 1월 12일인 경우 2007년 1월 9일을 반환합니다.

`RelativeDate([예약 날짜]; 1; MonthPeriod)`는 [예약 날짜]가 2007년 1월 12일인 경우 2007년 2월 12일을 반환합니다.

## 6.1.4.15 TimeDim

설명

`TimeDim` 시간 차원을 사용하면 날짜 유형 유니버스 개체에서 시간 축을 작성할 수 있습니다. `TimeDim`은 두 번째 매개 변수로 주어진 기간 동안 첫 번째 매개 변수로 제공된 날짜에 대한 데이터를 반환합니다. 데이터가 없는 기간이 있을 경우 비어있는 각 기간의 첫 번째 날짜가 반환됨으로써 주어진 기간에 대한 전체 축을 확보할 수 있습니다. 이를 통해 다음과 같은 사항이 보장됩니다.

- 축의 시간 순서가 유지됩니다(가장 오래된 개체부터 가장 최근의 개체 순).
- 현재 컨텍스트의 최소 날짜와 최대 날짜 사이의 모든 기간이 축에 포함됩니다.

### i 노트

`TimeDim` 함수로는 수식(예: 필터, 입력 컨트롤, 요소 링크, 필터 표시줄/드릴 도구 모음)을 필터링할 수 없습니다. 대신 기본 날짜 차원을 직접 필터링해야 합니다.

함수 그룹

날짜 및 시간

구문

```
TimeDim([Date Type]; Period Type)
```



## 입력

매개 변수	설명	유형	필수
날짜 형식	보고서의 날짜 개체(예: InvoiceDate)	날짜	예
기간 유형	<p>다음 값에서 얻어지는 결과에 대한 기간</p> <ul style="list-style-type: none"> <li>DayPeriod</li> <li>MonthPeriod</li> <li>QuarterPeriod</li> <li>YearPeriod</li> </ul> <p>값을 선택하지 않으면 기본적으로 DayPeriod가 사용됩니다. 이 개체는 데이터 공급자 개체여야 하고, 보고서 개체에서 사용 가능해야 하며, 변수가 될 수 없습니다.</p>	미리 정의됨	선택 사항

위 함수를 다음 함수와 함께 사용합니다.

- DayName
- DayNumberOfMonth
- DayNumberOfWeek
- DayNumberOfYear
- Month
- MonthNumberOfYear
- Quarter
- Year
- FormatDate

## 예

아래 첫 번째 표에는 특정 날짜에만 해당되는 데이터가 포함되어 있으며, 그 아래의 쿼리 예제를 통해 결과가 해석되는 방식을 볼 수 있습니다.

송장 날짜	수익
2000-01-03	31,607
2000-01-08	31,244
2000-07-03	38,154

DayName(TimeDim([Invoice Date] ; QuarterPeriod) 수식은 위 표의 일별 값을 반환합니다.

송장 날짜	수익
2000-01-03	31,607
2000-01-08	31,244
2000-04-01	
2000-07-03	38,154

TimeDim 함수의 결과를 Quarter 함수 형식으로 표현하여 결과를 Quarter(Q1, Q2...) 형식으로 반환함으로써 다음 결과 테이블을 얻을 수 있습니다.

송장 날짜	수익
Q1	62,851
Q2	
Q3	38,154

## 6.1.4.16 ToDate

### 설명

문자열을 날짜로 변환합니다. 날짜 형식을 매개 변수로 제공할 경우 문자열을 날짜로 변환하는 방법을 Web Intelligence에 표시합니다. 제공하는 날짜 형식은 원래 문자열의 날짜 형식과 일치해야 합니다. 가능한 날짜 형식은 아래의 링크를 참조하십시오.

### 함수 그룹

#### 날짜 및 시간

### 구문

```
date ToDate(date_string;format)
```

또는

```
date ToDate(date_string;"INPUT_DATE_TIME")
```

#### i 노트

사용자에 따라 [기본 설정 보기 로컬](#)이 달라질 수 있는 경우에는 특정 로컬을 위한 고정 형식은 적합하지 않습니다. 이 경우 위의 두 번째 예에 표시된 것과 같이 INPUT\_DATE\_TIME을 사용하십시오.

## 입력

매개 변수	설명	유형	필수
date_string	날짜로 해석할 문자	문자열	예
format	문자열에서 사용할 날짜 형식  기본 설정 보기 로케의 형식을 사용하려면 "INPUT_DATE_TIME"을 사 용하십시오.	문자열	예*

\* 위의 참고를 참조하십시오. 필요에 따라 format 또는 INPUT\_DATE\_TIME을 사용하십시오.

## 예제

`ToDate("12/15/2002"; "MM/dd/yyyy")`는 "12"를 월로, "15"를 일로, "2002"를 연도로 해석합니다.

`ToDate("Dec/02"; "Mmm/yy")`는 "Dec"를 월로, "02"을 연도의 마지막 두 자리로 해석합니다.

`ToDate("15-December-02"; "dd-Mmmm-yy")`는 "15"를 일로, "December"를 월로, "02"를 연도의 마지막 두 자리로 해석합니다.

`ToDate("12/15/02 11:00:00"; "INPUT_DATE_TIME")`는 사용자 컴퓨터의 기본 설정 보기 로케에 사용된 형식으로 "12/15/02 11:00:00"을 해석합니다.

### i 노트

- INPUT\_DATE\_TIME를 사용하면 date\_string 입력 문자열에 날짜와 시간을 모두 지정해야 합니다.
- date\_string을 지정한 형식의 날짜로 해석할 수 없는 경우에는 ToDate()가 #ERROR를 반환합니다.
- 셀에 날짜가 표시되는 방식은 해당 셀에서 선택한 날짜 형식에 따라 결정됩니다. 예를 들어 선택한 날짜 형식이 "MM/dd/yyyy"이면 `ToDate("Dec/15/02"; "MMM/dd/yy")`는 12/15/2002로 표시됩니다.

## 관련 정보

[사용자 지정 서식 \[페이지 36\]](#)

## 6.1.4.17 주

### 설명

연중 몇 번째 주인지를 숫자로 반환합니다.

## 함수 그룹

날짜 및 시간

## 구문

```
int Week(date)
```

## 입력

매개 변수	설명	유형	필수
date	입력 날짜	날짜	예

## 예제

Week([예약 날짜])는 [예약 날짜]의 날짜가 2004년 1월 4일(2004년의 첫 번째 주에 해당)인 경우 1을 반환합니다.

## 6.1.4.18 연도

## 설명

해당 날짜에 해당하는 연도를 반환합니다.

## 함수 그룹

날짜 및 시간

## 구문

```
int Year(date)
```

## 입력

매개 변수	설명	유형	필수
date	입력 날짜	날짜	예

## 예제

`Year([예약 날짜])`는 [예약 날짜]의 날짜가 2005년 12월 15일인 경우 2005를 반환합니다.

## 6.1.5 데이터 공급자 함수

### 6.1.5.1 Connection

## 설명

데이터 공급자에서 사용된 데이터베이스 연결 매개 변수를 반환합니다.

## 함수 그룹

데이터 공급자

## 구문

```
string Connection(dp)
```

## 입력

매개 변수	설명	유형	필수
dp	데이터 공급자	데이터 공급자	예

## 참고

- 데이터 공급자의 이름은 대괄호로 묶어야 합니다.
- 보안상 이유로 함수 출력에 데이터베이스 호스트 이름, 사용자 이름 및 사용자 암호가 포함되지 않습니다.

## 6.1.5.2 DataProvider

### 설명

보고서 개체가 포함된 데이터 공급자 이름을 반환합니다.

### 함수 그룹

데이터 공급자

### 구문

```
string DataProvider(obj)
```

### 입력

매개 변수	설명	유형	필수
obj	보고서 개체	보고서 개체	예

### 예제

[총 수익] 계수가 "판매"라는 데이터 공급자에 포함되어 있는 경우, `DataProvider([총 수익])`은 "판매"를 반환합니다.

#### **i** 노트

DataProvider가 해당 데이터 공급자 이름을 반환하려면 개체 이름이 필요합니다. DataProvider의 매개 변수로 다른 함수를 사용할 경우(예: 차원 변수), 개체 이름이 제공되지 않고 DataProvider 함수를 통해 오류가 반환됩니다.

### 6.1.5.3 DataProviderKeyDate

#### 설명

데이터 공급자의 keydate를 반환합니다.

#### 함수 그룹

데이터 공급자

#### 구문

```
date DataProviderKeyDate(dp)
```

#### 입력

매개 변수	설명	유형	필수
dp	데이터 공급자	데이터 공급자	예

#### 참고

- 데이터 공급자의 이름은 대괄호로 묶어야 합니다.
- 문서 로컬에 따라 반환된 keydate의 서식이 지정됩니다.

#### 예제

`DataProviderKeyDate([판매])`는 판매 데이터 공급자의 keydate가 2007년 8월 3일인 경우 2007년 8월 3일을 반환합니다.

## 6.1.5.4 DataProviderKeyDateCaption

### 설명

데이터 공급자의 keydate 캡션을 반환합니다.

### 함수 그룹

데이터 공급자

### 구문

```
string DataProviderKeyDateCaption(dp)
```

### 입력

매개 변수	설명	유형	필수
dp	데이터 공급자	데이터 공급자	예

### 참고

데이터 공급자의 이름은 대괄호로 묶어야 합니다.

### 예제

`DataProviderKeyDateCaption([판매])`는 판매 데이터 공급자의 keydate 캡션이 "현재 달력 날짜"인 경우 "현재 달력 날짜"를 반환합니다.



## 6.1.5.5 DataProviderSQL

### 설명

데이터 공급자에서 생성된 SQL을 반환합니다.

### 함수 그룹

데이터 공급자

### 구문

```
string DataProviderSQL(dp)
```

### 입력

매개 변수	설명	유형	필수
dp	데이터 공급자	데이터 공급자	예

### 참고

데이터 공급자의 이름은 대괄호로 묶어야 합니다.

### 예제

`DataProviderSQL([Query 1])`은 데이터 공급자 SQL이 `SELECT country.country_name FROM country`인 경우 `SELECT country.country_name FROM country`를 반환합니다.

## 6.1.5.6 DataProviderType

### 설명

데이터 공급자의 유형을 반환합니다.

### 함수 그룹

데이터 공급자

### 구문

```
string DataProviderType(dp)
```

### 입력

매개 변수	설명	유형	필수
dp	데이터 공급자	데이터 공급자	예

### 참고

- DataProviderType은 유니버스 데이터 공급자에 대해서는 "유니버스"를, 개인 데이터 공급자에 대해서는 "개인 데이터"를 반환합니다.
- 데이터 공급자의 이름은 대괄호로 묶어야 합니다.

### 예

DataProviderType([판매])는 "판매" 데이터 공급자가 유니버스를 기반으로 할 경우 "Universe"를 반환합니다.

## 6.1.5.7 IsPromptAnswered

### 설명

프롬프트가 응답되었는지 여부를 결정합니다.

### 함수 그룹

데이터 공급자

### 구문

```
bool IsPromptAnswered([dp;]prompt_string)
```

### 입력

매개 변수	설명	유형	필수
dp	프롬프트를 포함하는 데이터 공급자	데이터 공급자	아니요
prompt_string	프롬프트 텍스트	문자열	예

### 참고

- 데이터 공급자의 이름은 대괄호로 묶어야 합니다.
- IsPromptAnswered는 If 함수에 사용할 수 있는 부울 값을 반환합니다.
- IsPromptAnswered를 열에 직접 배치할 경우 정수를 반환합니다(1=true, 0=false). 부울 숫자 형식을 사용하여 이 정수에 서식을 지정할 수 있습니다.

### 예

IsPromptAnswered("Choose a city")는 "Choose a city" 텍스트로 식별된 프롬프트가 응답된 경우 true를 반환합니다.

IsPromptAnswered([Sales]; "Choose a city")는 [Sales] 데이터 공급자에서 "Choose a city" 텍스트로 식별된 프롬프트에 응답된 경우 true를 반환합니다.

## 6.1.5.8 LastExecutionDate

### 설명

데이터 공급자를 마지막으로 새로 고친 날짜를 반환합니다.

### 함수 그룹

데이터 공급자

### 구문

```
date LastExecutionDate (dp)
```

### 입력

매개 변수	설명	유형	필수
dp	데이터 공급자	데이터 공급자	예

### 참고

- 보고서에 데이터 공급자가 하나밖에 없는 경우에는 dp 매개 변수를 생략할 수 있습니다.
- 데이터 공급자의 이름은 대괄호로 묶어야 합니다.
- `DataProvider` 함수를 사용하여 데이터 공급자에 대한 참조를 제공할 수 있습니다.

### 예제

판매 쿼리 데이터 공급자를 2002년 3월 4일에 마지막으로 새로 고친 경우, `LastExecutionDate([Sales Query])`는 "2002/3/4"를 반환합니다.

관련 정보

[DataProvider](#) [페이지 118]

## 6.1.5.9 LastExecutionDuration

설명

데이터 공급자를 마지막으로 새로 고치는 데 걸린 시간을 초 단위로 반환합니다.

함수 그룹

데이터 공급자

구문

```
num LastExecutionDuration(dp)
```

입력

매개 변수	설명	유형	필수
dp	데이터 공급자	데이터 공급자	예

참고

데이터 공급자의 이름은 대괄호로 묶어야 합니다.

예제

`LastExecutionDuration([Sales])`는 "판매" 데이터 공급자가 마지막으로 실행되었을 때 데이터를 반환하는 데 3초 걸린 경우 3을 반환합니다.

## 6.1.5.10 LastExecutionTime

### 설명

데이터 공급자를 마지막으로 새로 고침 시간을 반환합니다.

### 함수 그룹

데이터 공급자

### 구문

```
time LastExecutionTime(dp)
```

### 입력

매개 변수	설명	유형	필수
dp	데이터 공급자	데이터 공급자	예

### 참고

- 보고서에 데이터 공급자가 하나밖에 없는 경우에는 dp 매개 변수를 생략할 수 있습니다.
- `DataProvider` 함수를 사용하여 데이터 공급자에 대한 참조를 제공할 수 있습니다.
- 데이터 공급자의 이름은 대괄호로 묶어야 합니다.

### 예제

`LastExecutionTime([Sales Query])`는 판매 쿼리 데이터 공급자를 2:48:00 PM에 마지막으로 새로 고침 경우 "2:48:00 PM"을 반환합니다.

---

관련 정보

[DataProvider](#) [페이지 118]

### 6.1.5.11 NumberOfDataProviders

설명

보고서의 데이터 공급자 수를 반환합니다.

함수 그룹

데이터 공급자

구문

```
int NumberOfDataProviders()
```

예제

보고서에 두 개의 데이터 공급자가 있는 경우, `NumberOfDataProviders()` 는 2를 반환합니다.

### 6.1.5.12 NumberOfRows

설명

데이터 공급자의 행 수를 반환합니다.

함수 그룹

데이터 공급자

## 구문

```
int NumberOfRows (dp)
```

## 입력

매개 변수	설명	유형	필수
dp	데이터 공급자	데이터 공급자	예

## 참고

- 데이터 공급자의 이름은 대괄호로 묶어야 합니다.
- `DataProvider` 함수를 사용하여 데이터 공급자에 대한 참조를 제공할 수 있습니다.

## 예제

"쿼리 1" 데이터 공급자에 10개의 행이 있는 경우, `NumberOfRows ([Query 1])` 은 10을 반환합니다.

## 관련 정보

[DataProvider](#) [페이지 118]

## 6.1.5.13 RefValueDate

### 설명

데이터 추적에 사용되는 참조 데이터의 날짜를 반환합니다.

### 함수 그룹

#### 데이터 공급자



## 구문

```
date RefValueDate()
```

## 예제

RefValueDate() 는 참조 날짜가 2008년 12월 15일인 경우 2008년 12월 15일을 반환합니다.

## 6.1.5.14 RefValueUserReponse

### 설명

참조 데이터가 현재 데이터인 시점에서 프롬프트에 응답된 값을 반환합니다.

### 함수 그룹

데이터 공급자

## 구문

```
string RefValueUserResponse([dp;]prompt_string[;Index])
```

## 입력

매개 변수	설명	유형	필수
dp	데이터 공급자	데이터 공급자	아니요
prompt_string	프롬프트 텍스트	문자열	예
인덱스	프롬프트 값의 데이터베이스 기본 값을 반환하도록 함수에 지시합니다.	키워드	아니요

## 참고

- 이 함수는 데이터 추적이 활성화되지 않은 경우 빈 문자열을 반환합니다.
- 데이터 공급자의 이름은 대괄호로 묶어야 합니다.
- `DataProvider` 함수를 사용하여 데이터 공급자에 대한 참조를 제공할 수 있습니다.
- 프롬프트 응답으로 둘 이상의 값을 선택한 경우 함수는 세미콜론으로 구분된 값 목록(또는 `Index` 연산자가 지정된 경우 기본 키)으로 구성된 문자열을 반환합니다.

## 예제

`RefValueUserResponse("Which City?"` 는 참조 데이터가 현재 데이터인 시점에서 "Which City?" 프롬프트에 "Los Angeles"를 입력한 경우 "Los Angeles"를 반환합니다.

`RefValueUserResponse([Sales Query];"Which city?")` 는 참조 데이터가 현재 데이터인 시점에서 "Sales Query" 데이터 공급자의 "Which City?" 프롬프트에 "Los Angeles"를 입력한 경우 "Los Angeles"를 반환합니다.

### 6.1.5.15 ServerValue

## 설명

계수의 데이터베이스 값을 반환합니다.

## 함수 그룹

데이터 공급자

## 구문

```
num ServerValue([measure])
```

## 입력

매개 변수	설명	유형	필수
measure	임의의 계수	계수	예

## 참고

- `ServerValue`는 계수 계산에 사용되는 차원 또는 계층구조에 적용된 모든 로컬 필터를 무시합니다.

## 예

`ServerValue([인터넷 매출])`은 [인터넷 매출] 계수의 데이터베이스 값을 반환합니다.

## 6.1.5.16 UniverseName

### 설명

데이터 공급자가 기반으로 하는 유니버스의 이름을 반환합니다.

### 함수 그룹

데이터 공급자

### 구문

```
string UniverseName(dp)
```

## 입력

매개 변수	설명	유형	필수
dp	데이터 공급자	데이터 공급자	예

## 참고

- 데이터 공급자의 이름이 변경되면 수식에서 dp의 값이 자동으로 업데이트됩니다. 데이터 공급자의 이름이 "Q1"로 바뀌면 수식은 `UniverseName([Q1])` 이 됩니다.
- 데이터 공급자의 이름은 대괄호로 묶어야 합니다.
- `DataProvider` 함수를 사용하여 데이터 공급자에 대한 참조를 제공할 수 있습니다.

## 예제

`UniverseName([쿼리 1])` 은 [쿼리 1] 데이터 공급자가 eFashion 유니버스를 기반으로 하는 경우 "eFashion"을 반환합니다.

## 관련 정보

[DataProvider \[페이지 118\]](#)

## 6.1.5.17 UserResponse

### 설명

프롬프트에 대한 응답을 반환합니다.

### 함수 그룹

#### 데이터 공급자

## 구문

```
string UserResponse([dp;]prompt_string[;Index])
```

## 입력

매개 변수	설명	유형	필수
dp	데이터 공급자	데이터 공급자	아니요
prompt_string	프롬프트 텍스트	문자열	예
인덱스	프롬프트 값의 데이터베이스 기본 값을 반환하도록 함수에 지시합니다.	키워드	아니요

## 참고

- 데이터 공급자의 이름은 대괄호로 묶어야 합니다.
- `DataProvider` 함수를 사용하여 데이터 공급자에 대한 참조를 제공할 수 있습니다.
- 프롬프트 대답에 둘 이상의 값을 선택한 경우 함수는 세미콜론으로 구분된 값 목록(또는 `Index` 연산자가 지정된 경우 기본 키)으로 구성된 문자열을 반환합니다.

## 예제

`UserResponse("어느 도시입니까?")` 는 "어느 도시입니까" 프롬프트에 "로스앤젤레스"를 입력한 경우 "로스앤젤레스"를 반환합니다.

`UserResponse([판매 쿼리];"어느 도시입니까?")` 는 "판매 쿼리" 데이터 공급자의 "어느 도시입니까?" 프롬프트에 "로스앤젤레스"를 입력한 경우 "로스앤젤레스"를 반환합니다.

`UserResponse([판매 쿼리];"어느 도시입니까?";Index)` 는 "판매 쿼리" 데이터 공급자의 "어느 도시입니까?" 프롬프트에 "로스앤젤레스"를 입력하고 로스앤젤레스의 데이터베이스 기본 키가 23인 경우 23을 반환합니다.

---

## 6.1.6 문서 함수

### 6.1.6.1 DocumentAuthor

#### 설명

문서 작성자의 InfoView 로그온을 반환합니다.

#### 함수 그룹

문서

#### 구문

```
string DocumentAuthor()
```

#### 예제

DocumentAuthor()는 문서 작성자의 로그인인 "gkn"인 경우 "gkn"을 반환합니다.

### 6.1.6.2 DocumentCreationDate

#### 설명

문서가 만들어진 날짜를 반환합니다.

#### 함수 그룹

문서

## 구문

```
date DocumentCreationDate()
```

## 예제

`DocumentCreationDate()`는 문서를 만든 날짜가 2008년 12월 15일인 경우 2008년 12월 15일을 반환합니다.

### 6.1.6.3 DocumentCreationTime

#### 설명

문서가 만들어진 시간을 반환합니다.

#### 함수 그룹

문서

## 구문

```
time DocumentCreationTime()
```

## 예제

`DocumentCreationTime()`은 문서를 만든 시간이 11시 15분인 경우 11:15를 반환합니다.

### 6.1.6.4 DocumentDate

#### 설명

문서를 마지막으로 저장한 날짜를 반환합니다.

---

## 함수 그룹

문서

### 구문

```
date DocumentDate()
```

### 예제

`DocumentDate()`는 문서가 2005년 8월 8일에 마지막으로 저장된 경우 2005년 8월 8일을 반환합니다.

## 6.1.6.5 DocumentName

### 설명

문서 이름을 반환합니다.

## 함수 그룹

문서

### 구문

```
string DocumentName()
```

### 예제

`DocumentName()`은 문서의 이름이 "판매 보고서"인 경우 "판매 보고서"를 반환합니다.



## 6.1.6.6 DocumentOwner

### 설명

문서 소유자(문서를 마지막으로 저장한 사람)의 BI 실행 패드 로그인/사용자 이름을 반환합니다. 원래 문서 만든 이/작성자를 반환하려면 DocumentAuthor 함수를 사용하십시오.

### 함수 그룹

문서

### 구문

```
string DocumentOwner()
```

### 예

DocumentOwner()는 문서를 마지막으로 저장한 사람의 사용자 이름 또는 로그인 이름이 "gkn"인 경우 "gkn"을 반환합니다.

## 6.1.6.7 DocumentPartiallyRefreshed

### 설명

문서를 부분적으로 새로 고쳤는지 여부를 확인합니다.

### 함수 그룹

문서

### 구문

```
bool DocumentPartiallyRefreshed()
```

---

## 참고

`DocumentPartiallyRefreshed`는 `If` 함수에서 사용할 수 있는 부울 값을 반환합니다.

## 예제

문서를 부분적으로 새로 고친 경우 `DocumentPartiallyRefreshed()` 는 `True`를 반환합니다.

## 6.1.6.8 DocumentTime

### 설명

문서를 마지막으로 저장한 시간을 반환합니다.

### 함수 그룹

문서

### 구문

```
time DocumentTime()
```

## 참고

반환되는 시간의 형식은 셀 서식에 따라 달라집니다.

## 예제

`DocumentTime()` 은 문서를 마지막으로 저장한 시간이 15시 45분인 경우 15:45를 반환합니다.

## 6.1.6.9 DrillFilters

### 설명

드릴 모드의 선언된 보고서에 문서나 개체에 적용된 드릴 필터의 결과를 반환합니다. 문서 내에서 다른 보고서를 선언할 수 있습니다. 이 보고서를 선언하지 않는 경우 현재 활성 보고서가 사용됩니다.

### 함수 그룹

### 문서

### 구문

**문자열** `DrillFilters([obj|separator[;report]])`

### 입력

매개 변수	설명	유형	필수
obj	보고서 개체	보고서 개체	obj 또는 separator 중 하나 필수
separator	드릴 필터 구분 기호	문자열	obj 또는 separator 중 하나 필수
보고서	선택 사항입니다. 사용하려는 보고서의 이름입니다. 문서에 있어야 합니다. 보고서가 선언되지 않은 경우 현재 보고서가 사용됩니다.	문자열	obj 또는 separator 중 하나 필수

### 참고

- 수식을 수동으로 입력할 필요없이 DrillFilters 셀을 삽입하여 DrillFilters 함수를 직접 삽입할 수 있습니다.
- 개체를 지정하지 않으면 문서에 적용된 모든 드릴 필터가 반환됩니다.

## 예제

`DrillFilters()`는 [국가] 개체를 미국으로 제한하는 드릴 필터가 문서에 있는 경우 "미국"을 반환합니다.

[국가]를 "미국"으로 제한하고 [연도]를 1999로 제한하는 필터가 문서에 있는 경우, `DrillFilters()`는 "미국 - 1999"를 반환합니다.

[국가]를 "미국"으로 제한하고 [연도]를 1999로 제한하는 필터가 문서에 있는 경우 `DrillFilters("/")`는 "미국 / 1999"를 반환합니다.

[분기]를 "3분기"로 제한하는 드릴 필터가 문서에 있는 경우 `DrillFilters([분기])`는 "3분기"를 반환합니다.

## 6.1.6.10 PromptSummary

### 설명

문서의 모든 프롬프트에 대한 프롬프트 텍스트 및 사용자 응답을 반환합니다.

### 함수 그룹

### 문서

### 구문

```
string PromptSummary()
```

## 예제

`PromptSummary` 함수 출력의 예는 다음과 같이 나타납니다.

```
Enter Quantity Sold: 5000
Enter value(s) for State (optional): California, Texas, Utah
Enter Customer (optional):
```

## 6.1.6.11 QuerySummary

### 설명

문서의 쿼리에 대한 정보를 반환합니다.

### 함수 그룹

문서

### 구문

```
string QuerySummary([dp])
```

### 입력

매개 변수	설명	유형	필수
dp	데이터 공급자	데이터 공급자	아니요

### 참고

- 데이터 공급자의 이름은 대괄호로 묶어야 합니다.

### 예제

QuerySummary()는 문서의 모든 쿼리에 대한 정보를 반환합니다.

QuerySummary([쿼리 1])는 [쿼리 1] 데이터 공급자를 기반으로 쿼리에 대한 정보를 반환합니다.

출력 예제:

```
Query 1:
Universe: eFashion
Last execution time: 1s
NB of rows: 34500
Result objects: State, Year, Sales Revenue
Scope of analysis: State, City, Year, Quarter, Month
```

```

Filters:
  (State inlist{"US";"France";}
  And (Sales Revenue Greater Than 1000000
  Or Sales Revenue Less Than 10000))
Query 2:
  Source file: D:\Data\dataacar.xls
  Result objects: State, Year, Sales Revenue

```

## 6.1.6.12 ReportFilter

### 설명

개체 또는 보고서에 적용된 보고서 필터를 반환합니다.

### 함수 그룹

### 문서

### 구문

```
string ReportFilter(obj)
```

### 입력

매개 변수	설명	유형	필수
obj	보고서 개체	보고서 개체	예

### 예제

`ReportFilter([국가])`는 국가 개체에 "미국"으로 제한하는 보고서 필터가 있는 경우 "미국"을 반환합니다.

## 6.1.6.13 ReportFilterSummary

### 설명

문서 또는 보고서의 보고서 필터 요약을 반환합니다.

### 함수 그룹

문서

### 구문

```
string ReportFilterSummary(report_name)
```

### 입력

매개 변수	설명	유형	필수
report_name	보고서의 이름	문자열	아니요

### 참고

report\_name이 생략된 경우 ReportFilterSummary는 문서의 모든 보고서의 모든 보고서 필터 요약을 반환합니다.

### 예제

ReportFilterSummary()는 문서의 모든 보고서 필터에 대한 정보를 반환합니다.

ReportFilterSummary("Report1")는 "보고서1" 보고서의 보고서 필터에 대한 정보를 반환합니다.

ReportFilterSummary 함수 출력의 예는 다음과 같이 나타납니다.

```
Filters on Report1:
    (Sales Revenue Greater Than 1000000
    Or (Sales Revenue Less Than 3000))
Filters on Section on City:
    (City InList{"Los Angeles";"San Diego";})
Ranking Filter:
```

```
Revenue] (Count)) (Top 10 & Bottom 10 [Customer] Based on [Sales
```

## 6.1.7 논리 함수

### 6.1.7.1 Even

#### 설명

숫자가 짝수인지 여부를 확인합니다.

#### 함수 그룹

#### 논리

#### 구문

```
bool Even(number)
```

#### 입력

매개 변수	설명	유형	필수
number	임의의 숫자	숫자	예

#### 참고

- Even은 If 함수에서 사용할 수 있는 부울 값을 반환합니다.
- Even을 열에 직접 배치할 경우 정수를 반환합니다(1=true, 0=false). 부울 숫자 형식을 사용하여 이 정수에 서식을 지정할 수 있습니다.



예

Even (4) 는 True를 반환합니다.

Even (3) 은 False를 반환합니다.

Even (23.2) 는 False를 반환합니다.

Even (-4) 는 True를 반환합니다.

Even (-2.2) 는 False를 반환합니다.

## 6.1.7.2 IsDate

설명

값이 날짜인지 여부를 결정합니다.

함수 그룹

논리

구문

```
bool IsDate (obj)
```

입력

매개 변수	설명	유형	필수
obj	모든 보고서 개체	보고서 개체	예

참고

- IsDate는 If 함수에서 사용할 수 있는 부울 값을 반환합니다.
- IsDate를 열에 직접 배치할 경우 정수를 반환합니다(1=true, 0=false). 부울 숫자 형식을 사용하여 이 정수에 서식을 지정할 수 있습니다.

예

IsDate([예약 날짜])는 [예약 날짜]가 날짜인 경우 True를 반환합니다.

다음은 [예약 날짜]가 날짜인 경우 "날짜"를 반환합니다.

- If(IsDate([예약 날짜])) Then "날짜" Else "날짜 아님"
- If IsDate([예약 날짜]) Then "날짜" Else "날짜 아님"

관련 정보

[If...Then...Else \[페이지 206\]](#)

## 6.1.7.3 IsError

설명

개체가 오류를 반환하는지 여부를 결정합니다.

함수 그룹

논리

구문

```
bool IsError(obj)
```

입력

매개 변수	설명	유형	필수
obj	모든 보고서 개체	보고서 개체	예

## 참고

- `IsError`는 `If` 함수에서 사용할 수 있는 부울 값을 반환합니다.
- `IsError`를 열에 직접 배치할 경우 정수가 반환됩니다(1=true, 0=false). 부울 숫자 형식을 사용하여 이 정수에 서식을 지정할 수 있습니다.

## 예

`IsError([수익])`은 [수익] 변수가 오류를 반환하지 않는 경우 `False`를 반환합니다.

`IsError([Average Guests])`는 [Average Guests] 변수가 0으로 나눔(#DIV/0) 오류를 반환하는 경우 `True`를 반환합니다.

`If IsError([Average Guests]) Then "Error" Else "No error"`는 [Average Guests] 변수가 0으로 나눔(#DIV/0) 오류를 반환하는 경우 "Error"를 반환합니다.

## 관련 정보

[If...Then...Else \[페이지 206\]](#)

## 6.1.7.4 IsLogical

### 설명

값이 부울인지 여부를 결정합니다.

### 함수 그룹

### 논리

### 구문

```
bool IsLogical(obj)
```

## 입력

매개 변수	설명	유형	필수
obj	모든 보고서 개체	보고서 개체	예

## 참고

- IsLogical은 If 함수에서 사용할 수 있는 부울 값을 반환합니다.
- IsLogical을 열에 직접 배치할 경우 정수를 반환합니다(1=true, 0=false). 부울 숫자 형식을 사용하여 이 정수에 서식을 지정할 수 있습니다.

## 예

IsLogical(IsString([국가]))는 True를 반환합니다.

[국가]가 부울 이외의 데이터 형식을 반환하는 경우, IsLogical([국가])는 False를 반환합니다.

If IsLogical(IsDate([국가])) Then "Boolean" Else "Not boolean"은 "Boolean"을 반환합니다.

## 관련 정보

[If...Then...Else \[페이지 206\]](#)

## 6.1.7.5 IsNull

### 설명

값이 null인지 여부를 결정합니다.

### 함수 그룹

### 논리

## 구문

```
bool IsNull (obj)
```

## 입력

매개 변수	설명	유형	필수
obj	모든 보고서 개체	보고서 개체	예

## 참고

- IsNull은 If 함수에서 사용할 수 있는 부울 값을 반환합니다.
- IsNull을 열에 직접 배치할 경우 정수를 반환합니다(1=true, 0=false). 부울 숫자 형식을 사용하여 이 정수에 서식을 지정할 수 있습니다.

## 예

IsNull ([수익]) 는 [수익] 변수가 null이 아닌 경우 False를 반환합니다.

IsNull ([Average Guests]) 는 [Average Guests] 변수가 null인 경우 True를 반환합니다.

## 관련 정보

[If...Then...Else \[페이지 206\]](#)

## 6.1.7.6 IsNumber

### 설명

값이 숫자인지 여부를 결정합니다.

## 함수 그룹

### 논리

## 구문

```
bool IsNumber (obj)
```

## 입력

매개 변수	설명	유형	필수
obj	모든 보고서 개체	보고서 개체	예

## 참고

- IsNumber는 If 함수에서 사용할 수 있는 부울 값을 반환합니다.
- IsNumber를 열에 직접 배치할 경우 정수를 반환합니다(1=true, 0=false). 부울 숫자 형식을 사용하여 이 정수에서 식을 지정할 수 있습니다.

## 예

IsNumber([수익])은 [수익] 변수가 숫자인 경우 True를 반환합니다.

IsNumber([Customer Name])은 [Customer Name] 변수가 숫자가 아닌 경우 False를 반환합니다.

If IsNumber([Customer Name]) Then "Number" Else "Not a number"는 [Customer Name] 변수가 숫자가 아닌 경우 "Not a number"을 반환합니다.

## 관련 정보

[If...Then...Else \[페이지 206\]](#)

## 6.1.7.7 IsString

### 설명

값이 문자열인지 여부를 결정합니다.

### 함수 그룹

### 논리

### 구문

```
bool IsString(obj)
```

### 입력

매개 변수	설명	유형	필수
obj	모든 보고서 개체	보고서 개체	예

### 참고

- IsString은 If 함수에서 사용할 수 있는 부울 값을 반환합니다.
- IsString을 열에 직접 배치할 경우 정수를 반환합니다(1=true, 0=false). 부울 숫자 형식을 사용하여 이 정수에 서식을 지정할 수 있습니다.

### 예

IsString([수익])은 [수익] 변수가 문자열이 아닌 경우 false를 반환합니다.

IsString([Customer Name])은 [Customer Name] 변수가 문자열인 경우 true를 반환합니다.

If IsString([Customer Name]) Then "String" Else "Not a string"은 [Customer Name] 변수가 문자열인 경우 "String"을 반환합니다.

## 관련 정보

[If...Then...Else \[페이지 206\]](#)

### 6.1.7.8 IsTime

#### 설명

변수가 시간 변수인지 여부를 확인합니다.

#### 함수 그룹

#### 논리

#### 구문

```
bool IsTime(obj)
```

#### 입력

매개 변수	설명	유형	필수
obj	모든 보고서 개체	보고서 개체	예

#### 참고

- IsTime은 If 함수에서 사용할 수 있는 부울 값을 반환합니다.
- IsTime을 열에 직접 배치할 경우 정수를 반환합니다(1=true, 0=false). 부울 숫자 형식을 사용하여 이 정수에 서식을 지정할 수 있습니다.

#### 예

IsTime([Reservation Time])은 [Reservation Time] 변수가 시간 변수인 경우 true를 반환합니다.



IsTime([Average Guests])는 [Average Guests] 변수가 시간 변수가 아닌 경우 false를 반환합니다.

If IsTime([Average Guests]) Then "Time" Else "Not time"은 [Average Guests] 변수가 시간 변수가 아닌 경우 "Not time"을 반환합니다.

## 관련 정보

[If...Then...Else \[페이지 206\]](#)

## 6.1.7.9 Odd

### 설명

숫자가 홀수인지 여부를 확인합니다.

### 함수 그룹

논리

### 구문

```
bool Odd(number)
```

### 입력

매개 변수	설명	유형	필수
number	임의의 숫자	숫자	예

### 참고

- Odd는 If 함수에서 사용할 수 있는 부울 값을 반환합니다.
- Odd를 옆에 직접 배치할 경우 정수를 반환합니다(1=true, 0=false). 부울 숫자 형식을 사용하여 이 정수에 서식을 지정할 수 있습니다.

- Odd는 10진수의 소수 자릿수를 무시합니다.

## 예제

Odd (5) 는 True를 반환합니다.

Odd (4) 는 False를 반환합니다.

Odd (23.2) 는 True를 반환합니다.

Odd (24.2) 는 True를 반환합니다.

Odd (-23.2) 는 True를 반환합니다.

Odd (-24.2) 는 True를 반환합니다.

## 관련 정보

[If...Then...Else \[페이지 206\]](#)

## 6.1.8 숫자 함수

### 6.1.8.1 Abs

#### 설명

숫자의 절대 값을 반환합니다.

#### 함수 그룹

숫자

#### 구문

```
num Abs (number)
```

## 입력

매개 변수	설명	유형	필수
number	임의의 숫자	숫자	예

## 예제

Abs (25) 는 25를 반환합니다.

Abs (-11) 는 11을 반환합니다.

## 6.1.8.2 Ceil

### 설명

가장 가까운 정수로 올림한 숫자를 반환합니다.

### 함수 그룹

숫자

### 구문

```
num Ceil (number)
```

## 입력

매개 변수	설명	유형	필수
number	임의의 숫자	숫자	예

## 예제

$\text{Ceil}(2.4)$  은 3을 반환합니다.

$\text{Ceil}(3.1)$  은 4를 반환합니다.

$\text{Ceil}(-3.1)$  은 -3을 반환합니다.

## 6.1.8.3 Cos

### 설명

각도의 코사인 값을 반환합니다.

### 함수 그룹

숫자

### 구문

```
num Cos (angle)
```

### 입력

매개 변수	설명	유형	필수
angle	라디안으로 표시된 각도	숫자	예

## 예제

$\text{Cos}(180)$  은 -0.6을 반환합니다.

## 6.1.8.4 EuroConvertFrom

### 설명

유로 금액을 다른 통화로 변환합니다.

### 함수 그룹

숫자

### 구문

```
num EuroConvertFrom(euro_amount; curr_code; round_level)
```

### 입력

매개 변수	설명	유형	필수
euro_amount	유로화 금액	숫자	예
curr_code	대상 통화의 ISO 코드	문자열	예
round_level	숫자를 반올림할 소수 자릿수	숫자	예

### 참고

통화 코드는 해당 통화의 사용을 2002년 1월에 중단하기 전에 통화가 유로화에 상대적으로 고정된 12개의 EU 통화 코드 중 하나여야 합니다. 그렇지 않은 경우, #ERROR가 반환됩니다. 해당 통화는 다음과 같습니다.

BEF	벨기에 프랑
DEM	독일 마르크
GRD	그리스 드라크마
ESP	스페인 페세타
FRF	프랑스 프랑
IEP	아일랜드 펀트
ITL	이탈리아 리라

LUF	룩셈부르크 프랑
NLG	네덜란드 길더
ATS	오스트리아 실링
PTS	포르투갈 에스쿠도
FIM	핀란드 마르크

## 예제

`EuroConvertFrom(1000;"FRF";2)` 은 6559.57을 반환합니다.

`EuroConvertFrom(1000;"FRF";1)` 은 6559.60을 반환합니다.

`EuroConvertFrom(1000.04;"DEM";2)` 은 1955.83을 반환합니다.

`EuroConvertFrom(1000.04;"DEM";1)` 은 1955.80을 반환합니다.

## 관련 정보

[값 반올림 및 자르기 \[페이지 242\]](#)

## 6.1.8.5 EuroConvertTo

### 설명

금액을 유로화로 변환합니다.

### 함수 그룹

숫자

### 구문

```
num EuroConvertTo(noneuro_amount;curr_code;round_level)
```

## 입력

매개 변수	설명	유형	필수
euro_amount	비유로화 금액	숫자	예
curr_code	비유로 통화의 ISO 코드	문자열	예
round_level	숫자를 반올림할 소수 자릿수	숫자	예

## 예제

`EuroConvertTo(6559;"FRF";2)`는 999.91을 반환합니다.

`EuroConvertTo(6559;"FRF";1)`는 999.90을 반환합니다.

`EuroConvertTo(1955;"DEM";2)`는 999.58을 반환합니다.

`EuroConvertTo(1955;"DEM";1)`는 999.60을 반환합니다.

## 참고

통화 코드는 해당 통화의 사용을 2002년 1월에 중단하기 전에 통화가 유로화에 상대적으로 고정된 12개의 EU 통화 코드 중 하나여야 합니다. 그렇지 않은 경우, #ERROR가 반환됩니다. 해당 통화는 다음과 같습니다.

BEF	벨기에 프랑
DEM	독일 마르크
GRD	그리스 드라크마
ESP	스페인 페세타
FRF	프랑스 프랑
IEP	아일랜드 펀트
ITL	이탈리아 리라
LUF	룩셈부르크 프랑
NLG	네덜란드 길더
ATS	오스트리아 실링
PTS	포르투갈 에스쿠도
FIM	핀란드 마르크

## 관련 정보

[값 반올림 및 자르기 \[페이지 242\]](#)

### 6.1.8.6 EuroFromRoundError

#### 설명

유로화->비유로화 변환 시의 반올림 오차를 반환합니다.

#### 함수 그룹

숫자

#### 구문

```
num EuroFromRoundError(euro_amount; curr_code; round_level)
```

#### 입력

매개 변수	설명	유형	필수
euro_amount	유로화 금액	숫자	예
curr_code	대상 통화의 ISO 코드	문자열	예
round_level	숫자를 반올림할 소수 자릿수	숫자	예

#### 출력

계산에서의 반올림 오차



## 예

`EuroFromRoundError(1000;"FRF";2)`는 0을 반환합니다(반올림하지 않은 변환과 소수 둘째 자리로 반올림한 변환 사이의 차이가 없음).

`EuroFromRoundError(1000;"FRF";1)`은 0.03을 반환합니다. (반올림하지 않은 변환은 6559.57, 소수 첫째 자리로 반올림한 변환은 6559.60. 따라서 반올림 오차는 0.03)

`EuroFromRoundError(1000;"DEM";2)`는 0을 반환합니다(반올림하지 않은 변환과 소수 둘째 자리로 반올림한 변환 사이의 차이가 없음).

`EuroFromRoundError(1000;"DEM";1)`은 -0.01을 반환합니다. (반올림하지 않은 변환은 1955.83, 소수 첫째 자리로 반올림한 변환은 1995.80. 따라서 반올림 오차는 -0.03)

## 참고

통화 코드는 해당 통화의 사용을 2002년 1월에 중단하기 전에 통화가 유로화에 상대적으로 고정된 12개의 EU 통화 코드 중 하나여야 합니다. 그렇지 않은 경우, #ERROR가 반환됩니다. 해당 통화는 다음과 같습니다.

BEF	벨기에 프랑
DEM	독일 마르크
GRD	그리스 드라크마
ESP	스페인 페세타
FRF	프랑스 프랑
IEP	아일랜드 펀트
ITL	이탈리아 리라
LUF	룩셈부르크 프랑
NLG	네덜란드 길더
ATS	오스트리아 실링
PTS	포르투갈 에스쿠도
FIM	핀란드 마르크

## 관련 정보

[값 반올림 및 자르기 \[페이지 242\]](#)

## 6.1.8.7 EuroToRoundError

### 설명

비유로화->유로화 변환 시의 반올림 오차를 반환합니다.

### 함수 그룹

### 숫자

### 구문

```
num EuroToRoundError(noneuro_amount;curr_code;round_level)
```

### 입력

매개 변수	설명	유형	필수
euro_amount	비유로화 금액	숫자	예
curr_code	비유로 통화의 ISO 코드	문자열	예
round_level	숫자를 반올림할 소수 자릿수	숫자	예

### 예

`EuroToRoundError(6559;"FRF";2)` 는 0을 반환합니다(반올림하지 않은 변환과 소수 둘째 자리로 반올림한 변환 사이의 차이가 없음).

`EuroToRoundError(6559;"FRF";1)` 은 -0.01을 반환합니다. (반올림하지 않은 변환은 999.91, 소수 첫째 자리로 반올림한 변환은 999.90. 따라서 반올림 오차는 -0.01)

`EuroToRoundError(1955;"DEM";2)` 는 0을 반환합니다(반올림하지 않은 변환과 소수 둘째 자리로 반올림한 변환 사이의 차이가 없음).

`EuroToRoundError(1955;"DEM";1)` 은 0.02를 반환합니다. (반올림하지 않은 변환은 999.58, 소수 첫째 자리로 반올림한 변환은 999.60. 따라서 반올림 오차는 0.02)

참고

통화 코드는 해당 통화의 사용을 2002년 1월에 중단하기 전에 통화가 유로화에 상대적으로 고정된 12개의 EU 통화 코드 중 하나여야 합니다. 그렇지 않은 경우, #ERROR가 반환됩니다. 해당 통화는 다음과 같습니다.

BEF	벨기에 프랑
DEM	독일 마르크
GRD	그리스 드라크마
ESP	스페인 페세타
FRF	프랑스 프랑
IEP	아일랜드 펜트
ITL	이탈리아 리라
LUF	룩셈부르크 프랑
NLG	네덜란드 길더
ATS	오스트리아 실링
PTS	포르투갈 에스쿠도
FIM	핀란드 마르크

관련 정보

[값 반올림 및 자르기 \[페이지 242\]](#)

6.1.8.8 Exp

설명

지수를 반환합니다(e의 거듭제곱)

함수 그룹

숫자

## 구문

```
num Exp(power)
```

## 입력

매개 변수	설명	유형	필수
거듭제곱	거듭제곱	숫자	예

## 참고

지수는 상수  $e(2.718...)$ 의 거듭제곱입니다.

## 예제

`Exp(2,2)`는 9.03을 반환합니다.

## 6.1.8.9 Fact

### 설명

숫자의 계승 값을 반환합니다.

### 함수 그룹

숫자

## 구문

```
int Fact(number)
```

## 입력

매개 변수	설명	유형	필수
number	임의의 숫자	숫자	예

## 참고

number의 계승 값은 1에서 number까지의 모든 정수의 곱입니다.

## 예제

Fact (4) 는 24를 반환합니다.

Fact (5.9) 는 120을 반환합니다.

## 6.1.8.10 Floor

### 설명

가장 가까운 정수로 내림한 숫자를 반환합니다.

### 함수 그룹

숫자

### 구문

```
int Floor(number)
```

## 입력

매개 변수	설명	유형	필수
number	임의의 숫자	숫자	예

## 예제

`Floor(24.4)`는 24를 반환합니다.

## 6.1.8.11 Ln

### 설명

숫자의 자연 로그를 반환합니다.

### 함수 그룹

숫자

### 구문

```
num Ln(number)
```

## 입력

매개 변수	설명	유형	필수
숫자	임의의 숫자	숫자	예

## 예

`Ln(10)`은 2.3을 반환합니다.

## 6.1.8.12 Log

### 설명

지정된 숫자에 대해 지정된 밑수를 사용한 로그값을 반환합니다.

### 함수 그룹

### 숫자

### 구문

```
num Log (number; base)
```

### 입력

매개 변수	설명	유형	필수
number	임의의 숫자	숫자	예
base	로그값의 밑수	숫자	예

### 예제

Log (125; 5) 는 3을 반환합니다.

## 6.1.8.13 Log10

### 설명

밑을 10으로 한 숫자의 로그값을 반환합니다.

## 함수 그룹

### 숫자

## 구문

```
num Log10 (number)
```

## 입력

매개 변수	설명	유형	필수
number	임의의 숫자	숫자	예

## 예제

Log10 (100) 은 2를 반환합니다.

## 6.1.8.14 Mod

## 설명

두 숫자를 나눈 후 남은 나머지를 반환합니다.

## 함수 그룹

### 숫자

## 구문

```
num Mod (dividend;divisor)
```



## 입력

매개 변수	설명	유형	필수
dividend	피제수입니다.	숫자	예
divisor	제수입니다.	숫자	예

## 예제

$\text{Mod}(10; 4)$  는 2를 반환합니다.

$\text{Mod}(10.2; 4.2)$  는 1.8을 반환합니다.

## 6.1.8.15 Power

### 설명

거듭제곱된 수를 반환합니다.

### 함수 그룹

숫자

### 구문

```
num Power(number;power)
```

## 입력

매개 변수	설명	유형	필수
number	거듭제곱할 숫자입니다.	숫자	예
거듭제곱	거듭제곱	숫자	예

## 예제

`Power(10;2)`는 100을 반환합니다.

## 6.1.8.16 Rank

### 설명

다양한 차원을 기준으로 계수의 순위를 설정합니다.

### 함수 그룹

숫자

### 구문

```
int Rank(measure;[ranking_dims][;Top|Bottom][;(reset_dims)])
```

### 입력

매개 변수	설명	유형	필수
measure	순위를 설정할 계수입니다.	계수	예
ranking_dims	계수의 순위를 설정하는 데 사용되는 차원입니다.	차원 목록	아니요
Top Bottom	순위 정렬 순서를 설정합니다. <ul style="list-style-type: none"><li>Top - 내림차순</li><li>Bottom - 오름차순</li></ul>	키워드	아니요(Top이 기본값)
reset_dims	순위를 재설정하는 차원입니다.	차원 목록	아니요

### 참고

- 순위 차원을 지정하지 않은 경우 이 함수는 기본 계산 컨텍스트를 사용하여 순위를 계산합니다.

- 순위 또는 재설정 차원 목록에 차원이 하나만 있는 경우에도 항상 차원을 괄호로 묶어야 합니다.
- 순위 또는 재설정 차원 집합을 지정하는 경우 세미콜론으로 구분해야 합니다.
- 기본적으로 순위는 섹션 또는 블록 나누기 후에 다시 설정됩니다.

## 예제

다음 표에서 순위는 `Rank([수익];([국가]))`에 의해 지정됩니다.

국가	수익	순위
프랑스	835,420	2
미국	2,451,104	1

다음 표에서 순위는 `Rank([수익];([국가]);Bottom)`에 의해 지정됩니다. `Bottom` 인수는 측정값 순위가 내림차순으로 정렬되는 것을 의미합니다.

국가	수익	순위
프랑스	835,420	1
미국	2,451,104	2

다음 표에서 순위는 `Rank([수익];([국가];[휴양지]))`에 의해 지정됩니다.

국가	휴양지	수익	순위
프랑스	French Riviera	835,420	3
미국	Bahamas Beach	971,444	2
미국	Hawaiian Club	1,479,660	1

다음 표에서 순위는 `Rank([수익];([국가];[연도]);([국가]))`에 의해 지정됩니다. 순위는 국가 차원에서 다시 설정됩니다.

국가	연도	수익	순위
프랑스	FY1998	295,940	1
프랑스	FY1999	280,310	2
프랑스	FY2000	259,170	3
미국	FY1998	767,614	3
미국	FY1999	826,930	2
미국	FY2000	856,560	1

## 관련 정보

[Bottom/Top 연산자 \[페이지 226\]](#)

## 6.1.8.17 Round

### 설명

숫자를 반올림합니다.

### 함수 그룹

숫자

### 구문

```
num Round (number;round_level)
```

### 입력

매개 변수	설명	유형	필수
숫자	반올림할 숫자	숫자	예
round_level	숫자를 반올림할 소수 자릿수	숫자	예

### 예제

Round (9.44;1) 는 9.4를 반환합니다.

Round (9.45;1) 는 9.5를 반환합니다.

Round (9.45;0) 는 9를 반환합니다.

Round (9.45;-1) 는 10을 반환합니다.

Round (4.45;-1) 는 0을 반환합니다.

### 관련 정보

[값 반올림 및 자르기 \[페이지 242\]](#)

## 6.1.8.18 Sign

### 설명

숫자의 부호를 반환합니다.

### 함수 그룹

숫자

### 구문

```
int Sign(number)
```

### 입력

매개 변수	설명	유형	필수
number	임의의 숫자	숫자	예

### 참고

Sign은 number가 음수인 경우에는 -1, number가 0인 경우에는 0, number가 양수인 경우에는 1을 반환합니다.

### 예제

Sign(3)은 1을 반환합니다.

Sign(-27.5)은 -1을 반환합니다.

## 6.1.8.19 Sin

### 설명

각도의 사인 값을 반환합니다.

### 함수 그룹

숫자

### 구문

```
num Sin (angle)
```

### 입력

매개 변수	설명	유형	필수
angle	라디안으로 표시된 각도	숫자	예

### 예

`Sin(234542)`는 소수점 설정에 따라 -0.116992 또는 -0.12를 반환할 수 있습니다.

## 6.1.8.20 Sqrt

### 설명

숫자의 제곱근을 반환합니다.

### 함수 그룹

숫자

## 구문

```
num Sqrt (number)
```

## 입력

매개 변수	설명	유형	필수
number	임의의 숫자	숫자	예

## 예제

`Sqrt (25)` 는 5를 반환합니다.

## 6.1.8.21 Tan

### 설명

각도의 탄젠트 값을 반환합니다.

### 함수 그룹

숫자

## 구문

```
num Tan (angle)
```

## 입력

매개 변수	설명	유형	필수
angle	라디안으로 표시된 각도	숫자	예

## 예제

$\text{Tan}(90)$ 은 -2를 반환합니다.

## 6.1.8.22 ToNumber

### 설명

문자열을 숫자로 반환합니다.

### 함수 그룹

숫자

### 구문

```
num ToNumber(string)
```

또는

## 입력

매개 변수	설명	유형	필수
문자열	숫자의 문자열	문자열	예



## 참고

string이 숫자 또는 날짜 시간이 아니면 ToNumber가 #ERROR를 반환합니다.

## 예

ToNumber("45")는 45를 반환합니다.

## 6.1.8.23 Truncate

### 설명

숫자를 자릅니다.

### 함수 그룹

숫자

### 구문

```
num Truncate(number;truncate_level)
```

### 입력

매개 변수	설명	유형	필수
숫자	반올림할 숫자	숫자	예
truncate_level	숫자를 자르는 소수 자릿수	숫자	예

## 참고

## 예제

`Truncate(3.423;2)` 는 3.42를 반환합니다.

## 관련 정보

[값 반올림 및 자르기 \[페이지 242\]](#)

## 6.1.9 집합 함수

### 6.1.9.1 Ancestor

#### 설명

멤버의 상위 멤버를 반환합니다.

#### 함수 그룹

#### 집합

#### 구문

```
member Ancestor (member; level | distance)
```

#### 입력

매개 변수	설명	유형	필수
member	임의의 멤버	멤버	예
level	상위 항목의 수준	수준	level과 distance 중 하나 필수

매개 변수	설명	유형	필수
distance	현재 수준에서 상위 수준까지의 거리	int	level과 distance 중 하나 필수

## 참고

- `Ancestor`는 독립 실행형 함수로 사용되지 않습니다. 집계 함수의 입력 매개 변수에서 집계에 대한 멤버 집합을 지정하는 경우 사용됩니다.
- `member`는 계층구조의 현재 멤버입니다. 계층구조가 블록 컨텍스트 안에 있지 않으면 수식이 빈 값을 반환합니다.
- `distance`는 양수여야 합니다.

## 예제

다음 예는 모두 영어 버전의 데이터 소스에서 가져왔습니다.

다음 지리 계층구조를 바탕으로 각 고객의 인터넷 매출액 영향력을 알아보려고 합니다.



먼저 도시별로 국가 내의 인터넷 매출액을 구합니다.

```
=Sum([Query 2].[Internet Sales].[Internet Sales Amount];{Ancestor([Customer Geography];[Customer Geography].[City])})
```

Customer Geography	'=Sum([Query 2].[Internet Sales].[	Internet Sales Amount
[-] All Customers		29,358,677.22
[-] Australia		9,061,000.58
[-] New South Wales		3,934,485.73
[-] Coffs Harbour	235,454.97	235,454.97
[-] 2450	235,454.97	235,454.97
Adriana Smith	235,454.97	5,333.25
Aimee Guo	235,454.97	77.27
Allison R. Young	235,454.97	39.98
Ann A. Sara	235,454.97	39.98
Antonio G. Pattersor	235,454.97	8,068.03
Ariana Stewart	235,454.97	6,070.59
Arthur Kapoor	235,454.97	23.97
Barbara W. Lal	235,454.97	2,795.01
Bobby D. Saunders	235,454.97	120.48
Brianna J. Johnson	235,454.97	38.98

그리고 전세계에서 해당 국가가 차지하는 인터넷 매출액 중 각 도시의 기여분을 계산합니다.

```
=([Query 2].[Internet Sales].[Internet Sales Amount] / Sum([Query 2].[Internet Sales].[Internet Sales Amount];{Ancestor([Customer Geography];[Customer Geography].[City]))))
```

Customer Geography	'=Sum([Query 2].[Internet Sales].[	Internet Sales Amount
[-] All Customers		29,358,677.22
[-] Australia		9,061,000.58
[-] New South Wales		3,934,485.73
[-] Coffs Harbour	100.00%	235,454.97
[-] 2450	100.00%	235,454.97
Adriana Smith	2.27%	5,333.25
Aimee Guo	0.03%	77.27
Allison R. Young	0.02%	39.98
Ann A. Sara	0.02%	39.98
Antonio G. Pattersor	3.43%	8,068.03
Ariana Stewart	2.58%	6,070.59
Arthur Kapoor	0.01%	23.97
Barbara W. Lal	1.19%	2,795.01
Bobby D. Saunders	0.05%	120.48
Brianna J. Johnson	0.02%	38.98
Bruce G. Madan	0.03%	65.96

#### i 노트

SAPBW 공급자에 대해 BICS 연결을 사용할 때는 수준의 이름을 지정하는 대신 오프셋을 지정해야 합니다.

```
=([Query 2].[Internet Sales].[Internet Sales Amount] / Sum([Query 2].[Internet Sales].[Internet Sales Amount];{Ancestor([Customer Geography];2)}))
```

여기에서는 시/도 및 국가에 대한 결과도 반환됩니다.

## 관련 정보

[Aggregate \[페이지 39\]](#)

[Average \[페이지 46\]](#)

[Count \[페이지 47\]](#)

[Max \[페이지 52\]](#)

[Min \[페이지 54\]](#)

[Sum \[페이지 74\]](#)

## 6.1.9.2 Children

### 설명

집계 함수 안에서 사용되어 계층구조 멤버의 하위 멤버를 반환합니다.

### 함수 그룹

### 집합

### 구문

```
member_set member.Children
```

### 입력

매개 변수	설명	유형	필수
member	임의의 멤버	멤버	예

### 참고

- Children은 독립 실행형 함수로 사용되지 않습니다. 집계 함수의 입력 매개 변수에서 집계에 대한 멤버 집합을 지정하는 경우 사용됩니다.
- member는 계층구조의 현재 멤버입니다. 계층구조가 블록 컨텍스트 안에 있지 않으면 수식이 빈 값을 반환합니다.

### 예제

[지역].[미국].[캘리포니아].Children은 [로스앤젤레스], [샌프란시스코], [샌디에이고]를 반환합니다.

[지역].Children은 [지역] 계층구조에서 현재 멤버가 [캘리포니아]인 경우 [로스앤젤레스], [샌프란시스코], [샌디에이고]를 반환합니다.

## 관련 정보

[Aggregate \[페이지 39\]](#)

[Average \[페이지 46\]](#)

[Count \[페이지 47\]](#)

[Max \[페이지 52\]](#)

[Min \[페이지 54\]](#)

[Sum \[페이지 74\]](#)

### 6.1.9.3 Depth

#### 설명

계층구조에 있는 멤버의 깊이를 반환합니다.

#### 함수 그룹

#### 집합

#### 구문

```
int member.Depth
```

#### 입력

매개 변수	설명	유형	필수
member	임의의 멤버	member	예

#### 참고

- 깊이란 계층구조의 최상위 수준에서 해당 멤버까지의 거리입니다.
- 계층구조의 최상위 수준은 0입니다.
- member는 계층구조의 현재 멤버입니다. 계층구조가 블록 컨텍스트 안에 있지 않으면 수식이 빈 값을 반환합니다.

예

계층구조 멤버의 깊이를 알아내려고 합니다.

```
=[Calendar].[Date.Calendar].Depth
```

Date.Calendar	'=[Calendar].[Date.Calendar].Depth
[-] All Periods	0
[-] CY 2001	1
[-] H2 CY 2001	2
[-] Q3 CY 2001	3
[-] July 2001	4
July 1, 2001	5
July 2, 2001	5
July 3, 2001	5
July 4, 2001	5
July 5, 2001	5
July 6, 2001	5
July 7, 2001	5

이제 Children 함수와 결합하여 매월 모든 일수가 포함되었는지 확인합니다.

```
=If [Calendar].[Date.Calendar].Depth = 4 Then Count([Internet Sales].[Internet Sales Amount];{[Calendar].[Date.Calendar].Children()})
```



Date.Calendar	Internet Sales Amount	[Date.Calendar].Depth	[Date.Calendar].Children()
[-] All Periods	29,358,677.22	0	
[-] CY 2001	3,266,373.66	1	
[-] H2 CY 2001	3,266,373.66	2	
[-] Q3 CY 2001	1,453,522.89	3	
[+] July 2001	473,388.16	4	31
[+] August 2001	506,191.69	4	30
[+] September 2001	473,943.03	4	29
[-] Q4 CY 2001	1,812,850.77	3	
[+] October 2001	513,329.47	4	30
[+] November 2001	543,993.41	4	30
[+] December 2001	755,527.89	4	31

## 6.1.9.4 Descendants

### 설명

집계 함수 안에서 사용되어 계층구조 멤버의 하위 항목을 반환합니다.

### 함수 그룹

### 집합

### 구문

```
member_set Descendants (member[;level|distance][;desc_flag])
```

### 입력

매개 변수	설명	유형	필수
member	임의의 멤버	멤버	예

매개 변수	설명	유형	필수
level	하위 항목의 수준	수준	아니요(기본값: member의 수준)
distance	현재 수준에서 하위 수준까지의 거리	int	아니요(기본값: member의 수준)
desc_flag	반환될 하위 멤버를 결정합니다.	키워드	아니요(기본값: Self)

## 참고

- Descendants는 독립 실행형 함수로 사용되지 않습니다. 집계 함수의 입력 매개 변수에서 집계에 대한 멤버 집합을 지정하는 경우 사용됩니다.
- member는 계층구조의 현재 멤버입니다. 계층구조가 블록 컨텍스트 안에 있지 않으면 수식이 빈 값을 반환합니다.
- desc\_flag의 Self는 level|distance 매개 변수에 지정된 수준을 참조합니다.
- desc\_flag의 Before는 level|distance 매개 변수에 지정된 수준보다 상위에 있는 모든 수준을 참조합니다.
- desc\_flag의 After는 level|distance 매개 변수에 지정된 수준보다 하위에 있는 모든 수준을 참조합니다.
- desc\_flag의 값은 다음과 같습니다.

Self	해당 수준에 있는 현재 멤버를 포함하여 level distance 매개 변수에 지정된 수준에 있는 하위 항목을 반환합니다.
이전	현재 멤버 및 level distance 매개 변수에 지정된 수준보다 상위에 있는 모든 하위 항목을 반환합니다.
이후	level distance 매개 변수에 지정된 수준보다 하위에 있는 하위 항목을 반환합니다.
Self_Before	현재 멤버 및 level distance 매개 변수에 지정된 수준에 있거나 이 수준보다 상위에 있는 모든 하위 항목을 반환합니다.
Self_After	현재 멤버 및 level distance 매개 변수에 지정된 수준에 있거나 이 수준보다 하위에 있는 모든 하위 항목을 반환합니다.
Before_After	현재 멤버 및 level distance 매개 변수에 지정된 수준에 있는 하위 항목을 제외한 모든 하위 항목을 반환합니다.
Self_Before_After	현재 멤버 및 모든 하위 항목을 반환합니다.
리프	현재 멤버와 level distance 매개 변수에 지정된 수준 사이에 있는 멤버 중 하위 멤버가 없는 모든 멤버를 반환합니다.

- distance는 양수여야 합니다.

예

사용하는 재무 계층구조를 구성하는 노드 중 누계 계산이 자동으로 되지 않는 노드의 하위 항목들을 합산하려고 합니다. 이 예에서는 각 대차대조표 멤버의 한 수준 아래 하위 항목들을 합산합니다.

```
=Sum([Query 3 (1)].[Financial Reporting].[Amount];{Descendants([Accounts]&[Balance Sheet];1)})
```

Accounts			
[-] Balance Sheet	0		27,481,462
[-] Assets	13,740,731		
[-] Liabilities and Owners Equity	13,740,731		
[-] Net Income	12,609,503		

```
=Sum([Query 3 (1)].[Financial Reporting].[Amount];{Descendants([Accounts]&[Balance Sheet].[Assets].[Current Assets];1;Leaves)})
```

[-] Balance Sheet	0		12,445,628
[-] Assets	13,740,731		
[-] Current Assets	12,445,628		
Cash	3,236,799		
[-] Receivables	3,475,923		
Trade Receivables	3,371,580		
Other Receivables	104,343		
Allowance for Bad Debt	67,429		
[-] Inventory	4,143,398		
Raw Materials	2,007,586		
Work in Process	1,393,582		
Finished Goods	742,230		
Deferred Taxes	505,424		
Prepaid Expenses	341,992		
Intercompany Receivable	674,663		

이제 유동자산에 속한 모든 멤버를 합산합니다.

```
=Sum([Query 3 (1)].[Financial Reporting].[Amount];{Descendants([Accounts]&[Balance Sheet].[Assets].[Current Assets];0;After)})
```

Balance Sheet	0	20,064,949
Assets	13,740,731	
Current Assets	12,445,628	
Cash	3,236,799	
Receivables	3,475,923	
Trade Receivables	3,371,580	
Other Receivables	104,343	
Allowance for Bad Debt	67,429	
Inventory	4,143,398	
Raw Materials	2,007,586	
Work in Process	1,393,582	
Finished Goods	742,230	
Deferred Taxes	505,424	
Prepaid Expenses	341,992	
Intercompany Receivable	674,663	

이제 유동자산 자체를 추가합니다.

```
=Sum([Query 3 (1)].[Financial Reporting].[Amount];{Descendants([Accounts]&[Balance Sheet].[Assets].[Current Assets];0;Self_After)})
```

Balance Sheet	0	32,510,577
Assets	13,740,731	
Current Assets	12,445,628	
Cash	3,236,799	
Receivables	3,475,923	
Trade Receivables	3,371,580	
Other Receivables	104,343	
Allowance for Bad Debt	67,429	
Inventory	4,143,398	
Raw Materials	2,007,586	
Work in Process	1,393,582	
Finished Goods	742,230	
Deferred Taxes	505,424	
Prepaid Expenses	341,992	
Intercompany Receivable	674,663	

## 관련 정보

[Aggregate \[페이지 39\]](#)

[Average \[페이지 46\]](#)

[Count \[페이지 47\]](#)

[Max \[페이지 52\]](#)

[Min \[페이지 54\]](#)

[Sum \[페이지 74\]](#)

## 6.1.9.5 IsLeaf

### 설명

멤버가 리프 멤버인지 여부를 확인합니다.

### 함수 그룹

### 기타

### 구문

```
bool member.IsLeaf
```

### 입력

매개 변수	설명	유형	필수
member	임의의 멤버	member	예

### 참고

- 리프 멤버는 하위가 없는 멤버입니다.
- member는 계층구조의 현재 멤버입니다. 계층구조가 블록 컨텍스트 안에 있지 않으면 수식이 빈 값을 반환합니다.

예

라인이 일자인지 확인합니다.

```
=[Calendar].[Date.Calendar].IsLeaf()
```

Date.Calendar	'=[Query 1].[Calendar].[Date.Calendar].IsLeaf
[-] All Periods	0
[-] CY 2001	0
[-] H2 CY 2001	0
[-] Q3 CY 2001	0
[-] July 2001	0
July 1, 2001	1
July 2, 2001	1
July 3, 2001	1
July 4, 2001	1
July 5, 2001	1
July 6, 2001	1
July 7, 2001	1
July 8, 2001	1

## 6.1.9.6 Key

설명

멤버의 키를 반환합니다.

구문

```
string member.Key
```

## 함수 그룹

### 집합

### 입력

매개 변수	설명	유형	필수
멤버	임의의 멤버	멤버	예

### 참고

- 키는 멤버의 내부 식별자입니다.
- `member`는 계층구조의 현재 멤버입니다. 계층구조가 블록 컨텍스트 안에 있지 않으면 수식이 빈 값을 반환합니다.

### 예

`[Geography].[US].Key`는 `[US]` 멤버의 키가 "XYZ"인 경우 "XYZ"를 반환합니다.

## 6.1.9.7 Lag

### 설명

집계 함수 내에서, 현재 멤버와 같은 수준에 있는 멤버를 반환하고 지정된 거리를 그 뒤에 표시합니다.

### 구문

```
member member.Lag(distance)
```

## 함수 그룹

### 집합

## 입력

매개 변수	설명	유형	필수
member	임의의 멤버	member	예
distance	현재 멤버를 기준으로 한 멤버 거리	int	예

## 참고

- Lag는 독립 실행형 함수로 사용되지 않습니다. 집계 함수의 입력 매개 변수에서 집계에 대한 멤버 집합을 지정하는 경우 사용됩니다.
- distance가 양수이면 Lag에서 멤버가 반환되고 distance가 member 뒤에 위치합니다. distance가 음수이면 Lag에서 멤버가 반환되고 distance가 member 앞에 놓이게 됩니다.
- member는 계층구조의 현재 멤버입니다. 계층구조가 블록 컨텍스트 안에 있지 않으면 수식이 빈 값을 반환합니다.
- Lag는 계층구조 내의 멤버 순서 및 쿼리를 사용하여 관련 멤버를 반환합니다.

## 예

이번 주와 저번 주의 인터넷 매출액의 차이를 알아내려고 합니다.

```
=Max([Internet Sales].[Internet Sales Amount];{[Calendar].[Date.Calendar].Lag(7)})
```

Date.Calendar	Internet Sales Amount	=Max([Query 1].[Internet Sales].[Internet Sales Amount];[Query 1].[Calendar].[Date.Calendar].Lag(7))
All Periods	29,358,677.22	
CY 2001	3,266,373.66	
H2 CY 2001	3,266,373.66	
Q3 CY 2001	1,453,522.89	1,623,971.06
July 2001	473,388.16	550,816.69
July 1, 2001	14,477.34	7,855.64
July 2, 2001	13,931.52	20,909.78
July 3, 2001	15,012.18	10,556.53
July 4, 2001	7,156.54	14,313.08
July 5, 2001	15,012.18	14,134.8
July 6, 2001	14,313.08	7,156.54
July 7, 2001	7,855.64	25,047.89
July 8, 2001	7,855.64	11,230.63
July 9, 2001	20,909.78	14,313.08
July 10, 2001	10,556.53	14,134.8

특정 연도를 그 2년 전의 연도와 비교하려고 합니다.



Date.Calendar	Internet Sales Amount	
[-] All Periods	29,358,677.22	
[-] CY 2001	3,266,373.66	11.13%
[+] H2 CY 2001	3,266,373.66	100.00%
[-] CY 2002	6,530,343.53	22.24%
[+] H1 CY 2002	3,805,710.59	58.28%
[+] H2 CY 2002	2,724,632.94	41.72%
[-] CY 2003	9,791,060.3	33.35%
[+] H1 CY 2003	3,037,501.36	31.02%
[+] H2 CY 2003	6,753,558.94	68.98%
[+] CY 2004	9,770,899.74	33.28%

CY 2002	CY 2002.Lag(2)	CY 2002 - CY 2002.Lag(2)
6,530,343.53	9,770,899.74	-3,240,556.21

이제 Lag와 IsLeaf를 결합하여 1주일 동안의 매출액 차이를 알아보려고 합니다. 마지막 열에 다음과 같은 수식을 설정하면 됩니다.

```
=If [Calendar].[Date.Calendar].IsLeaf() Then [Internet Sales].[Internet Sales Amount] - Max([Internet Sales].[Internet Sales Amount];{[Calendar].[Date.Calendar].Lag(7)})
```

Date.Calendar	Internet Sales Amount	[Calendar].[Date.Calendar].Lag(7))	Difference week to week
All Periods	29,358,677.22		
CY 2001	3,266,373.66		
H2 CY 2001	3,266,373.66		
Q3 CY 2001	1,453,522.89	1,623,971.06	
July 2001	473,388.16	550,816.69	
July 1, 2001	14,477.34	7,855.64	6,621.7
July 2, 2001	13,931.52	20,909.78	-6,978.26
July 3, 2001	15,012.18	10,556.53	4,455.65
July 4, 2001	7,156.54	14,313.08	-7,156.54
July 5, 2001	15,012.18	14,134.8	877.38
July 6, 2001	14,313.08	7,156.54	7,156.54
July 7, 2001	7,855.64	25,047.89	-17,192.25
July 8, 2001	7,855.64	11,230.63	-3,374.99
July 9, 2001	20,909.78	14,313.08	6,596.7
July 10, 2001	10,556.53	14,134.8	-3,578.27
July 11, 2001	14,313.08	6,953.26	7,359.82

## 관련 정보

[Aggregate \[페이지 39\]](#)

[Average \[페이지 46\]](#)

[Count \[페이지 47\]](#)

[Max \[페이지 52\]](#)

[Min \[페이지 54\]](#)

[Sum \[페이지 74\]](#)

## 6.1.9.8 Parent

### 설명

집계 함수 안에서 사용되어 계층구조 멤버의 상위 멤버를 반환합니다.

### 함수 그룹

#### 집합

## 구문

```
member member.Parent
```

## 입력

매개 변수	설명	유형	필수
member	임의의 멤버	멤버	예

## 참고

- Parent는 독립 실행형 함수로 사용되지 않습니다. 집계 함수의 입력 매개 변수에서 집계에 대한 멤버 집합을 지정하는 경우 사용됩니다.
- member는 계층구조의 현재 멤버입니다. 계층구조가 블록 컨텍스트 안에 있지 않으면 수식이 빈 값을 반환합니다.

## 예제

두 번째 열에는 각 계층구조 멤버의 상위 멤버를 가져올 수 있는 수식이 포함되어 있습니다.

```
=Max([Customer Geography];{[Customer Geography].Parent})
```

[-] All Customers	
[-] Australia	All Customer:
[-] New South Wales	Australia
[+] Alexandria	New South W
[-] Coffs Harbour	New South W
[-] 2450	Coffs Harbou
Adriana Smith	2450
Aimee Guo	2450
Allison R. Young	2450
Ann A. Sara	2450

## 관련 정보

[Aggregate \[페이지 39\]](#)

[Average \[페이지 46\]](#)

[Count \[페이지 47\]](#)

[Max \[페이지 52\]](#)

[Min \[페이지 54\]](#)

[Sum \[페이지 74\]](#)

## 6.1.9.9 Siblings

### 설명

집계 함수 안에서 사용되어 계층구조 멤버의 형제 멤버를 반환합니다.

### 함수 그룹

### 집합

### 구문

```
member_set member.Siblings
```

### 입력

매개 변수	설명	유형	필수
member	임의의 멤버	멤버	예

### 참고

- Siblings는 독립 실행형 함수로 사용되지 않습니다. 집계 함수의 입력 매개 변수에서 집계에 대한 멤버 집합을 지정하는 경우 사용됩니다.
- member는 계층구조의 현재 멤버입니다. 계층구조가 블록 컨텍스트 안에 있지 않으면 수식이 빈 값을 반환합니다.

- 형제 멤버는 동일한 수준에 있는 멤버로서 member와 동일한 상위를 갖습니다.

## 예제

시간 계층구조를 통해 특정 연도 안에서 각 분기가 차지하는 비율 또는 일정 기간 안에서 각 연도가 차지하는 비율을 알아내려고 합니다.

```
=[Query 1].[Internet Sales].[Internet Sales Amount] / Sum([Query 1].[Internet Sales].[Internet Sales Amount];{[Query 1].[Calendar].[Date.Calendar].Siblings()})
```

Date.Calendar	Internet Sales Amount	
[-] All Periods	29,358,677.22	
[-] CY 2001	3,266,373.66	11.13%
[+] H2 CY 2001	3,266,373.66	100.00%
[-] CY 2002	6,530,343.53	22.24%
[+] H1 CY 2002	3,805,710.59	58.28%
[+] H2 CY 2002	2,724,632.94	41.72%
[-] CY 2003	9,791,060.3	33.35%
[+] H1 CY 2003	3,037,501.36	31.02%
[+] H2 CY 2003	6,753,558.94	68.98%
[+] CY 2004	9,770,899.74	33.28%

자유 형식 셀에서, 2004년이 전체 기간에서 차지하는 비율을 알아내려고 합니다.

```
=Sum([Query 1].[Internet Sales].[Internet Sales Amount];{[Query 1].[Calendar].[Date.Calendar]&[All Periods].[CY 2004]}) / Sum([Query 1].[Internet Sales].[Internet Sales Amount];{[Query 1].[Calendar].[Date.Calendar]&[All Periods].[CY 2004].Siblings()})
```

{CY 2001;CY 2002}	2004 percentage in 2001 to 2004 perdioid
9,796,717.18	33.28%

Date.Calendar	Internet Sales Amount	
[-] All Periods	29,358,677.22	
[-] CY 2001	3,266,373.66	11.13%
[+] H2 CY 2001	3,266,373.66	100.00%
[-] CY 2002	6,530,343.53	22.24%
[+] H1 CY 2002	3,805,710.59	58.28%
[+] H2 CY 2002	2,724,632.94	41.72%
[-] CY 2003	9,791,060.3	33.35%
[+] H1 CY 2003	3,037,501.36	31.02%
[+] H2 CY 2003	6,753,558.94	68.98%
[+] CY 2004	9,770,899.74	33.28%

## 관련 정보

[Aggregate \[페이지 39\]](#)

[Average \[페이지 46\]](#)

[Count \[페이지 47\]](#)

[Max \[페이지 52\]](#)

[Min \[페이지 54\]](#)

[Sum \[페이지 74\]](#)

## 6.1.10 기타 함수

### 6.1.10.1 BlockName

#### 설명

블록 이름을 반환합니다.

#### 함수 그룹

#### 기타

#### 구문

```
string BlockName()
```

#### 예제

BlockName() 이 "Block1"이라는 이름의 블록에 있는 경우 "Block1"을 반환합니다.

### 6.1.10.2 ColumnNumber

#### 설명

열 번호를 반환합니다.

#### 함수 그룹

#### 기타

## 구문

```
int ColumnNumber()
```

## 예제

ColumnNumber()는 수식이 테이블의 두 번째 열에 있는 경우 2를 반환합니다.

### 6.1.10.3 CurrentUser

## 설명

현재 사용자의 BI 실행 패드 로그인을 반환합니다.

## 함수 그룹

## 기타

## 구문

```
string CurrentUser()
```

## 예제

CurrentUser()는 현재 사용자의 로그인이 "gkn"인 경우 "gkn"을 반환합니다.

### 6.1.10.4 ForceMerge

## 설명

차원이 계수의 계산 컨텍스트에 있지 않을 경우 동기화된 차원을 계수 계산에 포함시킵니다.



## 함수 그룹

기타

## 구문

```
num ForceMerge(measure)
```

## 입력

매개 변수	설명	유형	필수
measure	임의의 계수	계수	예

## 출력

동기화된 차원이 고려된 계산 결과

## 참고

- ForceMerge는 스마트 계수에 적용되면 스마트 계수 계산에 필요한 그룹화 집합이 존재하지 않기 때문에 #MULTIVALUE를 반환합니다.
- ForceMerge는 BusinessObjects/Desktop Intelligence의 Multicube 함수에 해당합니다.

## 예제

ForceMerge([수익])은 [수익] 계수와 같은 블록에 나타나지 않는 동기화된 차원을 고려하여 [수익] 값을 반환합니다.

## 6.1.10.5 GetContentLocale

### 설명

문서에 포함된 데이터의 로캘(문서 로캘)을 반환합니다.

---

함수 그룹

기타

구문

```
string GetContentLocale()
```

참고

문서 로캘은 문서의 데이터 서식을 지정하는 데 사용됩니다.

예제

GetContentLocale() 은 문서 로캘이 "프랑스어(프랑스)"인 경우 "fr\_FR"을 반환합니다.

## 6.1.10.6 GetDominantPreferredViewingLocale

설명

사용자의 기본 설정 보기 로캘 그룹에서 주요 로캘을 반환합니다.

함수 그룹

기타

구문

```
string GetDominantPreferredViewingLocale()
```

## 참고

- 관련 로캘의 각 그룹에는 그룹 내 다른 모든 로캘에 대한 기준으로 사용되는 주요 로캘이 포함되어 있습니다. 예를 들어, 미국 영어("en\_US")는 영어 로캘 그룹의 주요 로캘입니다. 뉴질랜드 영어("en\_NZ")도 이 그룹에 속합니다.
- *Translation Manager* 가이드에 주요 기본 설정 보기 로캘이 모두 나열되어 있습니다.

## 예제

`GetDominantPreferredViewingLocale`은 기본 설정 보기 로캘이 "영어(뉴질랜드)"인 경우 "en\_US"를 반환합니다.

## 관련 정보

[GetPreferredViewingLocale \[페이지 205\]](#)

## 6.1.10.7 GetLocale

### 설명

사용자 인터페이스 형식을 지정하는 데 사용되는 사용자의 로캘(제품 로캘)을 반환합니다.

### 함수 그룹

### 기타

### 구문

```
string GetLocale()
```

## 참고

제품 로캘은 사용자 인터페이스(예: 메뉴 항목 및 단추 텍스트)의 로캘입니다.

## 예제

`GetLocale()` 은 사용자의 제품 로캘이 "영어(미국)"인 경우 "en\_US"를 반환합니다.

## 6.1.10.8 GetLocalized

### 설명

사용자의 기본 설정 보기 로캘에 따라 지역화된 문자열을 반환합니다.

### 구문

```
string GetLocalized(string[,comment])
```

### 입력

매개 변수	설명	유형	필수
string	번역될 문자열	문자열	예
comment	번역사를 위한 설명	문자열	아니요

### 참고

- `string` 매개 변수는 수식의 문자열일 수 있습니다(예: 셀의 경고 메시지 또는 변수 정의).
- 보고서를 디자인할 때 `comment` 매개 변수를 사용하여 번역사의 문자열 번역을 지원하기 위한 추가 정보를 제공할 수 있습니다. `comment`는 번역사가 보고서를 번역하는 데 사용하는 Translation Manager 도구에서 `string`과 함께 나타납니다.
- 각 `string + comment` 쌍은 Translation Manager 도구에서 번역될 별도의 문자열을 생성합니다. 따라서 `GetLocalized("Product Total";"Max 20 characters")`와 `GetLocalized("Product Total";"Use no more than 20 characters")`는 다른 번역을 반환할 수 있습니다.

## 예제

`GetLocalized("Total for all products")`는 기본 설정 보기 로캘이 "fr\_FR"인 경우 "모든 제품에 대한 합계"의 프랑스어 번역을 반환합니다.

---

`GetLocalized("Total for all products"; "Try not to use more than 20 characters")`는 기본 설정 보기 로캘이 "de\_DE"인 경우 "모든 제품에 대한 합계"의 독일어 번역을 반환합니다. 또한 이 함수는 문자열을 번역할 때 가능하면 20자 이상을 사용하지 않도록 보고서 번역사에게 알립니다.

## 관련 정보

[GetPreferredViewingLocale \[페이지 205\]](#)

### 6.1.10.9 GetPreferredViewingLocale

#### 설명

문서 데이터를 보기 위한 사용자의 기본 설정 로캘(기본 설정 보기 로캘)을 반환합니다.

#### 함수 그룹

#### 기타

#### 구문

```
string GetPreferredViewingLocale()
```

#### 예제

`GetPreferredViewingLocale`은 기본 설정 보기 로캘이 "영어(미국)"인 경우 "en\_US"를 반환합니다.

## 관련 정보

[GetLocalized \[페이지 204\]](#)

[GetDominantPreferredViewingLocale \[페이지 202\]](#)

## 6.1.10.10 If...Then...Else

### 설명

식이 true인지 또는 false인지를 기반으로 값을 반환합니다.

### 함수 그룹

### 기타

### 구문

```
If bool_value Then true_value [Else false_value]
```

### 입력

매개 변수	설명	유형	필수
bool_value	부울 값	부울	예
true_value	bool_value가 true인 경우의 반환 값	임의의 값	예
false_value	bool_value가 false인 경우의 반환 값	임의의 값	예(Else가 포함된 경우)

### 참고

- true\_value 및 false\_value는 데이터 형식을 혼용할 수 있습니다.
- If 함수에 부울 연산자 And, Between, InList, Or 및 Not을 함께 사용할 수 있습니다.
- Else 절을 ElseIf 절로 바꿔 If 조건을 중첩할 수 있습니다. 이 구문은 다음과 같이 한 수준의 중첩을 나타냅니다.

```
If bool_value Then true_value [ElseIf bool_value Then true_value Else false_value...]
```

- If 함수의 원본 구문(If(bool\_value, true\_value, false\_value))도 지원됩니다.

## 예제

If [판매 수익]>1000000 Then "High Revenue"는 해당 수익이 1,000,000보다 큰 모든 행에 대해 "High Revenue"를 반환하고 다른 모든 행에 대해서는 아무 것도 반환하지 않습니다.

If [판매 수익] >1000000 Then "High Revenue" Else [수익]은 해당 수익이 1,000,000보다 큰 모든 행에 대해 "High Revenue"를 반환하고 다른 모든 행에 대해서는 수익 값을 반환합니다.

If [판매 수익]>1000000 Then "High Revenue" Else "Low Revenue"는 해당 수익이 1,000,000보다 큰 모든 행에 대해 "High Revenue"를 반환하고 해당 수익이 1,000,000보다 작은 모든 행에 대해 "Low Revenue"를 반환합니다.

If [판매 수익]>1000000 Then "High Revenue" ElseIf [판매 수익] > 800000 Then "Medium Revenue" Else "Low Revenue"는 해당 수익이 1,000,000보다 큰 모든 행에 대해 "High Revenue"를, 해당 수익이 800,000과 1,000,000 사이인 모든 행에 대해 "Medium Revenue"를, 그리고 다른 모든 행에 대해 "Low Revenue"를 반환합니다.

## 관련 정보

[If \[페이지 207\]](#)

[And 연산자 \[페이지 221\]](#)

[Between 연산자 \[페이지 223\]](#)

[InList 연산자 \[페이지 224\]](#)

[Or 연산자 \[페이지 222\]](#)

[Not 연산자 \[페이지 223\]](#)

## 6.1.10.11 If

### 설명

식이 true인지 또는 false인지를 기반으로 값을 반환합니다.

### 함수 그룹

### 기타

## 구문

```
If (bool_value; true_value; false_value)
```

## 입력

매개 변수	설명	유형	필수
bool_value	부울 값	부울	예
true_value	bool_value가 true인 경우의 반환 값	임의의 값	예
false_value	bool_value가 false인 경우의 반환 값	임의의 값	예

## 참고

- true\_value 및 false\_value는 데이터 형식을 혼용할 수 있습니다.
- false\_value를 추가 If 조건식으로 교체하여 If 조건문을 중첩할 수 있습니다. 이 구문은 다음과 같이 한 수준의 중첩을 표시합니다.

```
If (bool_value; true_value; If (bool_value; true_value; false_value); false_value)
```

- If...Then...Else 구문도 지원됩니다.

## 예제

If ([판매 수익]>1000000;"High Revenue";"Low Revenue")는 해당 수익이 1,000,000보다 큰 모든 행에 대해 "High Revenue"를 반환하고 해당 수익이 1,000,000 이하인 모든 행에 대해 "Low Revenue"를 반환합니다.

If ([판매 수익]>1000000;"High Revenue";[수익])은 해당 수익이 1,000,000보다 큰 모든 행에 대해 "High Revenue"를 반환하고 다른 모든 행에 대해서는 수익 값을 반환합니다.

## 관련 정보

[If...Then...Else \[페이지 206\]](#)



## 6.1.10.12 LineNumber

### 설명

테이블의 행 번호를 반환합니다.

### 함수 그룹

### 기타

### 구문

```
int LineNumber()
```

### 참고

테이블의 줄 번호 매기기는 머리글에서부터 시작합니다.

### 예제

LineNumber()는 함수가 테이블의 두 번째 줄에 나타나는 경우 2를 반환합니다.

## 6.1.10.13 NameOf

### 설명

개체의 이름을 반환합니다.

### 함수 그룹

### 기타

## 구문

```
string NameOf(obj)
```

## 입력

매개 변수	설명	유형	필수
obj	모든 보고서 개체	보고서 개체	예

## 참고

NameOf 함수는 보고서에서 열과 행의 머리글에 나타납니다.

## 예제

NameOf ([예약 날짜]) 는 "예약 날짜"를 반환합니다.

## 6.1.10.14 NoFilter

### 설명

값 계산 시 필터를 무시합니다. NoFilter는 계수 개체와 함께 사용되며 차원에는 적용되지 않습니다.

### 함수 그룹

### 기타

## 구문

```
input_type NoFilter(obj[;All|Drill])
```

## 입력

매개 변수	설명	유형	필수
obj	모든 보고서 개체	보고서 개체	예
All Drill	<ul style="list-style-type: none"><li>키워드 지정되지 않음 - 보고서 및 블록 필터 무시</li><li>All - 모든 필터 무시</li><li>Drill - 보고서 및 드릴 필터 무시</li></ul>	키워드	아니요

## 참고

- 드릴 필터가 보고서 데이터에 적용된 것과 다른 쿼리에 추가되므로 `NoFilter(obj;Drill)`는 쿼리 드릴 모드에서는 작동하지 않습니다.
- 드릴 필터가 적용된 상태로 드릴 모드를 끝내면 드릴 필터가 보고서 필터가 되며 `NoFilter(obj;Drill)`가 적용된 개체의 값이 변경될 수 있습니다.

## 예제

블록 바닥글에 있는 경우 `NoFilter(Sum([판매 수익]))`은 행이 블록 범위 밖에서 필터링된 경우에도 블록에 있는 가능한 모든 행의 총 판매 수익을 반환합니다.

`NoFilter(Sum([판매 수익]);All)`은 보고서에서 프랑스를 제외하는 필터가 있더라도 프랑스를 포함한 모든 국가의 판매 수익 합계를 반환합니다.

`NoFilter(Sum([판매 수익]);Drill)`은 [국가] 차원에 드릴 필터가 있는 경우에도 모든 국가에 대한 총 판매 수익을 반환합니다.

## 6.1.10.15 NumberOfPages

### 설명

보고서의 페이지 수를 반환합니다.

### 함수 그룹

### 기타

## 구문

```
integer NumberOfPages()
```

## 예제

보고서에 두 페이지가 있는 경우 `NumberOfDataPages()` 는 2를 반환합니다.

## 6.1.10.16 페이지

### 설명

보고서의 현재 페이지 번호를 반환합니다.

### 함수 그룹

### 기타

## 구문

```
integer Page()
```

## 예제

`Page()` 는 보고서의 두 번째 페이지에 나타나는 경우 2를 반환합니다.

## 6.1.10.17 Previous

### 설명

개체의 이전 값을 반환합니다.

## 함수 그룹

기타

## 구문

```
input_type Previous (dimension|measure|Self [;Row|col] [; (reset_dims)] [;offset] [;NotNull])
```

## 입력

매개 변수	설명	유형	필수
dimension measure Self	함수에서 반환한 이전 값의 차원 또는 계수이거나, 자체 키워드입니다.	차원, 계수 또는 키워드	예
Row/Col	계산 방향을 설정합니다.	키워드	아니요
reset_dims	계산을 다시 설정하는 데 사용되는 차원 목록	차원 목록	아니요
offset	현재 행 앞의 offset 행인 <b>차원</b> 또는 <b>계수</b> 의 값을 지정합니다.	정수	아니요(기본값은 1)
NotNull	함수에 offset에서 시작하여 Null이 아닌 첫 번째 값을 반환하도록 지시합니다.	키워드	아니요

## 참고

- offset의 기본 값은 1입니다. Previous([수익];1)와 Previous([수익])는 기능이 같습니다.
- NotNull 인수가 포함되어 있는 경우, 함수는 현재 행 앞의 셀 offset 행부터 시작하여 역방향으로 진행하면서 Null이 아닌 개체의 첫 번째 값을 반환합니다.
- Previous 함수와 함께 확장 구문 컨텍스트 연산자를 사용할 수 있습니다.
- 셀에 한 보고서 개체 이외의 내용이 포함된 경우, Self 연산자를 사용하여 해당 셀의 이전 값을 참조할 수 있습니다.
- 재설정 차원 목록에 차원이 하나만 있는 경우에도 항상 차원을 괄호로 묶어야 합니다.
- 재설정 차원 집합을 지정하는 경우 세미콜론으로 구분해야 합니다.
- Previous는 모든 보고서, 섹션 및 블록 필터, 그리고 모든 정렬이 적용된 뒤에 적용됩니다.
- Previous를 사용하는 수식에는 정렬 또는 필터를 적용할 수 없습니다.
- 계수에 Previous를 적용하고 해당 계수가 정의되지 않은 값을 반환하는 경우, Previous는 앞줄에서 값을 반환한 경우에도 정의되지 않은 값을 반환합니다.

- Previous는 나누기 머리글이나 나누기 바닥글 밖에 배치된 나누기를 무시합니다.
- Previous는 나누기 바닥글에 있는 경우 바닥글의 이전 인스턴스에 있는 값을 반환합니다.
- Previous는 각 보고서 섹션에서 재설정됩니다.
- Previous가 크로스탭에서 사용되는 경우에는 행의 마지막 값을 다음 행의 첫 번째 값의 이전 값으로 처리하지 않습니다.

## 예제

Previous([국가];1)는 아래 표에 다음 값을 반환합니다.

국가	수익	이전
미국	5,000,000	
영국	2,000,000	미국
프랑스	2,100,000	영국

Previous([수익])는 아래 표에 다음 값을 반환합니다.

국가	수익	이전
미국	5,000,000	
영국	2,000,000	5,000,000
프랑스	2,100,000	2,000,000

Previous([수익];([국가]))는 아래 표에 다음 값을 반환합니다.

국가	지역	수익	이전
미국	북부	5,000,000	
	남부	7,000,000	5,000,000
영국	북부	3,000,000	
	남부	4,000,000	3,000,000

Previous([수익])는 아래 크로스탭에 다음 값을 반환합니다.

	2004	이전	2005	이전
미국	5,000,000		6,000,000	5,000,000
영국	2,000,000		2,500,000	2,000,000
프랑스	3,000,000		2,000,000	3,000,000

Previous([수익])는 아래 표에 [국가]를 나누어 다음 값을 반환합니다.

국가	지역	수익	이전
----	----	----	----

미국	북부	5,000,000	
	남부	7,000,000	5,000,000
미국		12,000,000	

국가	지역	수익	이전
영국	북부	3,000,000	7,000,000
	남부	4,000,000	3,000,000
영국		7,000,000	12,000,000

Previous([수익]);2;NotNull)는 아래 표에 다음 값을 반환합니다.

연도	분기	수익	이전
2008	1분기	500	
2008	2분기		
2008	3분기	400	500
2008	4분기	700	500
2008	1분기	300	400
2008	2분기		700
2008	3분기		300
2008	4분기	200	300

2\*Previous(Self)는 2, 4, 6, 8, 10... 시퀀스를 반환합니다.

## 관련 정보

[Previous 함수를 사용하여 값 비교 \[페이지 252\]](#)

[Self 연산자 \[페이지 232\]](#)

## 6.1.10.18 RefValue

### 설명

데이터 추적이 활성화된 보고서 개체의 참조 값을 반환합니다.

## 함수 그룹

## 기타

## 구문

```
input_type RefValue(obj)
```

## 예제

`RefValue([최고 실적 지역])`는 참조 데이터에서 [최고 실적 지역] 변수 값이 "South West"인 경우 "South West"를 반환합니다.

`RefValue([수익])`는 참조 데이터에서 [수익] 계수 값이 1000인 경우 1000을 반환합니다.

## 참고

- `RefValue()` 함수는 계수 개체나 차원 개체와 함께 사용될 수 있습니다. 하지만 차원이나 설명으로 정규화된 변수에 사용되는 경우에는 `RefValue()` 함수를 통해 참조 값이 아닌 해당 개체의 현재 값이 반환됩니다. 참조 값이 반환되게 하려면 변수가 계수로 정규화되어야 합니다.
- 섹션, 테이블, 양식 또는 차트에서 수식이 직접 만들어질 때는 이 수식이 항상 계수로 정규화되므로 수식에 `RefValue()` 함수가 사용될 경우 적합한 참조 값이 반환됩니다.

## 변수가 있는 **RefValue** 함수의 예

[주] 차원에 대해 캘리포니아, 플로리다, 텍사스, 뉴욕 값 목록이 있습니다. 데이터를 새로 고치고 나면 애리조나, 캘리포니아, 플로리다, 텍사스, 뉴욕으로 목록이 바뀝니다. `Variable=RefValue([State])` 같은 변수는 다음을 반환합니다.

변수 정규화 대상	반환되는 값 목록
차원 또는 설명	애리조나, 캘리포니아, 플로리다, 텍사스, 뉴욕
계수	(null 값), 캘리포니아, 플로리다, 텍사스, 뉴욕



## 6.1.10.19 RelativeValue

### 설명

개체의 이전 또는 이후 값을 반환합니다.

### 함수 그룹

### 기타

### 구문

```
input_type RelativeValue(measure|detail;slicing_dims;offset)
```

### 입력

매개 변수	설명	유형	필수
measure detail	블록에 있는 임의의 계수 또는 차원의 세부 정보입니다.	계수 또는 세부 정보	예
slicing_dims	계산 컨텍스트를 제공하는 차원입니다.	차원 목록	예
offset	현재 행에서 제거된 offset 행인 measure 또는 detail의 값을 지정합니다.	정수	예

### 참고

- 개체는 블록에서 사용 가능한 계수 또는 차원 세부 정보여야 합니다.
- 분리 차원의 값 목록 정렬 순서는 함수의 출력을 결정하는 데 사용됩니다.  
정렬 순서는 다음 두 가지 요소에 의해 결정됩니다. 분리 차원에 적용되는 정렬과 분리 차원이 함수에 나열되는 순서입니다.
- 섹션 마스터로 사용되는 차원은 분리 차원으로 지정할 수 있습니다.
- 모든 분리 차원은 함수가 위치한 블록 또는 블록의 섹션 머리글에 있어야 합니다. 분리 차원이 이후에 블록에서 제거되는 경우 함수에서 #COMPUTATION 오류를 반환합니다.
- offset이 분리 차원의 값 목록에 있는 행 수를 초과하는 경우, 함수에서 Null을 반환합니다.

- RelativeValue는 재귀적으로 사용할 수 없습니다.
- 분리 차원 목록에 차원이 하나만 있는 경우에도 항상 차원을 괄호로 묶어야 합니다.

## 예제

아래 표의 RelativeValue 열에는 다음 수식이 포함됩니다.

```
RelativeValue([수익];([연도]);-1)
```

연도	분기	판매 직원	수익	RelativeValue
2007	Q1	Smith	1000	
2007	Q2	Jones	2000	
2007	Q3	Wilson	1500	
2007	Q4	Harris	3000	
2008	Q1	Smith	4000	1000
2008	Q2	Jones	3400	2000
2008	Q3	Wilson	2000	1500
2008	Q4	Harris	1700	3000

## 관련 정보

[#COMPUTATION \[페이지 246\]](#)

[RelativeValue 함수를 사용하여 값 비교 \[페이지 252\]](#)

## 6.1.10.20 ReportName

### 설명

보고서의 이름을 반환합니다.

### 함수 그룹

### 기타

## 구문

```
string ReportName()
```

## 예제

ReportName()은 "판매 보고서"라는 보고서에 있는 경우 "판매 보고서"를 반환합니다.

## 6.1.10.21 RowIndex

### 설명

행 수를 반환합니다.

### 함수 그룹

### 기타

## 구문

```
integer RowIndex()
```

## 참고

- 행 번호 매기기는 0에서 시작합니다.
- 테이블 머리글 또는 바닥글에 있는 경우 RowIndex는 #MULTIVALUE를 반환합니다.

## 예제

RowIndex는 테이블의 첫 번째 행에 나타나는 경우 0을 반환합니다.

## 6.1.10.22 UniqueNameOf

### 설명

개체의 고유 이름을 반환합니다.

### 함수 그룹

### 기타

### 구문

```
string UniqueNameOf(obj)
```

### 입력

매개 변수	설명	유형	필수
obj	모든 보고서 개체	보고서 개체	예

### 예제

UniqueNameOf([예약 날짜])는 "예약 날짜"를 반환합니다.

## 6.2 함수 및 수식 연산자

연산자는 수식의 다양한 구성 요소를 연결합니다.

수식에는 수학, 조건부, 논리, 함수별 또는 확장 구문 연산자가 포함될 수 있습니다.

### 6.2.1 수학 연산자

수학 연산자는 일상적인 산술과 비슷합니다.

수식에서 수학 연산을 수행하는 데는 더하기(+), 빼기(-), 곱하기(\*), 나누기(/) 연산자를 사용할 수 있습니다. 수식 [판매 수익] - [판매 비용]에는 수학 연산자인 빼기(-)가 포함되어 있습니다.

#### i 노트

'+' 연산자는 문자열과 함께 사용할 경우 문자열 연결 연산자가 됩니다. 즉, 문자열을 결합합니다. 예를 들어, 수식 "John" + " Smith"는 "John Smith"를 반환합니다.

## 6.2.2 조건부 연산자

조건부 연산자는 값 사이의 비교 유형을 결정합니다.

연산자	설명
=	같음
>	보다 큼
<	보다 작음
>=	크거나 같음
<=	작거나 같음
<>	같지 않음

다음과 같이 If 함수와 함께 조건부 연산자를 사용합니다.

```
If [수익]>10000 Then "높음" Else "낮음"
```

이 경우 수익이 10000 이상인 모든 행에 대해 "High"가 반환되고 다른 모든 행에 대해서는 "Low"가 반환됩니다.

## 6.2.3 논리 연산자

논리 연산자에는 And, Or, Not, Between 및 InList가 있습니다.

논리 연산자는 True 또는 False를 반환하는 부울 식에서 사용됩니다.

### 6.2.3.1 And 연산자

And 연산자는 부울 값을 연결합니다.

## 설명

And로 연결된 모든 부울 값이 true를 반환하는 경우 모든 값의 조합에서도 true를 반환합니다.

## 구문

```
bool_value And bool_value [And bool_value...]
```

## 예제

If [리조트] = "Bahamas Beach" And [수익]>100000 Then "High Bahamas Revenue"는 [휴양지] = "바하마 해변"인 동시에 [수익]>100000인 경우 "높은 바하마 수익"을 반환합니다.

### 6.2.3.2 Or 연산자

Or 연산자는 부울 값을 연결합니다.

## 설명

Or로 연결된 부울 값 중 하나가 true를 반환하는 경우 다른 모든 값의 조합도 true를 반환합니다.

## 구문

```
bool_value Or bool_value [Or bool_value...]
```

## 예제

If [휴양지] = "바하마 해변" Or [휴양지]="하와이안 클럽" Then "미국" Else "프랑스"는 [휴양지]="바하마 해변" 또는 "하와이안 클럽"인 경우 "미국"을 반환하고, 그렇지 않은 경우 "프랑스"를 반환합니다.

### 6.2.3.3 Not 연산자

#### 설명

Not 연산자는 부울 값의 반대 값을 반환합니다.

#### 구문

```
bool Not (bool_value)
```

#### 예제

If Not ([국가] = "미국") Then "미국 아님"은 [국가]에 "미국" 이외의 값이 있을 경우 "미국 아님"을 반환합니다.

### 6.2.3.4 Between 연산자

#### 설명

Between 연산자는 변수가 두 값 사이에 있는지 확인합니다.

#### 구문

```
bool Between (first_value; second_value)
```

#### 참고

- If 함수 및 Where 연산자에 Between 연산자를 사용합니다.
- 문서 로컬을 변경할 경우 Between 연산자가 반환하는 결과에 영향을 미칠 수 있습니다.

예

If [판매 수익] Between (800000;900000) Then "Medium Revenue"는 [판매 수익]이 800000과 900000 사이에 있는 경우 "중간 수익"을 반환합니다.

[판매 수익] Between (10000;20000)은 판매 수익이 10000과 20000 사이에 있는 경우 true를 반환합니다.

If ([판매 수익] Between (200000;500000); "Medium Revenue"; "Low/High Revenue")은 [판매 수익]이 300000인 경우 "중간 수익"을 반환합니다.

## 관련 정보

[If...Then...Else \[페이지 206\]](#)

[Where 연산자 \[페이지 233\]](#)

## 6.2.3.5 InList 연산자

### 설명

InList 연산자는 값이 값 목록에 있는지 확인합니다.

### 구문

```
bool test_value InList (value_list)
```

### 참고

InList 연산자는 InList 단독이 아닌 부울 값을 반환하는 test\_value와 InList의 조합입니다.

### 예제

If Not ([국가] InList ("영국"; "스코틀랜드"; "웨일스")) Then "대영 제국 아님" Else "대영 제국"은 [국가]가 "영국", "스코틀랜드" 또는 "웨일스"가 아닌 경우 "대영 제국 아님"을 반환하고, 그렇지 않은 경우 "대영 제국"을 반환합니다.



---

If [휴양지] InList("바하마 해변"; "하와이안 클럽") Then "미국 휴양지"는 [휴양지]가 "바하마 해변" 또는 "하와이안 클럽"인 경우 "미국 휴양지"를 반환합니다.

## 관련 정보

[If...Then...Else \[페이지 206\]](#)

[Where 연산자 \[페이지 233\]](#)

## 6.2.4 함수별 연산자

일부 함수는 특정 연산자를 인수로 사용할 수 있습니다.

예를 들어, `Previous` 함수는 `Self` 연산자를 사용할 수 있습니다.

모든 함수는 `)` 및 `(`을 사용하여 함수 인수를 묶습니다. 여러 개의 매개 변수를 허용하는 함수는 `;`을 사용하여 매개 변수를 구분합니다.

### 6.2.4.1 All 연산자

`All` 연산자는 `NoFilter` 함수가 모든 필터를 무시하도록 지시합니다.

`All` 연산자는 또한 `Count` 함수가 중복 요소를 포함한 모든 값을 세도록 지시할 수도 있습니다.

## 관련 정보

[Count \[페이지 47\]](#)

[Distinct/All 연산자 \[페이지 227\]](#)

[NoFilter \[페이지 210\]](#)

[All/Drill 연산자 \[페이지 225\]](#)

### 6.2.4.2 All/Drill 연산자

`All/Drill` 연산자는 `NoFilter` 함수와 함께 사용됩니다.

## 설명

All/Drill 연산자는 NoFilter 함수가 무시하는 필터를 확인합니다.

- 지정되지 않음 - NoFilter가 보고서 및 블록 필터 무시
- All - NoFilter가 모든 필터 무시
- Drill - NoFilter가 보고서 필터 및 드릴 필터 무시

### 6.2.4.3 Bottom/Top 연산자

Bottom/Top 연산자는 Rank 함수와 함께 사용됩니다.

## 설명

Bottom/Top 연산자가 내림차순 또는 오름차순으로 순위를 매기도록 Rank 함수에 지시합니다.

- Top - 내림차순으로 순위를 매깁니다.
- Bottom - 오름차순으로 순위를 매깁니다.

## 예

$\text{Rank}([수익]; ([국가]); \text{Top})$ 은 가장 높은 수익에서 낮은 수익 순으로 국가의 순위를 매깁니다.

## 관련 정보

[Rank \[페이지 170\]](#)

### 6.2.4.4 Break 연산자

Break 연산자는 Percentage 함수와 함께 사용됩니다.

## 설명

Break 연산자가 테이블 나누기를 처리하도록 Percentage 함수에 지시합니다.

## 예제

Percentage([수익]) 수식은 다음 표와 같이 값을 반환합니다. 백분율은 블록의 총 수익에 대해 계산됩니다.

연도	분기	수익	백분율
2005	1분기	10000	10%
2005	2분기	20000	20%
2006	1분기	30000	30%
2006	2분기	40000	40%

Percentage([수익];Break) 수식은 다음 표와 같이 값을 반환합니다. 백분율은 각 블록 부분의 총 수익에 대해 계산됩니다.

연도	분기	수익	백분율
2005	1분기	10000	33.3%
2005	2분기	20000	66.6%
2006	1분기	30000	42.9%
2006	2분기	40000	57.1%

## 관련 정보

[Percentage \[페이지 57\]](#)

### 6.2.4.5 Distinct/All 연산자

Distinct/All 연산자는 Count 함수와 함께 사용됩니다.

Distinct/All 연산자는 Count 함수에 고유 값만 또는 모든 값을 계산하도록 지시합니다.

## 예제

Count([수익];Distinct)는 [수익] 값이 (5;5;6;4)인 경우 3을 반환합니다.

Count([수익];All)은 [수익] 값이 (5;5;6;4)인 경우 4를 반환합니다.

## 관련 정보

[Count \[페이지 47\]](#)

### 6.2.4.6 IncludeEmpty 연산자

IncludeEmpty 연산자는 집계 함수와 함께 사용됩니다.

## 설명

IncludeEmpty 연산자는 일부 집계 함수(Average, Count, RunningAverage, RunningCount)가 빈 값을 계산에 포함하도록 지시합니다.

## 예제

Average([수익]; IncludeEmpty)는 [수익] 값이 (5;3;<공백>;4)인 경우 3을 반환합니다.

## 관련 정보

[Average \[페이지 46\]](#)

[Count \[페이지 47\]](#)

[RunningAverage \[페이지 60\]](#)

[RunningCount \[페이지 62\]](#)

### 6.2.4.7 Index 연산자

Index 연산자는 UserResponse 및 RefValueUserResponse 함수와 함께 사용됩니다.

## 설명

Index 연산자는 UserResponse 및 RefValueUserResponse 함수에 프롬프트 응답의 데이터베이스 기본 키를 반환하도록 지시합니다.

## 관련 정보

[UserResponse](#) [페이지 132]

[RefValueUserReponse](#) [페이지 129]

### 6.2.4.8 Linear 연산자

Linear 연산자는 `Interpolation` 함수와 함께 사용됩니다.

#### 설명

Linear 연산자는 `Interpolation` 함수에 최소 제곱 보간법을 이용한 선형 회귀 분석을 사용하여 누락된 계수 값을 제공하도록 지시합니다.

최소 제곱 보간법을 이용한 선형 회귀 분석을 사용하면 사용 가능한 모든 계수 값을 최대한 가깝게 통과하는  $f(x) = ax + b$  형식의 1차 방정식을 계산하여 누락된 값을 계산합니다.

## 관련 정보

[Interpolation](#) [페이지 50]

### 6.2.4.9 NoNull 연산자

NoNull 연산자는 `Previous` 함수와 함께 사용됩니다.

#### 설명

NoNull 연산자는 `Previous` 함수에 Null 값을 무시하도록 지시합니다.

NoNull과 함께 사용하는 경우 `Previous`는 현재 행 앞의 `offset` 행부터 시작하여 역방향으로 진행하면서 Null이 아닌 첫 번째 값을 반환합니다.

## 관련 정보

[Previous](#) [페이지 212]

---

### 6.2.4.10 NotOnBreak 연산자

NotOnBreak 연산자는 Interpolation 함수와 함께 사용됩니다.

#### 설명

NotOnBreak 연산자는 Interpolation 함수에 섹션 및 블록 나누기를 무시하도록 지시합니다.

#### 관련 정보

[Interpolation \[페이지 50\]](#)

### 6.2.4.11 PointToPoint 연산자

PointToPoint 연산자는 Interpolation 함수가 점 간 보간법을 사용하여 누락된 계수 값을 제공하도록 지시합니다.

#### 설명

점 간 보간법은 누락된 값의 두 인접 값을 통과하는  $f(x) = ax + b$  형식의 1차 방정식을 계산하여 누락된 값을 계산합니다.

#### 관련 정보

[Interpolation \[페이지 50\]](#)

### 6.2.4.12 Row/Col 연산자

Row 연산자는 행에 있는 각 값을 포함 컨텍스트에 있는 모든 행의 합계 값에 대한 백분율로 계산합니다. Col 연산자는 열에 있는 각 값을 포함 컨텍스트에 있는 모든 열의 합계 값에 대한 백분율로 계산합니다.

## 설명

Row/Col 연산자는 Percentage, Previous, RunningAverage, RunningCount, RunningMax, RunningMin, RunningProduct, RunningSum 함수의 계산 방향을 설정합니다.

## 참고

크로스탭에서 각 셀에 있는 값은 기본적으로 크로스탭에 있는 합계 값의 백분율로 계산됩니다. Row 연산자는 행에 있는 값을 행 합계 값에 대한 백분율로 계산합니다. Col 연산자는 열에 있는 값을 열 합계 값에 대한 백분율로 계산합니다.

## 예제

크로스탭에서 Percentage ([계수]) 는 다음과 같은 결과를 생성합니다.

계수	백분율	계수	백분율
100	10%	500	50%
200	20%	200	20%

Percentage ([계수]; Row) 는 다음과 같은 결과를 생성합니다.

계수	백분율	계수	백분율
100	16.7%	500	83.3%
200	50%	200	50%

Percentage ([계수]; Col) 는 다음과 같은 결과를 생성합니다.

계수	백분율	계수	백분율
100	33.3%	500	83.3%
200	66.6%	200	16.7%

Row 연산자는 행별 누적 집계를 계산합니다. Col 연산자는 열별 누적 집계를 계산합니다.

크로스탭에서 RunningSum ([계수]) 또는 RunningSum ([계수]; Row) 은 다음과 같은 결과를 생성합니다.

계수	RunningSum	계수	RunningSum
100	100	200	300
400	700	250	950

크로스탭에서 RunningSum ([계수]; Col) 은 다음과 같은 결과를 생성합니다.

계수	RunningSum	계수	RunningSum
100	100	200	700
400	500	250	950

## 관련 정보

[Percentage \[페이지 57\]](#)

[RunningAverage \[페이지 60\]](#)

[RunningCount \[페이지 62\]](#)

[RunningMax \[페이지 64\]](#)

[RunningMin \[페이지 66\]](#)

[RunningProduct \[페이지 68\]](#)

[RunningSum \[페이지 70\]](#)

## 6.2.4.13 Self 연산자

Self 연산자는 Previous 함수와 함께 사용됩니다.

## 설명

Previous 함수에 보고서 개체가 포함되지 않은 경우 이전 셀을 참조합니다.

## 예제

$5 + \text{Previous}(\text{Self})$  는 5, 10, 15, 20, 25, 30... 시퀀스를 반환합니다.

$1 + 0.5 * \text{Previous}(\text{Self})$  는 1, 1.5, 1.75, 1.88... 시퀀스를 반환합니다.

## 관련 정보

[Previous \[페이지 212\]](#)



## 6.2.4.14 Where 연산자

### 설명

Where 연산자는 계수를 계산하는 데 사용하는 데이터를 제한합니다.

### 예제

수식 `Average ([판매 수익]) Where ([국가] = "US")`는 국가 "US"의 평균 수익을 계산합니다.

수식 `Average ([판매 수익]) Where ([국가] = "US" Or [국가] = "France")`는 국가 "US" 또는 "France"의 평균 수익을 계산합니다.

수식 `[수익] Where (Not ([국가] Inlist ("US"; "France")))`는 US 및 France 이외의 국가에 대한 수익을 계산합니다.

변수 `[고수익]`은 수식 `[수익] Where [수익] > 500000`을 포함합니다. 블록에 있는 경우 `[고수익]`은 값이 500000보다 큰 경우 수익을 표시하고 이 값보다 작은 경우 아무 것도 표시하지 않습니다. 바닥글의 `[고수익]` 열 맨 아래에 있는 경우 수식 `Average ([고수익])`는 500000보다 큰 모든 수익의 평균을 반환합니다.

### 관련 정보

[And 연산자 \[페이지 221\]](#)

[Between 연산자 \[페이지 223\]](#)

[InList 연산자 \[페이지 224\]](#)

[Or 연산자 \[페이지 222\]](#)

[Not 연산자 \[페이지 223\]](#)

## 6.2.5 확장 구문 연산자

컨텍스트 연산자를 사용하여 입력 및 출력 컨텍스트를 명시적으로 지정합니다.

다음 표에는 컨텍스트 연산자가 나열되어 있습니다.

연산자	설명
In	컨텍스트에서 사용할 명시적인 차원 목록을 지정합니다.
ForEach	기본 컨텍스트에 차원을 추가합니다.
ForAll	기본 컨텍스트에서 차원을 제거합니다.

ForAll 및 ForEach 연산자는 차원이 여러 개인 기본 컨텍스트가 있을 때 유용합니다. 대개는 In을 사용하여 명시적으로 목록을 지정하는 것보다 ForAll과 ForEach를 사용하여 컨텍스트를 "추가"하거나 "제거"하는 것이 더 쉽습니다.

## 6.2.5.1 In 컨텍스트 연산자

In 컨텍스트 연산자는 컨텍스트에 차원을 명시적으로 지정합니다.



예

### In을 사용하여 컨텍스트에 차원 지정

이 예제의 보고서는 연도와 판매 수익을 보여 줍니다. 데이터 공급자에는 분기 개체도 들어 있지만 블록에는 이 차원이 포함되어 있지 않습니다. 대신 각 연도의 분기별 최고 수익을 나타내기 위한 추가 열을 포함하려고 합니다. 이 보고서는 다음과 같습니다.

연도	매출	최대 분기별 수익
2001	\$8,096,123.60	\$2,660,699.50
2002	\$13,232,246.00	\$4,186,120.00
2003	\$15,059,142.80	\$4,006,717.50

분기별 최고 수익 열의 값은 이 블록을 분기 차원이 포함된 블록과 함께 검사하여 얻은 값임을 알 수 있습니다.

연도	분기	매출
2001	Q1	\$2,660,699.50
2001	Q2	\$2,279,003.00
2001	Q3	\$1,367,841.00
2001	Q4	\$1,788,580.00
	최대값:	\$2,660,699.50

연도	분기	매출
	Q1	\$3,326,172.00
	Q2	\$2,840,651.00
	Q3	\$2,879,303.00
	Q4	\$4,186,120.00
	최대값:	\$4,186,120.00

연도	분기	매출
	Q1	\$3,742,989.00
	Q2	\$4,006,717.50
	Q3	\$3,953,395.00
	Q4	\$3,356,041.00
	최대값:	\$4,006,717.50

분기별 최고 수익 열에는 각 연도의 분기별로 가장 높은 수익이 표시됩니다. 예를 들어, 2002년에는 4분기의 수익이 가장 높았으므로 분기별 최고 수익에는 2002년을 나타내는 행의 4분기 수익이 표시됩니다.

In 연산자를 사용할 경우 분기별 최고 수익의 수식은 다음과 같습니다.

```
Max ([판매 수익] In ([연도];[분기])) In ([연도])
```

이 수식은 각 (연도, 분기) 조합의 최고 판매 수익을 계산한 다음 이 값을 연도별로 출력합니다.

#### 노트

블록의 기본 출력 컨텍스트는 연도이므로 이 수식에 출력 컨텍스트를 명시적으로 지정할 필요는 없습니다.

## 6.2.5.2 ForEach 컨텍스트 연산자

ForEach 연산자는 컨텍스트에 차원을 추가합니다.

### 예

#### ForEach 연산자를 사용하여 컨텍스트에 차원 추가

다음 표에서는 분기 차원이 포함되어 있지만 분기 차원이 블록에 포함되지 않은 보고서의 각 분기에 대한 최고 수익을 보여 줍니다.

연도	매출	최대 분기별 수익
2001	8096123.60	2660699.50
2002	13232246.00	4186120.00
2003	15059142.80	4006717.50

ForEach 연산자가 포함되지 않은 분기별 최고 수익 열에 대한 수식을 만들 수 있습니다.

```
Max ([판매 수익] In ([연도];[분기])) In ([연도])
```

ForEach 컨텍스트 연산자를 사용하여 다음 수식으로 동일한 결과를 얻을 수 있습니다.

```
Max ([판매 수익] ForEach ([분기])) In ([연도])
```

이유 연도 차원이 블록의 기본 입력 컨텍스트이기 때문입니다. ForEach 연산자를 사용하여 컨텍스트에 분기 차원을 추가하고 입력 컨텍스트를 ([연도];[분기])로 지정합니다.

## 6.2.5.3 ForAll 컨텍스트 연산자

ForAll 컨텍스트 연산자는 컨텍스트에서 차원을 제거합니다.



예

### ForAll을 사용하여 컨텍스트에서 차원 제거

연도, 분기 및 판매 수익이 표시된 보고서에 다음 블록에 표시된 것과 같이 각 연도의 총 수익을 표시하는 열을 추가하려고 합니다.

Year	Quarter	Sales revenue	Yearly Total
2001	Q1	\$2660700	\$8096124
2001	Q2	\$2279003	\$8096124
2001	Q3	\$1367841	\$8096124
2001	Q4	\$1788580	\$8096124
2002	Q1	\$3326172	\$13232246
2002	Q2	\$2840651	\$13232246
2002	Q3	\$2879303	\$13232246
2002	Q4	\$4186120	\$13232246
2003	Q1	\$3742989	\$15059143
2003	Q2	\$4006718	\$15059143
2003	Q3	\$3953396	\$15059143
2003	Q4	\$3356041	\$15059143

연도별로 총 수익을 구하려면 입력 컨텍스트가 (연도)여야 하는데 기본 입력 컨텍스트는 (연도; 분기)입니다. 따라서 다음과 같이 수식에 ForAll ([분기])를 지정하여 입력 컨텍스트에서 분기를 제거할 수 있습니다.

```
Sum([판매 수익] ForAll ([분기]))
```

In 연산자를 사용하여 분기를 제거할 수도 있습니다. 이 경우 수식은 다음과 같습니다.

```
Sum([판매 수익] In ([연도]))
```

이 버전의 수식에서는 분기를 제거하고 연도만 남겨 두는 것이 아니라 명시적으로 연도를 컨텍스트로 지정합니다.

## 6.2.6 집합 연산자

집합 연산자는 계층형 데이터의 멤버에 대해 작동합니다.

## 6.2.6.1 범위 연산자

### 설명

범위 연산자(:)는 같은 수준의 두 멤버를 포함하여 이 둘 사이에 있는 멤버 집합을 반환합니다.

### 구문

```
first_member:last_member
```

### 예

[지역]&[미국].[캘리포니아].[로스앤젤레스]:[지역]&[미국].[캘리포니아].[샌프란시스코] 는 같은 수준의 멤버가 ...[로스앤젤레스],[샌디에이고],[샌프란시스코]... 순서로 되어 있는 경우 [로스앤젤레스],[샌디에이고],[샌프란시스코]를 반환합니다.

Sum([수익];{[지역]&[미국].[캘리포니아].[로스앤젤레스]:[지역]&[미국].[캘리포니아].[샌프란시스코]})  
는 로스앤젤레스, 샌디에이고 및 샌프란시스코의 총 수익을 반환합니다.

## 6.3 확장 구문 키워드

확장 구문 키워드는 확장 구문에 차원을 명시적으로 지정하지 않고도 해당 차원을 참조할 수 있도록 하는 유용한 방법입니다.

이러한 키워드는 경쟁력 있는 보고서를 만드는 데 도움이 됩니다. 따라서 수식에 하드 코딩된 차원 참조가 포함되어 있지 않을 경우 보고서에서 차원이 추가되거나 제거되더라도 계속 작동합니다.

확장 구문 키워드에는 Report, Section, Break, Block 및 Body가 있습니다.

### 6.3.1 Block 키워드

이 항목은 보고서에 Block 키워드가 사용된 위치에 따라 이 키워드가 참조하는 차원에 대해 설명합니다. Block 키워드는 대개 Section 키워드와 동일한 데이터를 나타냅니다.

그러나 Block 키워드는 블록에 대한 필터를 고려하는 반면 Section 키워드는 이를 무시합니다.

사용 위치	참조하는 데이터
블록	전체 블록의 데이터(나누기 무시, 필터 고려)
블록 나누기(머리글 또는 바닥글)	전체 블록의 데이터(나누기 무시, 필터 고려)
섹션(머리글, 바닥글, 또는 블록 외부)	해당 없음
모든 블록 또는 섹션 외부	해당 없음



## Block 키워드

보고서에 연도, 분기 및 판매 수익이 나타나 있다고 가정합니다. 이 보고서에는 연도 기준 섹션이 있으며 블록은 3분기와 4분기를 제외하도록 필터링되어 있습니다.

2001			
Quarter	Sales revenue	First Half Average	Yearly Average
Q1	\$2,660,700	\$2,469,851.25	\$8,096,123.60
Q2	\$2,279,003	\$2,469,851.25	\$8,096,123.60
<b>합계:</b>	<b>4,939,702.5</b>		
2002			
Quarter	Sales revenue	First Half Average	Yearly Average
Q1	\$3,326,172	\$3,083,411.50	\$13,232,246.00
Q2	\$2,840,651	\$3,083,411.50	\$13,232,246.00
<b>합계:</b>	<b>6,166,823.0</b>		
2003			
Quarter	Sales revenue	First Half Average	Yearly Average
Q1	\$3,742,989	\$3,874,853.20	\$15,059,142.80
Q2	\$4,006,718	\$3,874,853.20	\$15,059,142.80
<b>합계:</b>	<b>7,749,706.4</b>		

연간 평균 열에는 다음과 같은 수식이 사용됩니다.

Average([판매 수익] In Section)

상반기 평균 열에는 다음과 같은 수식이 사용됩니다.

Average([판매 수익]) In Block

다음 그림을 보면 Block 키워드가 블록의 필터를 어떻게 고려하는지 알 수 있습니다.

## 6.3.2 Body 키워드

이 항목은 보고서에 Block 본문의 키워드가 사용된 위치에 따라 이 키워드가 참조하는 차원에 대해 설명합니다.

사용 위치	참조하는 데이터
블록	블록의 데이터
블록 나누기(머리글 또는 바닥글)	블록의 데이터
섹션(머리글, 바닥글, 또는 블록 외부)	섹션의 데이터
모든 블록 또는 섹션 외부	보고서의 데이터



### Body 키워드

연도, 분기 및 판매 수익을 표시하고 연도에 나누기가 포함된 보고서가 있다고 가정합니다. 이 보고서에는 연도 기준의 섹션과 분기 기준으로 적용된 나누기가 있습니다.

연도	분기	매출	본문
2001	Q1	2,660,700	2,660,699.5
	Q2	2,279,003	2,279,003
	Q3	1,367,841	1,367,840.7
	Q4	1,788,580	1,788,580.4
2001		8,096,123.6	

본문 열에는 다음과 같은 수식이 있습니다.

Sum ([판매 수익]) In Body

Body 키워드는 블록의 데이터를 참조하므로 실제 값 열의 합계는 판매 수익 열의 합계와 같습니다. 월 개체를 제거하면 블록 열의 값은 판매 수익 열의 변경된 값에 따라 변경됩니다. 보고서 바닥글에 수식을 삽입한 경우에는 본문에 대한 총 수익이 반환됩니다.

## 6.3.3 Break 키워드

다음 표에서는 보고서에 Break 키워드가 사용된 위치에 따라 이 키워드가 참조하는 차원에 대해 설명합니다.

사용 위치	참조하는 데이터
블록	나누기로 구분된 블록 요소의 데이터

사용 위치	참조하는 데이터
블록 나누기(머리글 또는 바닥글)	나누기로 구분된 블록 요소의 데이터
섹션(머리글, 바닥글, 또는 블록 외부)	해당 없음
모든 블록 또는 섹션 외부	해당 없음



### 예

**Break 키워드**

연도, 분기 및 판매 수익이 표시되는 보고서가 있습니다.

연도	분기	매출	나누기 합계
2001	Q1	\$2,660,700	\$8,096,124
	Q2	\$2,279,003	\$8,096,124
	Q3	\$1,367,841	\$8,096,124
	Q4	\$1,788,580	\$8,096,124

이 보고서에는 연도에 대한 나누기가 포함되어 있습니다. 나누기 합계 열에는 다음과 같은 수식이 있습니다.

Sum ([판매 수익]) In Break

이 열의 기본 출력 컨텍스트는 ([연도]:[분기])이므로 Break 키워드를 사용하지 않을 경우 이 열의 값은 판매 수익 열과 중복되게 됩니다.

## 6.3.4 Report 키워드

이 항목은 보고서에 Report 키워드가 사용된 위치에 따라 이 키워드가 참조하는 데이터에 대해 설명합니다.

사용 위치	참조하는 데이터
블록	보고서의 모든 데이터
블록 나누기(머리글 또는 바닥글)	보고서의 모든 데이터
섹션(머리글, 바닥글, 또는 블록 외부)	보고서의 모든 데이터
모든 블록 또는 섹션 외부	보고서의 모든 데이터



### 예

### Report 키워드

보고서에 연도, 분기 및 판매 수익이 나타나 있다고 가정합니다. 이 보고서에는 보고서의 모든 수익 합계를 표시하는 보고서 합계 열도 있습니다.



Year	Quarter	Sales revenue	Report Total
2001	Q1	\$2,660,700	36,387,512.4
2001	Q2	\$2,279,003	36,387,512.4
2001	Q3	\$1,367,841	36,387,512.4
2001	Q4	\$1,788,580	36,387,512.4
2002	Q1	\$3,326,172	36,387,512.4
2002	Q2	\$2,840,651	36,387,512.4
2002	Q3	\$2,879,303	36,387,512.4
2002	Q4	\$4,186,120	36,387,512.4
2003	Q1	\$3,742,989	36,387,512.4
2003	Q2	\$4,006,718	36,387,512.4
2003	Q3	\$3,953,395	36,387,512.4
2003	Q4	\$3,356,041	36,387,512.4

보고서 합계 열에 대한 수식은 다음과 같습니다.

Sum([판매 수익]) In Report

이 열의 기본 출력 컨텍스트는 ([연도]:[분기])이므로 Report 키워드를 사용하지 않을 경우 이 열의 값은 판매 수익 열과 중복되게 됩니다.

### 6.3.5 Section 키워드

이 항목은 보고서에 Section 키워드가 사용된 위치에 따라 이 키워드가 참조하는 데이터에 대해 설명합니다.

사용 위치	참조하는 데이터
블록	섹션의 모든 데이터
블록 나누기(머리글 또는 바닥글)	섹션의 모든 데이터
섹션(머리글, 바닥글, 또는 블록 외부)	섹션의 모든 데이터
모든 블록 또는 섹션 외부	해당 없음



#### Section 키워드

보고서에 연도, 분기 및 판매 수익이 나타나 있다고 가정합니다.

2001		
Quarter	Sales revenue	Section Total
Q1	\$2,660,700	8,096,124
Q2	\$2,279,003	8,096,124
Q3	\$1,367,841	8,096,124
Q4	\$1,788,580	8,096,124

이 보고서에는 연도 기준 섹션이 있으며 섹션 합계 열에는 다음과 같은 수식이 있습니다.

Sum ([판매 수익]) In Section

연도 개체에 대해 섹션 나누기를 적용했으므로 섹션 합계 열의 값은 2001년의 총 수익을 나타냅니다. 이 열의 기본 출력 컨텍스트는 ([연도];[분기])이므로 Section 키워드를 사용하지 않을 경우 이 열의 값은 판매 수익 열과 중복되게 됩니다.

## 6.4 값 반올림 및 자르기

몇몇 함수에는 반환 값을 반올림하고 자르는 수준을 결정하는 매개 변수가 포함되어 있습니다.

이 매개 변수는 0보다 큰 정수, 0, 그리고 0보다 작은 정수 중 하나를 입력 받습니다. 다음 표는 숫자가 어떻게 반올림되고 잘리는지를 보여 줍니다.

매개 변수	설명
> 0	<p>&lt;parameter&gt; 소수 자릿수로 반올림하고 자릅니다.</p> <p>예:</p> <p>Round(3.13;1)는 3.1을 반환합니다.</p> <p>Round(3.157;2)는 3.16을 반환합니다.</p>
0	<p>가장 가까운 정수로 반올림하고 자릅니다.</p> <p>예:</p> <p>Truncate(3.7;0)는 3을 반환합니다.</p> <p>Truncate(4.164;0)는 4를 반환합니다.</p>
< 0	<p>가장 가까운 10(매개 변수 = -1), 100(매개 변수 = -2), 1000(매개 변수 = -3) 자릿수로 반올림하고 자릅니다.</p> <p>예:</p> <p>Round(123.76;-1)는 120을 반환합니다.</p>

매개 변수	설명
	Round(459.9;-2)는 500을 반환합니다. Truncate(1600;-3)는 1000을 반환합니다.

### 노트

숫자는 내부적으로 배정밀도 부동 소수점 수 형식으로 표현되며 15~17자리의 정밀도를 가집니다.

## 관련 정보

[Round \[페이지 172\]](#)

[Truncate \[페이지 177\]](#)

[EuroConvertTo \[페이지 158\]](#)

[EuroConvertFrom \[페이지 157\]](#)

[EuroFromRoundError \[페이지 160\]](#)

[EuroToRoundError \[페이지 162\]](#)

## 6.5 계층구조에서 멤버 및 멤버 집합 참조

다음 구문을 사용하여 함수에서 멤버 및 멤버 집합을 참조합니다. `[hierarchy]&path.function`

`path` 및 `function` 부분은 옵션입니다. `path`의 경우, 마침표로 구분되는 꺾쇠 괄호 속의 각 멤버를 참조합니다. 멤버 및 수준의 이름은 대/소문자가 구분됩니다.

### 노트

멤버 집합을 사용하여 계층구조에 대한 기본 계산 컨텍스트를 무시할 수 있습니다. 멤버 집합을 허용하는 함수의 경우 멤버 집합을 {}로 묶습니다.

시작 멤버와 끝 멤버 사이에 콜론(:)을 사용하고 각 멤버에 대해 지정된 전체 경로를 사용하여 멤버 범위를 나타냅니다. 범위는 지정된 멤버와 수준이 동일한 멤버를 모두 포함합니다.

범위 구문 예: `[판매 계층구조]&[고객 유형].[기업];[대기업].[Nancy Davolio]: [판매 계층구조]&[고객 유형].[기업];[대기업].[Andrew Smith]`

### 예

#### 멤버 및 멤버 집합 참조

다음과 같은 계층구조가 있습니다.

판매 계층구조	주문 금액
고객 유형	277,290,434
기업	180,063,361
대기업	113,905,997
Nancy Davolio	44,855,689
Janet Leverling	44,050,308
Andrew Smith	30,000,000
세계	91,157,363

- `[판매 계층구조]&[고객 유형].[기업].[대기업].Children`은 `[Nancy Davolio]`, `[Janet Leverling]` 및 `[Andrew Smith]` 멤버를 참조합니다.
- `Sum([주문 금액];{[판매 계층구조]&[고객 유형].[기업].[대기업].children})`은 113,905,997(세 하위 멤버에 해당하는 계수의 합)을 반환합니다.
- `[판매 계층구조]&[고객 유형].[기업].[대기업].[Janet Leverling]`은 `[Janet Leverling]` 멤버를 참조합니다.
- `Sum([주문 금액];{[판매 계층구조]&[고객 유형].[기업].[대기업].[Janet Leverling];[판매 계층구조]&[고객 유형].[기업].[대기업].[Nancy Davolio]})`는 88,905,997(두 멤버에 해당하는 계수의 합)을 반환합니다.
- `[판매 계층구조]&[고객 유형].[기업].[대기업].[Nancy Davolio]:[판매 계층구조]&[고객 유형].[기업].[대기업].[Andrew Smith]`는 `[Nancy Davolio]`, `[Janet Leverling]` 및 `[Andrew Smith]` 멤버를 참조합니다.
- `Sum([주문 금액];{[판매 계층구조]&[고객 유형].[기업].[대기업].[Nancy Davolio]:[판매 계층구조]&[고객 유형].[기업].[대기업].[Andrew Smith]})`는 113,905,997(범위 내에 속한 세 멤버에 해당하는 계수의 합)을 반환합니다.
- `[판매 계층구조].children`은 `[판매 계층구조]` 계층구조의 모든 멤버를 참조합니다.
- `Sum([주문 금액];{[판매 계층구조].children})`은 277,290,434를 반환합니다.

## 7 수식 문제 해결

### 7.1 자동 수식 재작성 메커니즘

Web Intelligence의 후속 교정 유지보수 릴리스가 등장하면서, 버전에 따라 계산 결과가 달라지는 경우가 발생할 수 있습니다.

버전 4.1 SP3 이후, Web Intelligence는 이전 버전에서 마이그레이션된 문서에서 수식(아래 목록 참조) 선택을 자동으로 수정해 주는 자동 수식 재작성 메커니즘을 제공합니다. 이러한 수식은 특정 패턴을 따릅니다. 수정 후에는 수식이 계산 변경 전과 동일한 결과를 반환합니다. 따라서 수정사항이 문서에 유지되도록 문서를 저장하는 것이 좋으며, 이렇게 해서 수식 재작성 메커니즘이 완료됩니다.

자동 수식 재작성(Automatic Formula Rewrite) 메커니즘은 BI 4.1 SP3 이상으로 마이그레이션된 문서에 기본적으로 사용될 수 있으며, 다음과 같은 수식 패턴에 대해 적용됩니다.

1. 조건 내에서 차원을 매개 변수로 갖는 Where() 연산자,
2. 섹션 안에서 재설정되는 누적 계산
3. 크로스 테이블 안에서 재설정되는 누적 계산

이 규칙 목록은 이후 릴리스에서 더 많은 수식 패턴을 포함하도록 확장될 수도 있습니다.

#### 규칙(1)

이전 버전에서는 조건에서 매개 변수로 차원이 있는 Where() 연산자가 있는 경우 데이터가 특정 방식으로 계산되었습니다. 실제로, 차원이 계수 컨텍스트에 추가되었습니다. 규칙(1)은 이전의 동작을 재현합니다.

이 규칙은 XI 3.1 FP3.6, XI 3.1 FP4.1, XI 3.1 FP5.1, 4.0 SP5에서 마이그레이션된 모든 문서에 적용됩니다.

#### 규칙(2)

이전 버전에서는 계산이 각 섹션 인스턴스에서 재설정되었으므로 섹션의 누적 계산이 제대로 실행되지 않았습니다. 규칙(2)은 이전의 동작을 재현합니다.

이 규칙은 XI R2 SP4에서 마이그레이션된 모든 문서에 적용됩니다.

#### 규칙(3)

이전 버전에서, 크로스 테이블이 재설정된 누적 계산은 계산이 “Z” 패턴(행을 차례대로 실행)이 아닌 “N” 패턴(열을 차례대로 실행)으로 실행되었음을 의미합니다.

규칙(3)에서는 Web Intelligence가 “N” 패턴으로 계산을 실행하도록 강제 지정하는 FORCE\_COL 키워드가 도입되었습니다.

예를 들어, 규칙(3)에서 RunningSum([Sales revenue];([State]))이 RunningSum([Sales revenue];FORCE\_COL;([State]))으로 수정되면 열을 차례대로 실행하도록 강제 지정될 수 있습니다.

이 규칙은 XI 3.x, 4.0 패치 2.20, 4.0 SP5, 4.0 SP6, 4.0 SP7, 4.1, 4.1 SP1의 모든 버전에서 마이그레이션된 모든 문서에 적용됩니다.

## 7.2 수식 오류 및 정보 메시지

조건부 서식을 사용하여 오류 메시지를 반환하는 보고서 데이터의 서식을 지정할 수 있습니다.

일부 경우에 수식이 값을 반환할 수 없으며 “#”으로 시작하는 오류 또는 정보 메시지를 반환합니다. 이 메시지는 수식이 배치된 셀에 나타납니다.

### 7.2.1 #COMPUTATION

RelativeValue 함수에서 지정한 분리 차원을 함수가 위치한 블록의 계산 컨텍스트에서 더 이상 사용할 수 없게 되면 #COMPUTATION 오류가 발생합니다.

#COMPUTATION은 또한 계층구조를 포함하는 병합된 개체가 보고서에 포함된 경우에도 발생합니다.

#COMPUTATION 오류는 수식에서 컨텍스트 연산자를 잘못 사용한 것과도 관련이 있습니다.

#### 관련 정보

[RelativeValue \[페이지 217\]](#)

### 7.2.2 #CONTEXT

존재하지 않는 계산 컨텍스트가 계수에 포함된 경우 #CONTEXT 오류가 계수에 표시됩니다.

#CONTEXT 오류는 #INCOMPATIBLE 및 #DATASYNC 오류 메시지와 관련되어 있습니다. 두 오류 모두 존재하지 않는 계산 컨텍스트가 블록에 포함된 경우 차원에 표시됩니다.

#INCOMPATIBLE의 경우 차원이 호환되지 않기 때문에 컨텍스트가 존재하지 않으며, #DATASYNC의 경우 차원이 동기화되지 않은 여러 데이터 공급자에서 제공된 것이기 때문에 컨텍스트가 존재하지 않습니다.



#### 예

**쿼리에서 존재하지 않는 계산 컨텍스트**  
Island Resorts Marketing 유니버스 기반의 블록에 예약 연도 및 수익 개체가 포함되어 있으면 예약 연도별로 수익을 집계할 수 없기 때문에 #CONTEXT 오류 메시지가 나타납니다. (예약에서 아직 어떠한 수익도 생성하지 않은 경우)

## 7.2.3 #DATASYNC

#DATASYNC는 다른 데이터 공급자의 차원이 포함된 블록에 또 다른 데이터 공급자의 차원을 넣을 때 두 데이터 공급자가 병합된 차원을 통해 동기화되지 않을 경우 발생합니다.

#DATASYNC는 블록의 모든 차원에 표시되고 계수에는 #CONTEXT가 표시됩니다.



#### 예

**블록의 여러 데이터 공급자 차원**  
Island Resorts Marketing 유니버스 기반의 보고서에 (연도, 수익) 개체와 (분기) 개체가 있는 데이터 공급자가 포함되어 있으면 두 데이터 공급자가 병합된 차원을 통해 동기화되지 않으므로 연도, 분기 및 수익이 포함된 블록의 연도 및 분기 옆에 #DATASYNC가 표시됩니다.

## 7.2.4 #DIV/0

#DIV/0은 수식에서 산술적으로 불가능한 계산인 0으로 나누기를 시도할 때 발생합니다.

0은 제수로 나타낼 수 없습니다.



#### 예

**품목당 수익 결정**  
판매 수익, 판매된 품목 수 및 품목당 수익(판매 수익을 판매된 품목 수로 나누어 계산)을 보여 주는 보고서가 있습니다.

그런데 전혀 수익이 발생하지 않은 분기가 있었습니다. 이 경우 수식에서 0으로 나누기(즉, 판매된 품목 수 0으로 수익 나누기)를 시도하므로 이 분기에 대한 품목당 수익 옆에는 #DIV/0이 반환됩니다.

## 7.2.5 #ERROR

#ERROR는 다른 오류 메시지에서 다루지 않는 모든 오류를 다루는 기본 오류 메시지입니다.

## 7.2.6 #EXTERNAL

#EXTERNAL은 수식이 Web Intelligence에서 사용할 수 없는 외부 함수를 참조하는 경우에 발생합니다.

## 7.2.7 #INCOMPATIBLE

#INCOMPATIBLE은 블록에 호환되지 않는 개체가 포함되어 있는 경우 발생합니다.



예

### 쿼리에서 호환되지 않는 개체

Island Resorts Marketing 유니버스 기반의 블록에 Year 및 Reservation Year 차원이 포함되어 있으면 이러한 개체가 호환되지 않기 때문에 해당 차원을 포함하는 열에 #INCOMPATIBLE이 표시됩니다.

## 7.2.8 #MIX

#MIX는 집계 계수의 단위가 서로 다른 경우에 발생합니다.

예를 들어, 서로 다른 통화로 지정된 통화 값을 집계하는 경우 셀에 #MIX가 표시됩니다.

## 7.2.9 #MULTIVALUE

#MULTIVALUE는 값을 하나만 출력하는 셀에 둘 이상의 값을 반환하는 수식이 있을 경우 발생합니다.



예

### 셀의 다중 값

국가, 휴양지 및 수익을 보여 주는 보고서가 있고, 수식 [수익] ForEach ([국가])가 포함된 보고서를 셀에 추가합니다. 보고서에 '미국'과 '프랑스'라는 두 개의 국가 값이 들어 있으므로 이 셀은 #MULTIVALUE를 반환합니다.

하나의 셀이 미국과 프랑스에 대한 수익을 모두 표시할 수 없습니다. 테이블 외부에서 수익을 포함하는 셀은 단지 일정한 방식(예: 합계 또는 평균 계산)으로 테이블의 수익을 집계합니다.

보고서를 국가에 대해 섹션으로 나눌 경우 섹션당 하나의 국가 값만 존재하기 때문에 섹션에 있는 수식은 올바릅니다. 하지만 섹션 외부에서는 여전히 수식이 #MULTIVALUE를 반환합니다.

## 7.2.10 #N/A

보고서의 셀 값이 기본 데이터베이스에서 사용할 수 없는 보고서의 값을 기반으로 하는 경우(예: BEx 셀의 BW 오류) 셀에 #N/A(해당 없음)이 표시되는데, 이는 데이터를 검색할 수 없어 셀이 비어 있음을 의미합니다.



## 7.2.11 #OVERFLOW

#OVERFLOW는 계산식이 소프트웨어에서 처리하기에 너무 큰 값을 반환할 경우에 발생합니다.

이 값은 지수 형식으로 1.7E308(1.7 다음에 307개의 0이 붙음)입니다.

## 7.2.12 #PARTIALRESULT

#PARTIALRESULT는 보고서 개체와 연결된 행을 모두 가져올 수 없는 경우에 발생합니다.

#PARTIALRESULT가 보고서에서 자주 발생하고 사용자에게 적합한 보안 권한이 있는 경우, 가져올 수 있는 최대 행 수 쿼리 속성을 수정하여 더 많은 데이터를 가져올 수 있도록 설정하십시오. 쿼리 수정 권한이 없는 경우 BI 관리자에게 문의하십시오.

보고서에 스마트 계수가 있는 경우에는 스마트 계수는 기본 계수보다 더 많은 양의 데이터를 가져오도록 요구하므로 #PARTIALRESULT가 더 많이 발생할 수 있습니다.

## 7.2.13 #RANK

#RANK는 값의 순서에 종속되는 개체를 기준으로 데이터의 순위를 지정할 경우 발생합니다.

Previous 함수나 누적 집계 함수를 사용하는 개체는 값의 순서에 의존합니다.

순위를 지정하면 이러한 개체가 값을 다시 계산하므로 순위가 변경되고 순환 종속성이 생깁니다. 이러한 종속성은 순위 대화 상자에서 순위를 지정하거나 Rank 함수를 사용하는 경우에 발생할 수 있습니다.



예

### 누적 평균 또는 이전 값에 대한 순위 지정

Previous 함수나 누적 집계 함수가 포함된 열에서 블록의 순위를 지정하려고 하면 전체 블록에서 #RANK를 반환합니다.

## 7.2.14 #RECURSIVE

#RECURSIVE는 순환 종속성 때문에 계산을 수행할 수 없는 경우에 발생합니다.



예

### NumberOfPages() 함수 사용

NumberOfPages 함수를 자동 맞춤 셀에 배치하면 순환 종속성이 생기므로 이 함수를 높이 자동 맞춤 또는 너비 자동 맞춤 속성이 설정된 셀에 배치할 경우 셀에서 #RECURSIVE가 반환됩니다. 함수가 값을 반환하기 위해서는 보고서의 정확한 크기를 알아야 하지만, 보고서 크기에 영향을 주는 셀 크기는 셀 내용에 의해 결정됩니다.

## 7.2.15 #REFRESH

#REFRESH는 쿼리에서 제거된 다음 쿼리에 다시 추가된 개체에서 파생된 값을 갖는 보고서 셀에 나타납니다.

**쿼리 제거 사용** 쿼리 속성이 선택되어 있을 경우 쿼리 기반의 보고서에 기여하지 않은 개체는 쿼리에서 제거됩니다.

그런 다음 쿼리를 새로 고치면 셀이 개체에서 파생된 값으로 다시 채워집니다.

## 7.2.16 #SECURITY

#SECURITY는 보안 권한이 없는 함수를 사용할 경우 발생합니다.



예

### DataProviderSQL() 함수 사용

데이터 공급자 SQL을 볼 수 있는 권한이 없는 사용자가 셀에 DataProviderSQL() 함수를 넣으면 #SECURITY 메시지가 셀에 표시됩니다.

## 7.2.17 #SYNTAX

#SYNTAX는 보고서에 더 이상 존재하지 않는 개체를 수식에서 참조할 경우 발생합니다.



예

### 존재하지 않는 개체 참조

원래 연도, 분기 및 판매 수익을 보여 주었던 보고서가 있고, 여기에 수익과 연간 평균 수익의 차이를 보여 주는 열이 추가되었습니다. 이 수치는 연간 평균 차이 변수로 제공됩니다.

보고서에서 연간 평균 차이 변수를 삭제하면 이 변수가 포함된 열에서 #SYNTAX를 반환합니다.

## 7.2.18 #TOREFRESH

#TOREFRESH는 스마트 계수로 반환된 값을 사용할 수 없는 경우 스마트 계수를 기반으로 하는 셀에 나타납니다.

이러한 상황은 데이터 공급자에서 해당 값이 포함된 그룹화 집합을 사용할 수 없는 경우 발생합니다.

데이터를 새로 고치면 #TOREFRESH 오류가 제거됩니다.

계수 중 일부가 "위임"되었습니다(BW에서는 SUM으로 집계되지 않는 계수를 위임된 계수라 함). 즉, 어떤 계수에 테이블 또는 계산을 정의하면 이 계수가 특정 집계 상황에서 쿼리됩니다(특정 차원 집합에 해당 계수가 주어짐). 이 차원 집합이 쿼리 차원 집합의 하위 집합인 경우, 해당 계수는 주어진 차원 집합(또는 SQL에서 group by 절을 참조하는 그룹화 집합)에 따라 집계되어야 합니다.

일반 계수의 경우 시스템에서 집계를 수행하지만, 위임된 계수의 경우에는 집계가 관련 데이터베이스로 위임됩니다. 이를 위해 시스템에서는 이 데이터베이스를 다시 쿼리해야 합니다. 이 작업은 자동으로 수행되지 않으므로 #TOREFRESH

---

가 표시되며 사용자가 새로 고침을 수행할 때까지 기다립니다. 사용자가 새로 고치면 요청된 집계를 가져온 다음 #TOREFRESH를 해당 값으로 바꾸는 쿼리가 추가로 실행됩니다.

## 7.2.19 #UNAVAILABLE

#UNAVAILABLE은 스마트 계수의 값을 계산할 수 없는 경우에 나타납니다.

이러한 상황은 필터를 쿼리에 적용하지 않으면 필터링된 스마트 계수의 값을 표시할 수 없는 경우에 발생합니다. 이 경우 같은 쿼리를 기반으로 하는 다른 보고서에 영향을 미칠 수 있기 때문에 필터가 적용되지 않습니다.

## 8 함수를 사용하여 값 비교

### 8.1 Previous 함수를 사용하여 값 비교

Previous 함수는 식의 이전 비교 값을 반환합니다.

반환되는 값은 보고서의 레이아웃에 따라 다릅니다.

보다 강력한 비교를 수행하려면 RelativeValue 함수를 사용합니다. RelativeValue 함수는 식의 이전 또는 이후 비교 값을 반환합니다. 반환되는 값은 보고서의 레이아웃에 종속되지 않습니다.

#### 관련 정보

[Previous \[페이지 212\]](#)

[RelativeValue \[페이지 217\]](#)

[RelativeValue 함수를 사용하여 값 비교 \[페이지 252\]](#)

### 8.2 RelativeValue 함수를 사용하여 값 비교

RelativeValue 함수는 식의 비교 값을 반환합니다. 함수는 보고서의 레이아웃과는 관계 없이 이러한 값을 반환합니다.

RelativeValue 함수를 사용할 경우 다음을 지정합니다.

- 찾으려는 비교 값의 식(해당 식은 블록에서 사용 가능한 계수 또는 세부 차원이어야 함)
- 분리 차원 목록
- 오프셋

함수는 분리 차원, 오프셋 및 하위 축 차원(분리 차원에 포함)을 사용하여 비교 값을 반환합니다. 하위 축 차원은 계산 컨텍스트에서 분리 차원 이외의 모든 차원입니다.

일반적으로 RelativeValue는 현재 행에서 제거된 offset 행이고 하위 축 차원 값이 현재 행에서 동일한 식의 값을 분리 차원 값 목록 행에 반환합니다.

#### i 노트

모든 분리 차원은 함수가 있는 블록의 계산 컨텍스트에 항상 있어야 합니다. 이후에 분리 차원이 제거되면 함수는 #COMPUTATION을 반환합니다.



이 예제에서 RelativeValue 열은 다음 수식을 포함합니다.

```
RelativeValue([수익];([연도]);-1)
```

- [수익]은 식을 나타내고,
- [연도]는 분리 차원을 나타내며,
- -1(함수가 목록에서 이전 값을 즉시 반환)은 오프셋입니다.

연도	분기	판매 직원	수익	상대값
2007	Q1	Smith	1000	
2007	Q2	Jones	2000	
2007	Q3	Wilson	1500	
2007	Q4	Harris	3000	
2008	Q1	Smith	4000	1000
2008	Q2	Jones	3400	2000
2008	Q3	Wilson	2000	1500
2008	Q4	Harris	1700	3000

비즈니스 질문으로 표현하는 경우, 수익은 이전 연도 동 분기에 같은 판매 직원에 의해 발생한 수익을 반환합니다.

단어 단위 계산식으로 표현하는 경우, 수익은 [연도](분기 차원)의 값이 [연도] 개체의 값 목록에서 이전 값이고 [분기] 및 [판매 직원](하위 축 차원)의 값이 현재 행과 같은 행에 있는 행의 [수익](식)을 반환합니다.

## 관련 정보

[RelativeValue \[페이지 217\]](#)

## 8.2.1 분리 차원 및 RelativeValue 함수

RelativeValue 함수는 분리 차원의 값 목록을 사용하여 비교 행을 찾습니다.

함수는 해당 함수에서 지정한 식의 비교 값을 반환하여, 이 값은 분리 차원 목록과 떨어져 있는 행의 오프셋숫자입니다.

따라서 분리 차원의 정렬 순서는 함수의 출력값을 결정하는 데 매우 중요합니다.



### 다중 분리 차원

아래 표의 RelativeValue 열에는 다음 수식이 포함됩니다.

```
RelativeValue([수익];([연도];[분기]);-1)
```

- [수익]은 식을 나타내고,
- ([연도];[분기])는 분리 차원을 나타내며,
- -1(함수가 목록에서 이전 값을 즉시 반환)은 오프셋입니다.

연도	분기	판매 직원	수익	상대값
2007	Q1	Smith	1000	
2007	Q2	Smith	2000	
2007	Q3	Smith	1500	
2007	Q4	Smith	3000*	
2007	Q1	Jones	4000	
2007	Q2	Jones	3400	
2007	Q3	Jones	2000	
2007	Q4	Jones	1700	
2008	Q1	Smith	5000**	3000*
2008	Q2	Smith	3000***	5000**
2008	Q3	Smith	2700****	3000***
2008	Q4	Smith	6800	2700****

비즈니스 질문으로 표현하는 경우, 수익은 이전 분기에 같은 판매 직원에 의해 발생한 수익을 반환합니다.

단어 단위 계산식으로 표현하는 경우, 수익은 [연도] 및 [분기]의 값이 ([연도];[분기]) 값 목록에서 이전 값을 나타내고 [판매 직원]의 값이 현재 행과 같은 행에 있는 행의 [수익]을 반환합니다.

이 함수는 분리 차원의 값 목록을 사용하여 비교 수익을 찾습니다

연도	분기	
2007	Q1	
2007	Q2	
2007	Q3	
2007	Q4	*
2008	Q1	**
2008	Q2	***
2008	Q3	****
2008	Q4	

분리 차원의 정렬 순서는 함수의 출력값을 결정합니다. 표에서 \*는 정렬 순서를 표시합니다.

## 관련 정보

[RelativeValue \[페이지 217\]](#)

## 8.2.2 분리 차원 및 섹션

분리 차원은 보고서의 섹션 마스터 셀에 위치할 수 있습니다.



예  
섹션 머리글의 분리 차원

아래 표의 RelativeValue 열에는 다음 수식이 포함됩니다.

```
RelativeValue([수익];([연도];[분기]);-1)
```

2007

분기	판매 직원	수익	상대값
Q1	Smith	1000	
Q2	Smith	2000	
Q3	Smith	1500	
Q4	Smith	3000*	
Q1	Jones	4000	
Q2	Jones	3400	
Q3	Jones	2000	
Q4	Jones	1700	

2008

분기	판매 직원	수익	상대값
Q1	Smith	5000**	3000*
Q2	Smith	3000***	5000**
Q3	Smith	2700****	3000***
Q4	Smith	6800	2700****

이 함수는 분리 차원의 값 목록을 사용하여 비교 수익을 찾습니다

연도	분기	
2007	Q1	
2007	Q2	
2007	Q3	
2007	Q4	*
2008	Q1	**
2008	Q2	***
2008	Q3	****
2008	Q4	

분리 차원의 정렬 순서는 함수의 출력값을 결정합니다. 표에서 \*는 정렬 순서를 표시합니다.

## 관련 정보

[RelativeValue \[페이지 217\]](#)

### 8.2.3 차원 분리 순서

분리 차원에 있는 값 목록의 정렬 순서가 `RelativeValue`의 출력값을 결정하므로 분리 차원의 지정 순서는 함수의 출력값에 영향을 줍니다.



#### 차원 분리 순서

아래 표의 `RelativeValue` 열에는 다음 수식이 포함됩니다.

```
RelativeValue([수익];([연도];[분기]);-1)
```

연도	분기	판매 직원	수익	상대값
2007	Q1	Smith	1000	
2007	Q2	Smith	2000	
2007	Q3	Smith	1500	
2007	Q4	Smith	3000*	
2007	Q1	Jones	4000	
2007	Q2	Jones	3400	
2007	Q3	Jones	2000	
2007	Q4	Jones	1700	
2008	Q1	Smith	5000**	3000*
2008	Q2	Smith	3000***	5000**
2008	Q3	Smith	2700****	3000***
2008	Q4	Smith	6800	2700****

비즈니스 질문으로 표현하는 경우, 수식은 이전 분기에 같은 판매 직원에 의해 발생한 수익을 반환합니다.

분리 차원의 정렬 순서는 다음과 같습니다.

연도	분기
2007	Q1
2007	Q2



연도	분기	
2007	Q3	
2007	Q4	*
2008	Q1	**
2008	Q2	***
2008	Q3	****
2008	Q4	

함수가 다음과 같이 변경됩니다.

```
RelativeValue([수익];([분기];[연도]);-1)
```

분리 차원의 정렬 순서는 다음과 같습니다.

분기	연도	
Q1	2007	*
Q1	2008	**
Q2	2007	***
Q2	2008	****
Q3	2007	*****
Q3	2008	*****
Q4	2007	*****
Q4	2008	*****

정렬 순서는 함수 결과에 다음과 같은 영향을 줍니다.

연도	분기	판매 직원	수익	상대값
2007	Q1	Smith	1000*	
2007	Q2	Smith	2000***	
2007	Q3	Smith	1500*****	
2007	Q4	Smith	3000*****	
2007	Q1	Jones	4000	
2007	Q2	Jones	3400	
2007	Q3	Jones	2000	
2007	Q4	Jones	1700	
2008	Q1	Smith	5000**	1000*
2008	Q2	Smith	3000****	2000***
2008	Q3	Smith	2700*****	1500*****
2008	Q4	Smith	6800*****	3000*****

비즈니스 질문으로 표현하는 경우, 수식이 이제 이전 연도 동 분기에 같은 판매 직원에 의해 발생한 수익을 반환합니다.

분리 차원의 정렬 순서를 변경하는 수식의 결과도 변경됩니다. 테이블에서 \*은 정렬 순서를 나타냅니다.

## 관련 정보

[RelativeValue \[페이지 217\]](#)

## 8.2.4 분리 차원 및 정렬

분리 차원에 있는 값 목록의 정렬 순서가 함수의 출력값을 결정하므로 분리 차원의 차원에 적용된 정렬은 함수의 출력값에 영향을 줍니다.

### 예

#### 분리 차원에 적용된 사용자 지정 정렬

아래 표의 RelativeValue 열에는 다음 수식이 포함됩니다.

```
RelativeValue([수익];([연도];[분기]);-1)
```

사용자 지정 정렬(1분기, 2분기, 4분기, 3분기)은 [분기]에 적용되어 함수에 다음 결과를 제공합니다.

연도	분기	판매 직원	수익	상대값
2007	Q1	Smith	1000	
2007	Q2	Smith	2000	
2007	Q4	Smith	3000	
2007	Q3	Smith	1500*	
2007	Q1	Jones	4000	
2007	Q2	Jones	3400	
2007	Q4	Jones	1700	
2007	Q3	Jones	2000	
2008	Q1	Smith	5000**	1500*
2008	Q2	Smith	3000***	5000**
2008	Q4	Smith	6800****	3000***
2008	Q3	Smith	2700	6800****

분리 차원의 정렬 목록은 다음과 같습니다.

연도	분기	
2007	Q1	
2007	Q2	
2007	Q4	
2007	Q3	*
2008	Q1	**
2008	Q2	***
2008	Q4	****
2008	Q3	

표에서 \*는 정렬 순서를 표시합니다.

## 관련 정보

[RelativeValue \[페이지 217\]](#)

## 8.2.5 크로스탭에서 RelativeValue 사용

크로스탭에서 RelativeValue 함수는 수직 테이블에서와 동일한 방법으로 작용합니다.

크로스탭에서 데이터의 레이아웃은 함수의 출력값에 영향을 주지 않습니다.

## 관련 정보

[RelativeValue \[페이지 217\]](#)

# 중요 법적 면책 사항 및 법률 정보

## 코딩 샘플

이 문서에 포함된 어떠한 소프트웨어 코딩 및/또는 코드 라인/문자열 ("코드")도 예시 목적으로만 사용되며 운영 시스템 환경에의 사용을 의도하지 않습니다. 코드는 특정 코딩의 구문 또는 구문 지정 규칙을 좀 더 잘 설명하고 표시하기 위해서만 사용됩니다. SAP는 이 문서에 제공된 코드의 정확성과 완전성을 보증하지 않으며, SAP의 의도나 중과실로 인해 발생한 손해가 아닌 한, 코드의 사용으로 인해 발생한 오류나 손해 부분에 대한 책임을 지지 않습니다.

## 성 중립적 언어 사용

SAP 문서는 가능한 범위에서 성 중립성을 유지합니다. 문맥에 따라 독자의 경우 직접 "사용자"로 언급되고, 성 중립적 명사(예: "영업 직원" 또는 "근무일")가 사용됩니다. 양쪽 성별을 모두 나타낼 때 3인칭 단수를 배제할 수 없거나 성 중립적 명사가 없는 경우, SAP는 명사 및 대명사의 남성형을 사용할 권리가 있습니다. 이는 문서의 이해를 돕기 위한 것입니다.

## 인터넷 하이퍼링크

SAP 문서에는 인터넷으로 연결된 하이퍼링크가 포함될 수 있습니다. 이러한 하이퍼링크는 관련 정보를 찾기 위한 힌트로 사용됩니다. SAP는 이와 관련된 정보의 가용성 및 정확성, 또는 이 정보가 특정 목적으로 사용될 가능성에 대해 보증하지 않습니다. SAP는 SAP의 중과실 또는 고의적 불법 행위에 의해 손해가 발생한 경우 외에, 관련 정보의 사용으로 발생한 어떠한 손해에 대해서도 책임을 지지 않습니다. 투명성을 위해 모든 링크가 범주별로 분류되어 있으니 참고해 주시기 바랍니다(참조: <https://help.sap.com/viewer/disclaimer>).





**go.sap.com/registration/  
contact.html**

© 2018 SAP SE 또는 SAP 계열사. 모든 권한 보유.

본 발행물의 어떠한 부분도 SAP SE 또는 SAP 계열사의 명시적 허가 없이는 어떠한 형태나 목적으로도 복제 또는 배포할 수 없습니다. 본 문서의 정보는 사전 예고 없이 변경될 수 있습니다.

SAP SE 및 그 유통업자가 판매하는 일부 소프트웨어 제품에는 다른 소프트웨어 공급업체가 소유한 소프트웨어 구성 요소가 포함되어 있습니다. 국가별 제품 명세는 다를 수 있습니다.

이 문서는 SAP SE 또는 SAP 계열사에 의해 정보 전달 목적으로만 제공되며 어떠한 종류의 진술이나 보증도 포함되지 않습니다. SAP 또는 SAP 계열사는 이 문서의 오류나 누락 부분에 대해 책임을 지지 않습니다. SAP 또는 SAP 계열사 제품 및 서비스에 대한 유일한 보증은 해당 제품 및 서비스와 함께 제공되는 보증서에 명시된 내용으로 제한됩니다. 본 문서의 어떤 내용도 추가 보증의 근거로 해석할 수 없습니다.

SAP 및 본 문서에서 언급된 기타 SAP 제품, 서비스와 해당 로고는 독일 및 기타 국가에서 사용되는 SAP SE(또는 SAP 계열사)의 상표 또는 등록 상표입니다. 기타 언급된 모든 제품 및 서비스 이름은 각각의 해당 기업 상표입니다.

추가적인 상표 정보 및 공지사항은 <https://www.sap.com/corporate/en/legal/copyright.html>에서 확인하십시오.