

# 4강

# 각종연산자





## 1. 사칙연산

연산자	의미
+,-,*,/	각각 더하기, 빼기, 곱하기, 나누기
%, MOD	나머지

```

SELECT 1 + 2;
SELECT 5 - 2.5 AS DIFFERENCE;
SELECT 3 * (2 + 4) / 2 AS Number, 'Hello' AS Text;
SELECT 10 % 3;

```

문자열에 사칙연산을 가하면 0으로 인식

```

SELECT 'ABC' + 3;
SELECT 'ABC' * 3;
SELECT '1' + '002' * 3;
-- 숫자로 구성된 문자열은 숫자로 자동인식

```

```

SELECT
    OrderID + ProductID
FROM OrderDetails;

```

**SELECT**

```
ProductName,  
Price / 2 AS HalfPrice  
FROM Products;
```

## 2. 참/거짓 관련 연산자

**SELECT** TRUE, FALSE;

Number of Records: 1

TRUE	FALSE
1	0

**SELECT** !TRUE, NOT 1, !FALSE, NOT FALSE;

Number of Records: 1

!TRUE	NOT 1	!FALSE	NOT FALSE
0	0	1	1



※ MySQL에서는 TRUE는 1, FALSE는 0으로 저장됩니다

```
SELECT 0 = TRUE, 1 = TRUE, 0 = FALSE, 1 = FALSE;
```



Number of Records: 1			
<b>0 = TRUE</b>	<b>1 = TRUE</b>	<b>0 = FALSE</b>	<b>1 = FALSE</b>
0	1	1	0

```
SELECT * FROM Customers WHERE TRUE;
```



Number of Records: 91						
CustomerID	CustomerName	ContactName	Address	City	PostalCode	Country
1	Alfreds Futterkiste	Maria Anders	Obere Str. 57	Berlin	12209	Germany
2	Ana Trujillo Emparedados y helados	Ana Trujillo	Avda. de la Constitución 2222	México D.F.	05021	Mexico
3	Antonio Moreno Taquería	Antonio Moreno	Mataderos 2312	México D.F.	05023	Mexico
4	Around the Horn	Thomas Hardy	120 Hanover Sq.	London	WA1 1DP	UK

```
SELECT * FROM Customers WHERE FALSE;
```





연산자	의미
IS	양쪽이 모두 TRUE 또는 FALSE 일 때 TRUE
IS NOT	한쪽은 TRUE, 한쪽은 FALSE 일 때 TRUE

`SELECT TRUE IS TRUE;`  
`SELECT TRUE IS NOT FALSE;`  
`SELECT (TRUE IS FALSE) IS NOT TRUE;`

TRUE IS TRUE	TRUE IS NOT FALSE	(TRUE IS FALSE) IS NOT TRUE
1	1	1

연산자	의미
AND, &&	양쪽이 모두 TRUE일 때만 TRUE
OR,	한쪽이 TRUE면 TRUE

`SELECT TRUE AND FALSE, TRUE OR FALSE;`  
`SELECT 2 + 3 = 6 OR 2 * 3 = 6;`  
`SELECT * FROM Orders WHERE CustomerId = 15 AND EmployeeId = 4;`  
`SELECT * FROM Products WHERE ProductName = 'Tofu' OR CategoryId = 8;`  
`SELECT * FROM OrderDetails WHERE ProductId = 20  
 AND (OrderId = 10514 OR Quantity = 50);`



연산자	의미
=	양쪽 값이 같음
!=, <>	양쪽 값이 다름
>,<	(왼쪽, 오른쪽) 값이 더 큼
>=,<=	(왼쪽, 오른쪽) 값이 같거나 더 큼

```
SELECT 1 = 1, !(1 <> 1), NOT (1 < 2), 1 > 0 IS NOT FALSE;
```



Number of Records: 1			
1 = 1	!(1 <> 1)	NOT (1 < 2)	1 > 0 IS NOT FALSE
1	1	0	1

```
SELECT 'A' = 'A', 'A' != 'B', 'A' < 'B', 'A' > 'B';
```

```
SELECT 'Apple' > 'Banana' OR 1 < 2 IS TRUE;
```

※ MySQL의 기본 사칙연산자는 대소문자 구분을 하지 않습니다.

```
SELECT 'A' = 'a';
```



```
SELECT
```

```
ProductName, Price,  
Price > 20 AS EXPENSIVE  
FROM Products;
```

※ 테이블의 컬럼이 아닌 값으로 선택하기.

```
SELECT
```

```
ProductName, Price,  
NOT Price > 20 AS CHEAP  
FROM Products;
```

연산자	의미
BETWEEN {MIN} AND {MAX}	두 값 사이에 있음
NOT BETWEEN {MIN} AND {MAX}	두 값 사이가 아닌 곳에 있음

```
SELECT * FROM OrderDetails
```

```
WHERE ProductID BETWEEN 1 AND 4;
```

```
SELECT * FROM Customers
```

```
WHERE CustomerName BETWEEN 'b' AND 'c';
```



연산자	의미
IN(.....)	괄호 안의 값들 가운데 있음
NOT IN(.....)	괄호 안의 값들 가운데 없음

```
SELECT 1 + 2 IN (2, 3, 4) ;
SELECT 'Hello' IN (1, TRUE, 'hello') ;
```

```
SELECT * FROM Customers
WHERE City IN ('Torino', 'Paris', 'Portland', 'Madrid');
```



Number of Records: 8

CustomerID	CustomerName	ContactName	Address	City	PostalCode	Country
8	Bólido Comidas preparadas	Martín Sommer	C/ Araquil, 67	Madrid	28023	Spain
22	FISSA Fabrica Inter. Salchichas S.A.	Diego Roel	C/ Moralzarzal, 86	Madrid	28034	Spain
27	Franchi S.p.A.	Paolo Accorti	Via Monte Bianco 34	Torino	10100	Italy
48	Lonesome Pine Restaurant	Fran Wilson	89 Chiaroscuro Rd.	Portland	97219	USA
57	Paris spécialités	Marie Bertrand	265, boulevard Charonne	Paris	75012	France
69	Romero y tomillo	Alejandra Camino	Gran Vía, 1	Madrid	28001	Spain
74	Spécialités du monde	Dominique Perrier	25, rue Lauriston	Paris	75016	France
77	The Big Cheese	Liz Nixon	89 Jefferson Way Suite 2	Portland	97201	USA



연산자	의미
LIKE 'A%	A로 시작하는 문자 패턴
LIKE '%A'	A로 끝나는 문자 패턴
LIKE '%A%'	A를 포함하는 문자 패턴
NOT LIKE 'A%'	A로 시작하지 않는 문자 패턴
LIKE '..._...'	_ 개수만큼의 문자를 가진 패턴

```
SELECT * FROM Employees
WHERE Notes LIKE '%economics%';
```

```
SELECT
'HELLO' LIKE 'HEL__',
'HELLO' LIKE 'h___O',
'HELLO' LIKE 'HE__LO',
'HELLO' LIKE '__',
'HELLO' LIKE '_HELLO',
'HELLO' LIKE 'HEL_',
'HELLO' LIKE 'H_O';
```



'HELLO' LIKE 'HEL__'	'HELLO' LIKE 'h___O'	'HELLO' LIKE 'HE__LO'	'HELLO' LIKE '__'	'HELLO' LIKE '_HELLO'	'HELLO' LIKE 'HEL_'	'HELLO' LIKE 'H_O'
1	1	1	1	0	0	0



```
SELECT * FROM OrderDetails  
WHERE OrderID LIKE '1025_';
```



Number of Records: 29

OrderDetailID	OrderID	ProductID	Quantity
6	10250	41	10
7	10250	51	35
8	10250	65	15
9	10251	22	6
10	10251	57	15
11	10251	65	20
12	10252	20	40
13	10252	33	25
14	10252	60	40
15	10253	31	20
16	10253	39	42



연산자	의미
IS NULL	NULL 인 값 출력
IS NOT NULL	NULL이 아닌 값 출력

```
SELECT * FROM Shippers WHERE PHONE IS NOT NULL ;
```



Number of Records: 3

ShipperID	ShipperName	Phone
1	Speedy Express	(503) 555-9831
2	United Package	(503) 555-3199
3	Federal Shipping	(503) 555-9931


☞ 여기서 잠깐 !!

< **Customers** > 테이블의 Country가 'Spain'과 'Mexico'인 모든 데이터가 출력되도록 SQL문을 작성하여 아래 화면과 같은 결과를 도출하세요  
(단, IN 연산자 이용)

## 출력 결과

Number of Records: 10						
CustomerID	CustomerName	ContactName	Address	City	PostalCode	Country
2	Ana Trujillo Emparedados y helados	Ana Trujillo	Avda. de la Constitución 2222	México D.F.	05021	Mexico
3	Antonio Moreno Taquería	Antonio Moreno	Mataderos 2312	México D.F.	05023	Mexico
8	Bólido Comidas preparadas	Martín Sommer	C/ Araquil, 67	Madrid	28023	Spain
13	Centro comercial Moctezuma	Francisco Chang	Sierras de Granada 9993	México D.F.	05022	Mexico
22	FISSA Fabrica Inter. Salchichas S.A.	Diego Roel	C/ Moralzarzal, 86	Madrid	28034	Spain
29	Galería del gastrónomo	Eduardo Saavedra	Rambla de Cataluña, 23	Barcelona	08022	Spain
30	Godos Cocina Típica	José Pedro Freyre	C/ Romero, 33	Sevilla	41101	Spain
58	Pericles Comidas clásicas	Guillermo Fernández	Calle Dr. Jorge Cash 321	México D.F.	05033	Mexico



## ☞ 여기서 잠깐 !!

< **Products** > 테이블의 ProductName의 자료 중에서 'chef'를 포함하는 모든 데이터가 출력되도록 SQL문을 작성하여 아래 화면과 같은 결과를 도출하세요  
(단, Like 연산자 이용)

### 출력 결과

Number of Records: 2

ProductID	ProductName	SupplierID	CategoryID	Unit	Price
4	Chef Anton's Cajun Seasoning	2	2	48 - 6 oz jars	22.00
5	Chef Anton's Gumbo Mix	2	2	36 boxes	21.35

# 4강

# 연습문제





# 1. 다음 <처리 조건>에 유의하여 <학생> 테이블에 대한 알맞은 SQL문을 작성하시오.

## <학생>

학번	이름	학년	수강과목	점수	연락처
180101	김기사	1	데이버베이스	90	123-4567
180102	박산업	2	전자계산기	80	234-5678
180103	최사무	3	정보통신	98	345-6789
180104	조합격	4	시스템분석	85	456-7890

## <처리 조건>

1. 학생 테이블에서 3학년과 4학년 학생의 학번과 이름을 검색하는 SQL문을 작성하시오.
2. 원소 함수 In을 사용하시오.  
원소 함수 In(value1, value2.....)
3. SQL 문의 끝에는 세미콜론(;)으로 표시하시오.

정답



2. 다음은 <회원> 테이블에서 '이름'이 "이"로 시작하는 회원들을 '가입일' 순으로 내림차순 정렬하는 SQL문을 작성하시오.

&lt;회원&gt;

회원번호	이름	성별	가입일
1001	이진성	남	2021-06-23
1002	조이령	여	2021-06-24
1003	최민수	남	2021-06-28
1004	김차희	여	2021-07-03
1005	이미경	여	2021-07-10

정답



3. <EMP>(사원) 테이블을 참고하여 직속상사(MGR)가 NULL 인 사원의 이름(ENAME)과 직급(JOB)을 조회하는 SQL문을 <결과>와 같이 출력되도록 작성하시오.

&lt;EMP&gt;

EMPNO	ENAME	JOB	MGR	HIREDATE	SAL	COMM	DEPTNO
7369	SMITH	CLERK	7902	80/12/17	800		20
7499	ALLEN	SALESMAN	7698	81/02/20	1600	300	30
7521	WARD	SALESMAN	7698	81/02/22	1250	500	30
7566	JONES	MANAGER	7839	81/04/02	2975		20
7654	MARTIN	SALESMAN	7698	81/09/28	1250	1400	30
7698	BLAKE	MANAGER	7839	81/05/01	2850		30
7782	CLARK	MANAGER	7839	81/06/09	2450		10
7788	SCOTT	ANALYST	7566	87/07/13	3000		20
7839	KING	PRESIDENT		81/11/17	5000		10
7844	TURNER	SALESMAN	7698	81/09/08	1500	0	30
7876	ADAMS	CLERK	7788	87/07/13	1100		20
7900	JAMES	CLERK	7698	81/12/03	950		30
7902	FORD	ANALYST	7566	81/12/03	3000		20
7934	MILLER	CLERK	7782	82/01/23	1300		10

&lt;결과&gt;

ENAME	JOB
KING	PRESIDENT