

# Part 4.

---

SELECT 옵션02



## 03 GROUP BY

### GROUP BY - 데이터의 그룹화

```
USE sqlDB;  
-- SELECT userID,amount FROM buyTbl ORDER BY userID;  
SELECT userID,SUM(amount) FROM buyTbl GROUP BY userID;
```

### GROUP BY + 별칭 지정

```
use sqldb;  
SELECT userID,SUM(amount) AS '총 구매개수' FROM buyTbl GROUP BY userID;
```

```
use sqldb;  
select userID AS '사용자 아이디' , sum(price*amount) AS '총구매액' from buytbl group by userid;
```

## 03 GROUP BY

### [집계함수]

함수명	설명
AVG()	평균을 구한다
MIN()	최소값을 구한다
MAX()	최대값을 구한다
COUNT()	행의 개수를 센다
COUNT(DISTINCT)	개수를 센다. (중복은 1개만 인정)
STDEV()	표준편차를 구한다
VAR()	분산을 구한다

## 03 GROUP BY

### AVG()

USE sqlDB;

SELECT AVG(amount) AS [ 평균구매 개수] FROM buyTbl;

select AVG(amount) as '평균 구매 개수' from buytbl;

select userid,AVG(amount) as '평균 구매 개수' from buytbl group by userid;

### MAX(),MIN()

USE sqlDB;

SELECT AVG(amount\*1.0) AS [ 평균구매 개수] FROM buyTbl;

SELECT AVG(CAST(amount AS DECIMAL(10,6))) AS [ 평균구매 개수] FROM buyTbl;

## 03 GROUP BY

### 응용 - usertbl 가장 큰키, 가장 작은키

```
select name,height from usertbl where height=(select Max(height) from usertbl)
or height=(select min(height) from usertbl);
```

### count(\*)

```
select count(*) from usertbl; -- count(*) 전체 행개수
select count(mobile1) as '휴대폰이 있는 사용자' from usertbl;
-- count(mobile1):null을 제외한 행개수
```

## 03 중간 문제

buyTbl

157 %

결과 메시지

	num	userID	prodName	groupName	price	amount
1	1	KBS	운동화	NULL	45	2
2	2	KBS	노트북	전자	1500	1
3	3	JYP	모니터	전자	300	1
4	4	BBK	모니터	전자	300	5
5	5	KBS	청바지	의류	75	3
6	6	BBK	메모리	전자	120	10
7	7	SSK	책	서적	22	5
8	8	EJW	책	서적	22	2
9	9	EJW	청바지	의류	75	1
10	10	BBK	운동화	NULL	45	2
11	11	EJW	책	서적	22	1
12	12	BBK	운동화	NULL	45	2
13	13	JBH	구두	NULL	50	1

1 buytbl에서 userid 별로 구매량(amount)의 합을 출력하세요

3 usertbl에서 키의 평균값을 구하세요

4 buy테이블에서 최대구매량과 최소구매량을 userid와함께 출력하세요

5 buytbl의 groupname 의 개수를 출력하세요

# Part 5.

---

SELECT 옵션03



# 01 HAVING

## HAVING - 집계함수에 대한 조건절

[오류]

```
select userid,sum(price*amount) as '총구매액' from buytbl where sum(price*amount) >1000 group by userid ;
```

- 집계함수는 where절에 쓸수 없다

[정정]

```
select userid,sum(price*amount) as '총구매액' from buytbl group by userid Having sum(price*amount) > 1000;
```

-주의 : Having 은 반드시 group by 뒤에서 사용해야한다



# 01 HAVING

## HAVING 정렬

```
select userid,sum(price*amount) as '총구매액' from buytbl group by userid having  
sum(price*amount) > 1000  
order by Sum(price*amount);
```

# 01 ROLLUP()

## ROLLUP() - 소합계

```
select num, groupName, SUM(price*amount) AS '비용' FROM buyTbl
GROUP BY groupName, num With ROLLUP;
```

결과	메시지
1	num groupName 비용
2	1 NULL 60
3	10 NULL 60
4	12 NULL 60
5	NULL NULL 180
6	7 서적 75
7	8 서적 30
8	11 서적 15
9	NULL 서적 120
10	5 의류 150
11	9 의류 50
12	NULL 의류 200
13	2 전자 1000
14	3 전자 200
15	4 전자 1000
16	6 전자 800
17	NULL 전자 3000
18	NULL NULL 3500

## ROLLUP() - 총합계

```
select num, groupName, SUM(price*amount) AS '비용' FROM buyTbl
GROUP BY groupName With ROLLUP;
```

결과	메시지
	groupName 비용
1	NULL 180
2	서적 120
3	의류 200
4	전자 3000
5	NULL 3500

## 03 중간 문제

buyTbl

	num	userID	prodName	groupName	price	amount
1	1	KBS	운동화	NULL	45	2
2	2	KBS	노트북	전자	1500	1
3	3	JYP	모니터	전자	300	1
4	4	BBK	모니터	전자	300	5
5	5	KBS	청바지	의류	75	3
6	6	BBK	메모리	전자	120	10
7	7	SSK	책	서적	22	5
8	8	EJW	책	서적	22	2
9	9	EJW	청바지	의류	75	1
10	10	BBK	운동화	NULL	45	2
11	11	EJW	책	서적	22	1
12	12	BBK	운동화	NULL	45	2
13	13	JBH	구두	NULL	50	1

1. prodName별로 그룹화 한뒤 userID / prodName/Price\*amount 순으로 출력될 수 있도록 설정

```
SELECT userid,prodName ,SUM(price*amount) AS '총합' FROM buyTbl GROUP BY userid,prodName
```

2. 1 번 명령어에서 price\*amount 값이 1000이상인 행만 출력

```
SELECT userid,prodName ,SUM(price*amount) AS '총합' FROM buyTbl GROUP BY userid,prodName HAVING SUM(price*amount)>=1000
```

3. price 가격이 가장 큰 행과 작은 행의 userid , prodName,price을 출력

```
SELECT userid,prodName,price FROM buyTbl WHERE price=(SELECT MAX(price) FROM buyTbl) OR price=(SELECT MIN(price) FROM buytbl)
```

4. 다음 행중에 그룹네임이 있는 행만 출력

```
답 : SELECT COUNT(groupName) FROM buyTbl
```

5 prodName 별로 총합을 구해보세요(ROLLUP 사용)

```
SELECT prodName ,SUM(price*amount) AS '합계' FROM buyTbl GROUP BY prodName with ROLLUP;
```

ame)

END.

---

고생하셨습니다.

