

통계프로그래밍 HomeWork #1

목차

1. 과제방 DATABASE과 실습 TABLE 생성하기
2. 과제방 실습 데이터 LOAD 하기
3. 실습 1번
4. 실습 2번
5. 실습 3번
6. 실습 4번

과제방 DATABASE과 실습 TABLE 생성하기

```
CREATE DATABASE `과제방`;
USE `과제방`;
CREATE TABLE IF NOT EXISTS `실습` (
  `기업명` VARCHAR(15),
  `고용인원` VARCHAR(10),
  `기업형태` VARCHAR(10),
  `기업규모` VARCHAR(10)
);
```

데이터베이스와 테이블 생성 결과

```
mysql> CREATE DATABASE `과제방`; USE `과제방`; CREATE TABLE IF NOT EXISTS `실습`
( `기업명` VARCHAR(15), `고용인원` varchar(10), `기업형태` VARCHAR(10), `기업규모`
VARCHAR(10));
Query OK, 1 row affected (0.00 sec)

Database changed
Query OK, 0 rows affected (0.02 sec)
```

데이터베이스 테이블 결과

```
mysql> describe `실습`;
+-----+-----+-----+-----+-----+-----+
| Field          | Type          | Null | Key | Default | Extra |
+-----+-----+-----+-----+-----+-----+
| 기업명         | varchar(15)   | YES  |     | NULL    |       |
| 고용인원       | varchar(10)   | YES  |     | NULL    |       |
| 기업형태       | varchar(10)   | YES  |     | NULL    |       |
| 기업규모       | varchar(10)   | YES  |     | NULL    |       |
+-----+-----+-----+-----+-----+-----+
4 rows in set (0.00 sec)
```

과제방 실습 데이터 LOAD 하기

```
LOAD DATA LOCAL INFILE 'C:/data.txt' INTO TABLE `실습`
FIELDS TERMINATED BY '\t' OPTIONALLY ENCLOSED BY '"'
ESCAPED BY '\\'
LINES TERMINATED BY '\r\n'
IGNORE 1 LINES;
```

실습 데이터 LOAD 결과

```
mysql> LOAD DATA LOCAL INFILE '~/Downloads/data.txt' INTO TABLE `실습` FIELDS TE
RMINATED BY '\t' OPTIONALLY ENCLOSED BY '"' ESCAPED BY '\\' LINES TERMINATED BY
'\r\n' IGNORE 1 LINES;
Query OK, 228 rows affected, 2 warnings (0.01 sec)
Records: 228 Deleted: 0 Skipped: 0 Warnings: 2
```

※ 실습환경이 macOS Sierra 이기 때문에, 그에 맞춰 경로 수정을 했다.

1. 실습 Table에서 고용인원의 평균, 합계를 산출하시오

```
SELECT CONCAT(ROUND(AVG(SUBSTRING_INDEX(`고용인원`, '명', 1)), 2), '명') AS '평균',
CONCAT(SUM(SUBSTRING_INDEX(`고용인원`, '명', 1)), '명') AS '합계' FROM `실습`;
```

고용인원 평균, 합계 결과

```
mysql> SELECT CONCAT(ROUND(AVG(SUBSTRING_INDEX(`고용인원`, '명', 1)), 2), '명')
AS '평균', CONCAT(SUM(SUBSTRING_INDEX(`고용인원`, '명', 1)), '명') AS '합계' FRO
M `실습`;
+-----+-----+
| 평균      | 합계      |
+-----+-----+
| 23.54명    | 5366명    |
+-----+-----+
1 row in set (0.00 sec)
```

구분자로 값 추출하기

- SUBSTRING_INDEX(str, delim, count)
Returns the substring from string str before count occurrences of the delimiter delim. If count is positive, everything to the left of the final delimiter (counting from the left) is returned. If count is negative, everything to the right of the final delimiter (counting from the right) is returned. SUBSTRING_INDEX() performs a case-sensitive match when searching for delim.

평균 함수와 합계 함수

- AVG([DISTINCT] expr)
Returns the average value of expr. The DISTINCT option can be used to return the average of the distinct values of expr.
If there are no matching rows, AVG() returns NULL.
- SUM([DISTINCT] expr)
Returns the sum of expr. If the return set has no rows, SUM() returns NULL. The DISTINCT keyword can be used to sum only the distinct values of expr.
If there are no matching rows, SUM() returns NULL.

반올림 함수

- ROUND(X), ROUND(X, D)
Rounds the argument X to D decimal places. The rounding algorithm depends on the data type of X. D defaults to 0 if not specified. D can be negative to cause D digits left of the decimal point of the value X to become zero.

문자열 합치는 함수

- CONCAT(str1, str2, ...)

Returns the string that results from concatenating the arguments. May have one or more arguments. If all arguments are nonbinary strings, the result is a nonbinary string. If the arguments include any binary strings, the result is a binary string. A numeric argument is converted to its equivalent nonbinary string form.

CONCAT() returns NULL if any argument is NULL.

2. 실습 Table에서 기업형태가 일반법인인 기업만 SELECT 하시오

```
SELECT * FROM `실습` WHERE `기업형태`='일반법인';
```

일반법인 SELECT 결과

(주)한경아이넷	90명	일반법인	중
기업			
(주)래트론	96명	일반법인	중
기업			
(주)래트론	96명	일반법인	중
기업			
+-----+-----+-----+-----+			
-----+			
176 rows in set (0.00 sec)			

'일반' 또는 '일반법', '반법' 만으로 '일반법인'을 검색하기

```
SELECT * FROM `실습` WHERE `기업형태` LIKE '%일반%';
SELECT * FROM `실습` WHERE `기업형태` LIKE '%일반법%';
SELECT * FROM `실습` WHERE `기업형태` LIKE '%반법%';
```

```
SELECT * FROM `실습` WHERE `기업형태` LIKE '일반%';
SELECT * FROM `실습` WHERE `기업형태` LIKE '일반법%';
SELECT * FROM `실습` WHERE `기업형태` LIKE '%반법인';
```

```
SELECT * FROM `실습` WHERE `기업형태` LIKE '일반법_';
SELECT * FROM `실습` WHERE `기업형태` LIKE '일반__';
SELECT * FROM `실습` WHERE `기업형태` LIKE '일__';
```

- '%': 어떤 문자(열), 그 길이가 0 또는 그 이상
- '_': 길이가 1인 문자

3. 실습 Table에서 기업규모가 소기업이며, 고용인원이 20인 미만인 기업만 SELECT 하시오

```
SELECT * FROM `실습` WHERE `기업규모`='소기업' AND `고용인원` < 20;
```

소기업, 고용인원 20미만 기업 결과

(주)아르고넷	19명	일반법인	
소기업			
(주)비타바이오	19명	외감	
소기업			
(주)비타바이오	19명	외감	
소기업			
+-----+-----+-----+-----+			
-----+			
114 rows in set, 170 warnings (0.00 sec)			

기업규모에서 '소규모' 검색하기

```
SELECT * FROM `실습` WHERE `기업규모` LIKE '소%' AND `고용인원` < 20;
SELECT * FROM `실습` WHERE `기업규모` LIKE '소__' AND `고용인원` < 20;
```

4. 실습 TABLE에서 전체 고용인원이 20 이상 50 미만인 기업을 SELECT 하시오.

```
SELECT * FROM `실습` WHERE
SUBSTRING_INDEX(`고용인원`, '명', 1) >= 20 &&
SUBSTRING_INDEX(`고용인원`, '명', 1) < 50;

SELECT * FROM `실습`
WHERE SUBSTRING_INDEX(`고용인원`, '명', 1) >= 20 AND
SUBSTRING_INDEX(`고용인원`, '명', 1) < 50;

SELECT * FROM `실습`
WHERE SUBSTRING_INDEX(`고용인원`, '명', 1)
BETWEEN 20 AND 49;
```

고용인원 20이상 50미만 결과

(주)인소프트	47명	일반법인	소기업
(주)인소프트	47명	일반법인	소기업
+-----+-----+-----+-----+			
---+			
80 rows in set (0.00 sec)			

BETWEEN 표현

- *expr* BETWEEN *min* AND *max* If *expr* is greater than or equal to *min* and *expr* is less than or equal to *max*, BETWEEN returns 1, otherwise it returns 0.
This is equivalent to the expression (*min* <= *expr* AND *expr* <= *max*) if all the arguments are of the same type. Otherwise type conversion takes place according to the rules described in Section 13.2, "Type Conversion in Expression Evaluation", but applied to all the three arguments.