

MySQL - CRUD Challenge & String Functions

1. Create a new database shirts_db;
2. Create a table name "Shirts". And here is the table structure.

Primary Key



shirt_id	article	color	shirt_size	last_worn
1	t-shirt	white	S	10
2	t-shirt	green	S	200
3	polo shirt	black	M	10
4	tank top	blue	S	50
5	t-shirt	pink	S	0
6	polo shirt	red	M	5
7	tank top	white	S	200
8	tank top	blue	M	15

```

mysql> USE shirts_db;
Database changed
mysql> SELECT DATABASE();
+-----+
| DATABASE() |
+-----+
| shirts_db |
+-----+
1 row in set (0.00 sec)

mysql> CREATE TABLE Shirts(
    ->     shirt_id INT PRIMARY KEY auto_increment,
    ->     article varchar(100),
    ->     color varchar(50),
    ->     shirt_size char(1) ,
    ->     last_worn int
    -> );
Query OK, 0 rows affected (0.04 sec)

mysql> DESC Shirts;
+-----+-----+-----+-----+-----+-----+
| Field | Type   | Null | Key  | Default | Extra       |
+-----+-----+-----+-----+-----+-----+
| shirt_id | int    | NO   | PRI  | NULL    | auto_increment
| article  | varchar(100) | YES  |      | NULL    |
| color    | varchar(50)  | YES  |      | NULL    |
| shirt_size | char(1)  | YES  |      | NULL    |
| last_worn | int    | YES  |      | NULL    |
+-----+-----+-----+-----+-----+-----+
5 rows in set (0.01 sec)

```

3.

```
INSERT INTO Shirts(article , color , shirt_size , last_worn)
VALUES('t-shirt', 'white', 'S', 10),
('t-shirt', 'green', 'S', 200),
('polo shirt', 'black', 'M', 10),
('tank top', 'blue', 'S', 50),
('t-shirt', 'pink', 'S', 0),
('polo shirt', 'red', 'M', 5),
('tank top', 'white', 'S', 200),
('tank top', 'blue', 'M', 15);
```

```
mysql> INSERT INTO Shirts(article , color , shirt_size , last_worn)
-> VALUES('t-shirt', 'white', 'S', 10),
-> ('t-shirt', 'green', 'S', 200),
-> ('polo shirt', 'black', 'M', 10),
-> ('tank top', 'blue', 'S', 50),
-> ('t-shirt', 'pink', 'S', 0),
-> ('polo shirt', 'red', 'M', 5),
-> ('tank top', 'white', 'S', 200),
-> ('tank top', 'blue', 'M', 15);
Query OK, 8 rows affected (0.01 sec)
Records: 8  Duplicates: 0  Warnings: 0
```

```
mysql> SELECT * FROM Shirts;
+-----+-----+-----+-----+-----+
| shirt_id | article | color | shirt_size | last_worn |
+-----+-----+-----+-----+-----+
| 1 | t-shirt | white | S | 10 |
| 2 | t-shirt | green | S | 200 |
| 3 | polo shirt | black | M | 10 |
| 4 | tank top | blue | S | 50 |
| 5 | t-shirt | pink | S | 0 |
| 6 | polo shirt | red | M | 5 |
| 7 | tank top | white | S | 200 |
| 8 | tank top | blue | M | 15 |
+-----+-----+-----+-----+
8 rows in set (0.00 sec)
```

4. Add A New Shirt Purple polo shirt, size M last worn 50 days ago

```
mysql> INSERT INTO Shirts(article , color , shirt_size , last_worn)
-> VALUES('polo-shirt' , 'purple' , 'M' , 50);
Query OK, 1 row affected (0.04 sec)

mysql> SELECT * FROM Shirts;
+-----+-----+-----+-----+-----+
| shirt_id | article | color | shirt_size | last_worn |
+-----+-----+-----+-----+-----+
| 1 | t-shirt | white | S | 10 |
| 2 | t-shirt | green | S | 200 |
| 3 | polo shirt | black | M | 10 |
| 4 | tank top | blue | S | 50 |
| 5 | t-shirt | pink | S | 0 |
| 6 | polo shirt | red | M | 5 |
| 7 | tank top | white | S | 200 |
| 8 | tank top | blue | M | 15 |
| 9 | polo-shirt | purple | M | 50 |
+-----+-----+-----+-----+
9 rows in set (0.00 sec)
```

5. SELECT all shirts

But Only Print Out Article and Color

```
mysql> SELECT article , color From Shirts;
+-----+-----+
| article | color |
+-----+-----+
| t-shirt | white |
| t-shirt | green |
| polo shirt | black |
| tank top | blue |
| t-shirt | pink |
| polo shirt | red |
| tank top | white |
| tank top | blue |
| polo-shirt | purple |
+-----+-----+
9 rows in set (0.00 sec)
```

6. SELECT all medium shirts

Print Out Everything Except shirt_id.

```
mysql> SELECT article , color , shirt_size , last_worn FROM Shirts
-> WHERE shirt_size = 'M';
+-----+-----+-----+-----+
| article | color | shirt_size | last_worn |
+-----+-----+-----+-----+
| polo shirt | black | M | 10 |
| polo shirt | red | M | 5 |
| tank top | blue | M | 15 |
| polo-shirt | purple | M | 50 |
+-----+-----+-----+-----+
4 rows in set (0.00 sec)
```

```
mysql> SELECT article , color , shirt_size , last_worn FROM Shirts
-> WHERE shirt_size LIKE 'M';
+-----+-----+-----+-----+
| article | color | shirt_size | last_worn |
+-----+-----+-----+-----+
| polo shirt | black | M | 10 |
| polo shirt | red | M | 5 |
| tank top | blue | M | 15 |
| polo-shirt | purple | M | 50 |
+-----+-----+-----+-----+
4 rows in set (0.00 sec)
```

7. Update all polo shirts Change their size to L.

```
mysql> UPDATE Shirts SET shirt_size = 'L' WHERE article LIKE "polo shirt"
```

```
mysql> SELECT * FROM Shirts;
+-----+-----+-----+-----+-----+
| shirt_id | article | color | shirt_size | last_worn |
+-----+-----+-----+-----+-----+
| 1 | t-shirt | white | S | 10 |
| 2 | t-shirt | green | S | 200 |
| 3 | polo shirt | black | L | 10 |
| 4 | tank top | blue | S | 50 |
| 5 | t-shirt | pink | S | 0 |
| 6 | polo shirt | red | L | 5 |
| 7 | tank top | white | S | 200 |
| 8 | tank top | blue | M | 15 |
| 9 | polo shirt | purple | L | 50 |
+-----+-----+-----+-----+-----+
9 rows in set (0.00 sec)
```

8. Update the shirt last worn 15 days ago. Change last_worn to 0

```
mysql> UPDATE Shirts SET last_worn = 0 WHERE last_worn = 15;
Query OK, 1 row affected (0.01 sec)
Rows matched: 1  Changed: 1  Warnings: 0
```

```
mysql> SELECT * FROM Shirts;
+-----+-----+-----+-----+-----+
| shirt_id | article | color | shirt_size | last_worn |
+-----+-----+-----+-----+-----+
| 1 | t-shirt | white | S | 10 |
| 2 | t-shirt | green | S | 200 |
| 3 | polo shirt | black | L | 10 |
| 4 | tank top | blue | S | 50 |
| 5 | t-shirt | pink | S | 0 |
| 6 | polo shirt | red | L | 5 |
| 7 | tank top | white | S | 200 |
| 8 | tank top | blue | M | 0 |
| 9 | polo shirt | purple | L | 50 |
+-----+-----+-----+-----+-----+
9 rows in set (0.00 sec)
```

9. Update all white shirts,
Change size to 'XS' and color to 'off white'

```
mysql> DESC Shirts;
+-----+-----+-----+-----+-----+
| Field | Type | Null | Key | Default | Extra |
+-----+-----+-----+-----+-----+
| shirt_id | int | NO | PRI | NULL | auto_increment |
| article | varchar(100) | YES | | NULL |
| color | varchar(50) | YES | | NULL |
| shirt_size | char(1) | YES | | NULL |
| last_worn | int | YES | | NULL |
+-----+-----+-----+-----+-----+
5 rows in set (0.05 sec)

mysql> ALTER TABLE Shirts
-> MODIFY COLUMN shirt_size char(2);
Query OK, 9 rows affected (0.21 sec)
Records: 9 Duplicates: 0 Warnings: 0

mysql> DESC Shirts;
+-----+-----+-----+-----+-----+
| Field | Type | Null | Key | Default | Extra |
+-----+-----+-----+-----+-----+
| shirt_id | int | NO | PRI | NULL | auto_increment |
| article | varchar(100) | YES | | NULL |
| color | varchar(50) | YES | | NULL |
| shirt_size | char(2) | YES | | NULL |
| last_worn | int | YES | | NULL |
+-----+-----+-----+-----+-----+
5 rows in set (0.00 sec)
```

```
mysql> SELECT * FROM Shirts;
+-----+-----+-----+-----+-----+
| shirt_id | article | color | shirt_size | last_worn |
+-----+-----+-----+-----+-----+
| 1 | t-shirt | white | S | 10 |
| 2 | t-shirt | green | S | 200 |
| 3 | polo shirt | black | L | 10 |
| 4 | tank top | blue | S | 50 |
| 5 | t-shirt | pink | S | 0 |
| 6 | polo shirt | red | L | 5 |
| 7 | tank top | white | S | 200 |
| 8 | tank top | blue | M | 0 |
| 9 | polo shirt | purple | L | 50 |
+-----+-----+-----+-----+-----+
9 rows in set (0.00 sec)

mysql> UPDATE Shirts SET shirt_size = 'XS' , color = 'off white' WHERE color LIKE "white";
Query OK, 2 rows affected (0.01 sec)
Rows matched: 2 Changed: 2 Warnings: 0

mysql> SELECT * FROM Shirts;
+-----+-----+-----+-----+-----+
| shirt_id | article | color | shirt_size | last_worn |
+-----+-----+-----+-----+-----+
| 1 | t-shirt | off white | XS | 10 |
| 2 | t-shirt | green | S | 200 |
| 3 | polo shirt | black | L | 10 |
| 4 | tank top | blue | S | 50 |
| 5 | t-shirt | pink | S | 0 |
| 6 | polo shirt | red | L | 5 |
| 7 | tank top | off white | XS | 200 |
| 8 | tank top | blue | M | 0 |
| 9 | polo shirt | purple | L | 50 |
+-----+-----+-----+-----+-----+
9 rows in set (0.00 sec)
```

10. Delete all old shirts Last worn 200 days ago

```
mysql> DELETE FROM Shirts WHERE last_worn = 200;
Query OK, 2 rows affected (0.01 sec)

mysql> SELECT * FROM Shirts;
+-----+-----+-----+-----+-----+
| shirt_id | article | color | shirt_size | last_worn |
+-----+-----+-----+-----+-----+
| 1 | t-shirt | off white | XS | 10 |
| 3 | polo shirt | black | L | 10 |
| 4 | tank top | blue | S | 50 |
| 5 | t-shirt | pink | S | 0 |
| 6 | polo shirt | red | L | 5 |
| 8 | tank top | blue | M | 0 |
| 9 | polo shirt | purple | L | 50 |
+-----+-----+-----+-----+-----+
7 rows in set (0.00 sec)
```

11. Delete all tank tops

```
mysql> DELETE FROM Shirts WHERE article LIKE "tank top";
Query OK, 2 rows affected (0.01 sec)

mysql> SELECT * FROM Shirts;
+-----+-----+-----+-----+-----+
| shirt_id | article | color | shirt_size | last_worn |
+-----+-----+-----+-----+-----+
| 1 | t-shirt | off white | XS | 10 |
| 3 | polo shirt | black | L | 10 |
| 5 | t-shirt | pink | S | 0 |
| 6 | polo shirt | red | L | 5 |
| 9 | polo shirt | purple | L | 50 |
+-----+-----+-----+-----+-----+
5 rows in set (0.00 sec)
```

12. Delete all shirts

```
mysql> DELETE FROM Shirts;
Query OK, 5 rows affected (0.01 sec)

mysql> SELECT * FROM Shirts;
Empty set (0.00 sec)

mysql> SHOW TABLES;
+-----+
| Tables_in_shirts_db |
+-----+
| shirts |
+-----+
1 row in set (0.03 sec)

mysql> DESC Shirts;
+-----+-----+-----+-----+-----+
| Field | Type | Null | Key | Default | Extra |
+-----+-----+-----+-----+-----+
| shirt_id | int | NO | PRI | NULL | auto_increment |
| article | varchar(100) | YES | | NULL | |
| color | varchar(50) | YES | | NULL | |
| shirt_size | char(2) | YES | | NULL | |
| last_worn | int | YES | | NULL | |
+-----+-----+-----+-----+-----+
5 rows in set (0.00 sec)
```

Bonus Question: Drop the entire shirts table.

```
mysql> DROP TABLE Shirts;
Query OK, 0 rows affected (0.03 sec)

mysql> SHOW TABLES;
Empty set (0.00 sec)

mysql> DESC Shirts;
ERROR 1146 (42S02): Table 'shirts_db.shirts' doesn't exist
mysql>
```

```
CREATE TABLE shirts
(
    shirt_id INT NOT NULL AUTO_INCREMENT,
    article VARCHAR(100),
    color VARCHAR(100),
    shirt_size VARCHAR(100),
    last_worn INT,
    PRIMARY KEY(shirt_id)
);
```

String Functions

Here, you learn n number function that helps to manipulate string data.

- UPPER
- LOWER
- REVERSE
- Palindrome
- CONCAT
- CONCAT_WS
- REPLACE
- CHAR_LENGTH
- WILDCARD
- % _
- LIKE
- NOT LIKE

%oing	bo <u> </u>	bat bad ban bay bam bar baz
- painting - Gardening - Swimming - Walking - Reading. etc...	↑ % can replace 0 or more character	replace 1 character only