1️⃣

Database (MYSQL)

1. INSERT VALUES

```
INSERT INTO TABLE(attr1, attr1)
VALUES (val1, val2);
SELECT * FROM TABLE
```

2. INSERT SELECT

1. Use the result of SELECT statement to insert rows into another table;

```
INSERT INTO TABLE1(attr1, attr2)
SELECT attr3, attr4 FROM TABLE2
```

> ⚠️ Here, table1 and table 2 could be the same table.

3. UPDATE

```
UPDATE TABLE SET attr1 = val1 WHERE attr2 = val2;
SELECT * FROM games
```

4. DELETE

```
DELETE FROM TABLE WHERE attr1 = val1;
SELECT * FROM games
```

5. DROP

```
DROP TABLE/VIEW name;
DROP INDEX name ON attr;
```

6. ALTER

```
ALTER TABLE name ADD attr type;
```

7. UNION

1. Connect two select statements.

```
SELECT attr1 FROM TABLE1 WHERE attr1 LIKE "xx"
UNION
SELECT attr2 FROM TABLE2 WHERE attr2 LIKE "xx"
```

8. JOIN

1. Use the value comes from TABLE2 while SELECT the value FROM TABLE1

LEFT JOIN

1. Receive the entire data of the first (left) table and get data from the second (right) table. Even if some data on the right is not exist in the table on the left, it will showed as NONE(empty).

RIGHT JOIN

1. The opposite situation as LEFT JOIN.

INNER JOIN

9. GROUP BY / ORDER BY/ HAVING

1. Sequence:

> ⚠️ *from-->where-->group by -->having --- >select --> order*

2. Usage and when to use:
    a. Usage:
        i. GROUP BY: Group the data being select from the table;
        ii. ORDER BY: Give ranks to the data being selected according to a specific column (could have several columns as reference).
        iii. HAVING: Alike the function of WHERE, after grouping, use HAVING to refilter the data.
    b. When to use:
        i. GROUP BY: Apart from the direct usage, when there are “**aggregate function”** existed in SELECT statement, we have to GROUP the rest of

columns.
Always only after WHERE clause.

> ⚠️ “A**ggregate function”: sum(), count(), avg(), min(), max()**

ii. ORDER BY: When we want to make order to the data being selected. The following methods will be ASC(Ascending) and Desc(Descending)
Always after GROUP BY

iii. HAVING: After grouping, give a further filtering to the data being selected, it is always combined with aggregate function.

10. CREATE TABLE/VIEW/INDEX

CREATE TABLE:

```
CREATE TABLE name(
attr1 type {NOT NULL} PRIMARY KEY,
attr2 type
);
INSERT INTO name(attr1, attr2) VALUES(val1, val2)
```

CREATE VIEW

1. Create a view names a SELECT query, this query may be used as if it were a table in many contexts.

```
CREATE VIEW name AS SELECT attr1, attr2 FROM TABLE WHERE attr1 < xxx;
```

CREATE INDEX

```
CREATE INDEX index_name ON table_name (column_name)
```

> ⚠️ ‘column_name’ limits the column needs index

11. Useful functions:

1. LIMIT

1. Used for limit the lines of data being selected.

```
SELECT * FROM TABLE LIMIT 0, 5(or a number);
```

2. DISTINCT

1. Make sure that the result being selected could be unique.

```
SELECT DISTINCT(attr1) FROM TABLE;
```