Limiting the Rows Selected

- **Restrict the rows returned by using the WHERE clause.**

```
SELECT    *|{[DISTINCT] column|expression [alias],...}
FROM      table
[WHERE    condition(s)];
```

- **The WHERE clause follows the FROM clause.**

Using where clause

1.SELECT employee_id, last_name, job_id, department_id FROM employees7 WHERE department_id = 90;

2. SELECT last_name, job_id, department_FROM employees7 WHERE last_name = 'Gagan'

Comparison Conditions

Operator	Meaning
=	Equal to
>	Greater than
>=	Greater than or equal to
<	Less than
<=	Less than or equal to
<>	Not equal to

3.SELECT last_name, salary FROM employees7 WHERE salary <= 3000;

Other Comparison Conditions

Operator	Meaning
BETWEEN ...AND...	Between two values (inclusive)
IN(set)	Match any of a list of values
LIKE	Match a character pattern
IS NULL	Is a null value

Using the BETWEEN Condition

4. SELECT last_name, salary FROM employees7 WHERE salary BETWEEN 2500 AND 3500;

Using the IN Condition

5.SELECT employee_id, last_name, salary, manager_id FROM employees7 WHERE manager_id IN (100, 101, 201);

Using the LIKE Condition

6. SELECT first_name FROM employees7 WHERE first_name LIKE 'S%';

7.SELECT last_name, hire_date FROM employee7 WHERE hire_date LIKE '%95';\

Using the NULL Conditions

8. SELECT last_name, manager_id FROM employees WHERE manager_id IS NULL;

Logical Conditions

Operator	Meaning
AND	Returns TRUE if *both* component conditions are true
OR	Returns TRUE if *either* component condition is true
NOT	Returns TRUE if the following condition is false

Using the AND Operator

9.SELECT employee_id, last_name, job_id, salary FROM employees7 WHER salary>= 10000 AND job_id LIKE '%MAN%';

Using the or Operator

10. SELECT employee_id, last_name, job_id, salary FROM employees7 WHER salary>= 10000 OR job_id LIKE '%MAN%';

Using the NOT Operator

11. SELECT last_name, job_id FROM employees WHERE job_id NOT IN ('IT_PROG', 'ST_CLERK', 'SA_REP');

Rules of Precedence

Order Evaluated	Operator
1	Arithmetic operators
2	Concatenation operator
3	Comparison conditions
4	IS [NOT] NULL, LIKE, [NOT] IN
5	[NOT] BETWEEN
6	NOT logical condition
7	AND logical condition
8	OR logical condition

Override rules of precedence by using parentheses.

ORDER BY Clause

12.SELECT last_name, job_id, department_id, hire_date FROM employees7 ORDER BY hire_date;

Sorting in Descending Order

13.SELECT last_name, job_id, department_id, hire_date FROM employees ORDER BY hire_date DESC;

Sorting by Column Alias

14.SELECT employee_id, last_name, salary*12 annsal FROM employees ORDER BY annsal;

Sorting by Multiple Columns

15.SELECT last_name, department_id, salary FROM employees ORDER BY department_id, salary DESC;