

SQL:

- SQL stands for structured query language its owned by oracle
- The version of the oracle is 23.1.1
- This language is basically used to communicate with the database.

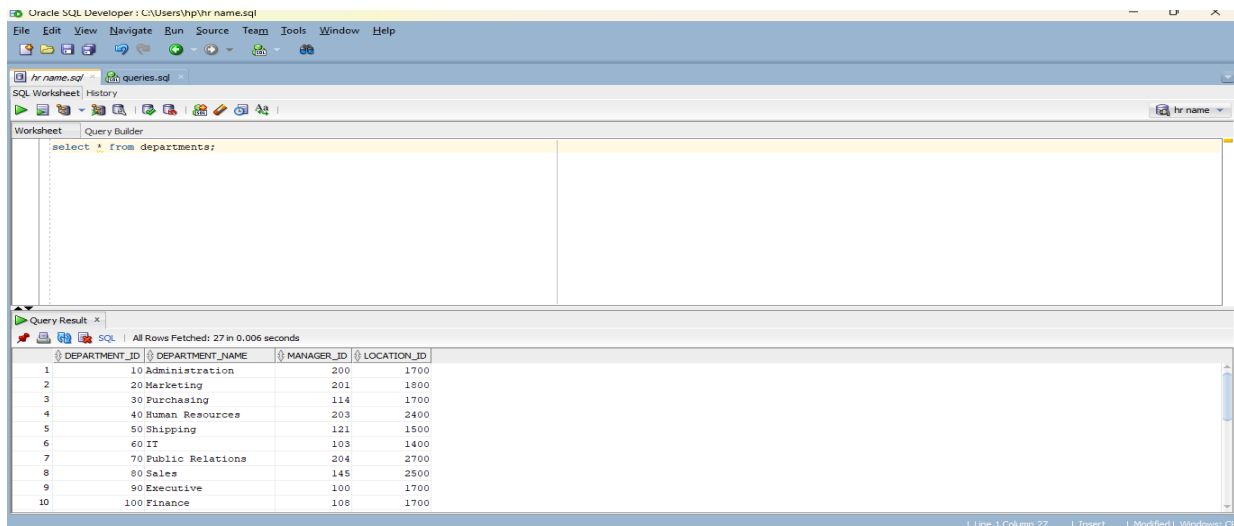
DRL or DQL:

- DRL stands for data retrieval language or data query language
- This command is used to retrieve data from the database

SELECT statements:

- Select statement is used to retrieve all the information from the database
- Syntax: select * from table_name;
- * - refers to retrieve all information from table

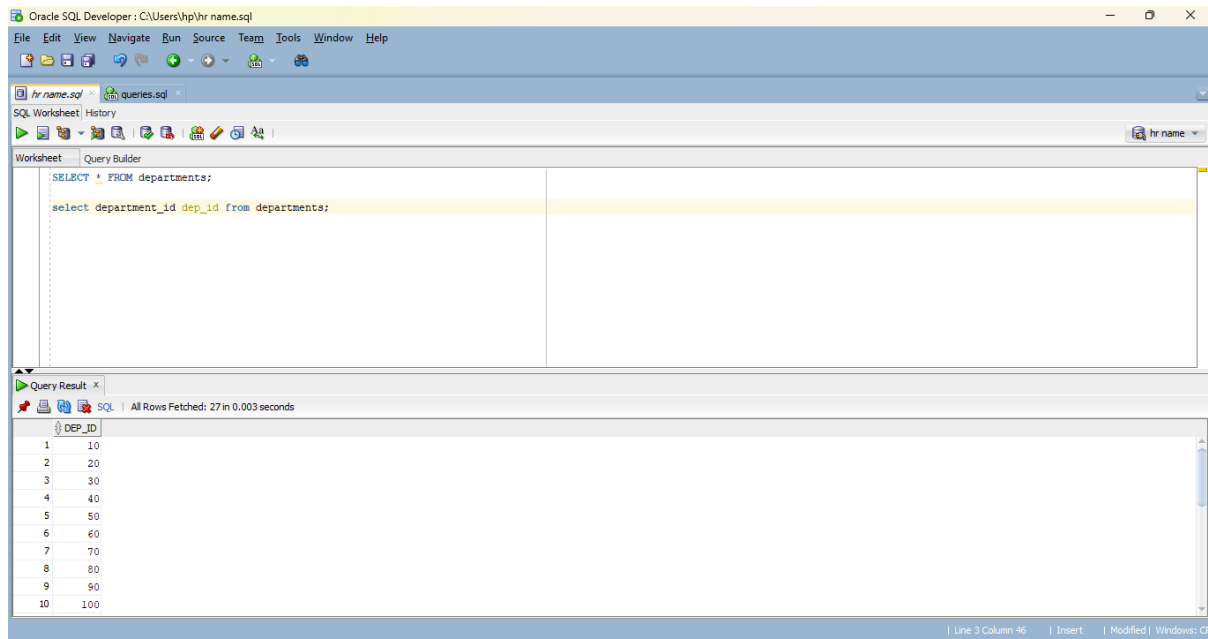
Select * from departments -> select and from is a keyword and departments is my tablename



Dual:

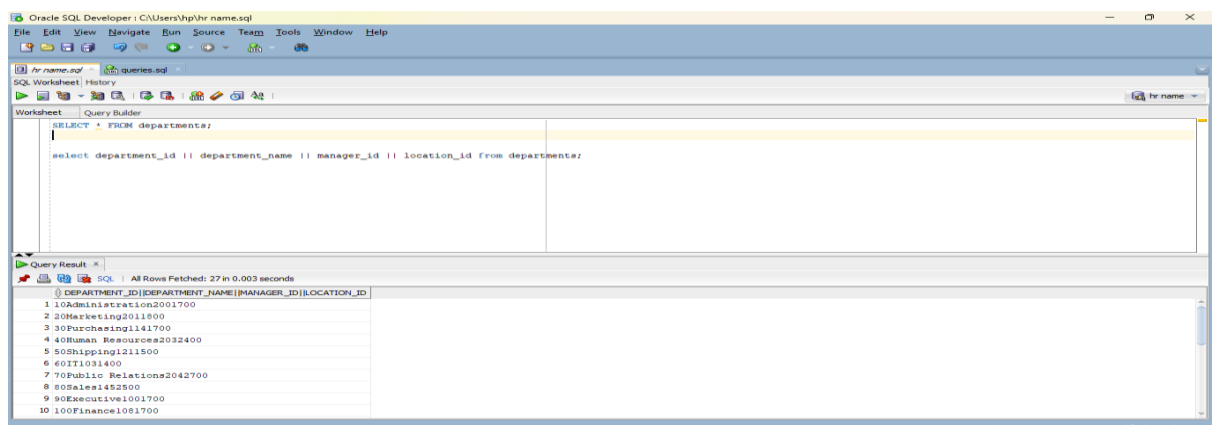
- Syntax: select expression from dual;
- Dual is a command which is used to get instance output or immediate output
- Dual has only one rows and one columns.
- It is used for arithmetic operations.

- Alias name is used to change the column name at the runtime
- As – as is a keyword for alias name
- Even if we not mention ‘as’ then we can only give extra space and alias name
- Syntax: select column_name as my_own_name from table_name.
- Syntax : select column_name my_own_name from table_name.



Pipeline operator (||):

- The pipeline operator is used to combine multiple columns into a single columns.



Null:

- Null is not equal to null
- Null is not equal to zero

- Is – it is a keyword we can only use this keyword in Null

Oracle SQL Developer: C:\Users\hp\hr name.sql

File Edit View Navigate Run Source Team Tools Window Help

hr name.sql queries.sql

SQL Worksheet History

Worksheet Query Builder

```
SELECT * FROM departments;

select * from departments where manager_id is null;
```

Query Result x

All Rows Fetched: 16 in 0.004 seconds

	DEPARTMENT_ID	DEPARTMENT_NAME	MANAGER_ID	LOCATION_ID
1	120	Treasury	(null)	1700
2	130	Corporate Tax	(null)	1700
3	140	Control And Credit	(null)	1700
4	150	Shareholder Services	(null)	1700
5	160	Benefits	(null)	1700
6	170	Manufacturing	(null)	1700
7	180	Construction	(null)	1700
8	190	Contracting	(null)	1700
9	200	Operations	(null)	1700
10	210	IT Support	(null)	1700

Line 5 Column 1 Insert Modified Windows: C

Oracle SQL Developer: C:\Users\hp\hr name.sql

File Edit View Navigate Run Source Team Tools Window Help

hr name.sql queries.sql

SQL Worksheet History

Worksheet Query Builder

```
SELECT * FROM departments;

select * from departments where manager_id is not null;
```

Query Result x

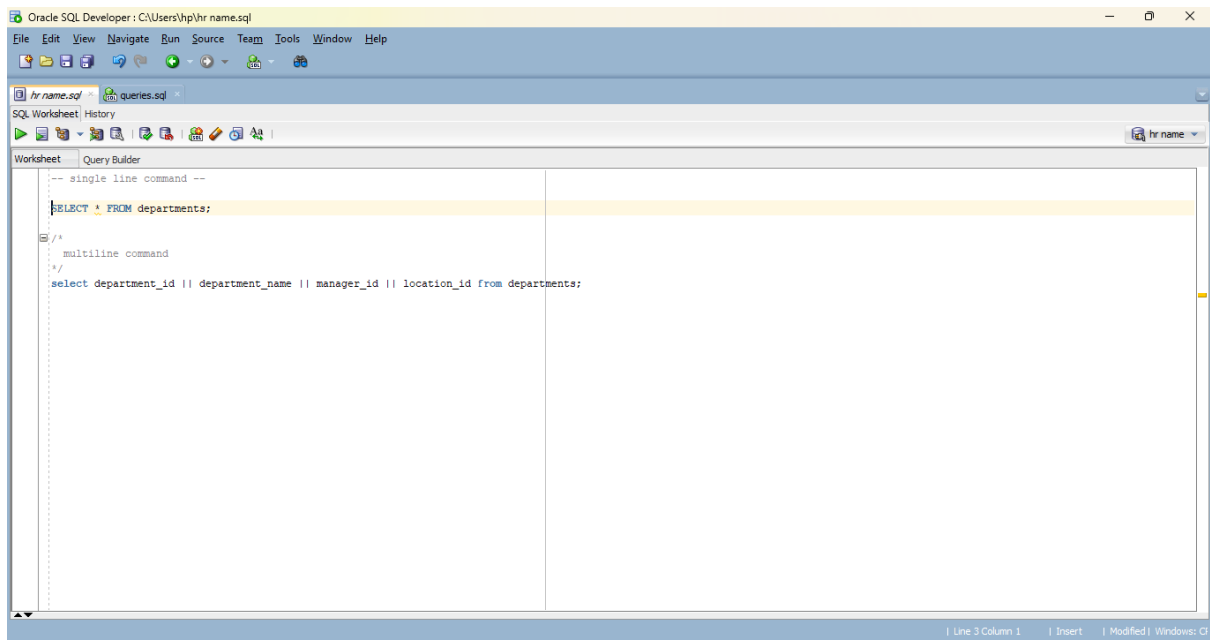
All Rows Fetched: 11 in 0.003 seconds

	DEPARTMENT_ID	DEPARTMENT_NAME	MANAGER_ID	LOCATION_ID
1	10	Administration	200	1700
2	20	Marketing	201	1800
3	30	Purchasing	114	1700
4	40	Human Resources	203	2400
5	50	Shipping	121	1500
6	60	IT	103	1400
7	70	Public Relations	204	2700
8	80	Sales	145	2500
9	90	Executive	100	1700
10	100	Finance	108	1700

Line 5 Column 1 Insert Modified Windows: C

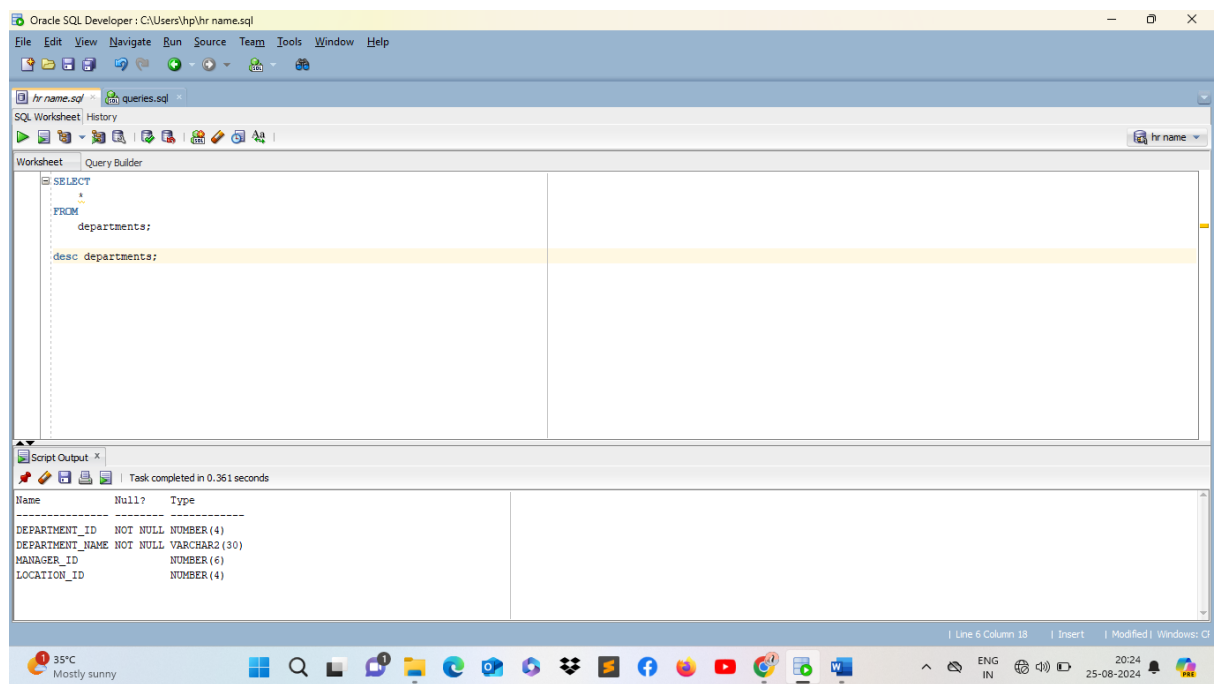
Command - line:

- In command line, we have two types of command
 - 1) Single-line command.
 - 2) Multi-level command.



Describe:

- Describe – it is used to describe the table
- And also if you want to see that what datatype we have used in the table then we can use describe
- Syntax: describe or desc table_name;

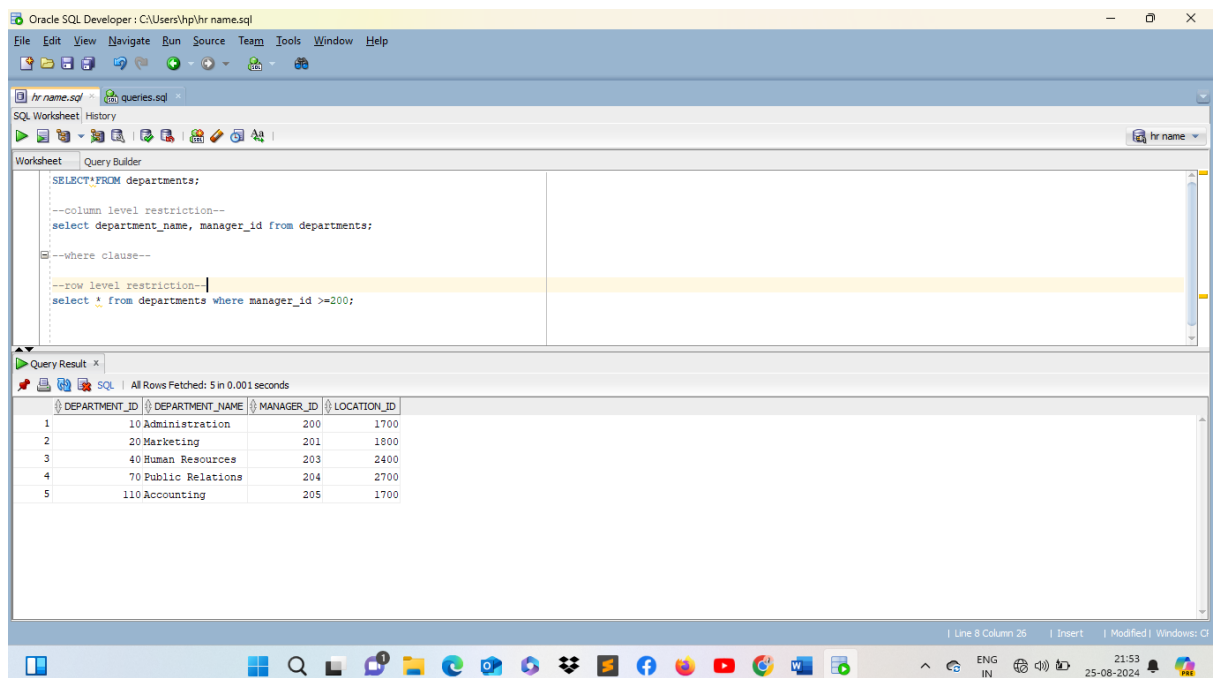


Datatypes in oracle SQL:

- Varchar2
- Char
- Number
- Date
- Time
- OB large object
- CLOB – char large object
- BLOB – binary large object

Row level restriction: (WHERE CLAUSE)

- Where clause is mainly used for row level restriction
- When you want to retrieve particular row then we need to use where clause



BETWEEN & AND operator:

- This operator is used to get an output within the given range
- Between, and both are keywords for this Queries
- Syntax: select * from table_name where column_name between (range value) and (range value)

Oracle SQL Developer: C:\Users\hp\hr name.sql

File Edit View Navigate Run Source Team Tools Window Help

hr name.sql queries.sql

SQL Worksheet History

Worksheet Query Builder

```

SELECT * FROM departments;

select * from departments where manager_id BETWEEN 200 and 250;

```

Query Result: All Rows Fetched: 5 in 0.002 seconds

	DEPARTMENT_ID	DEPARTMENT_NAME	MANAGER_ID	LOCATION_ID
1	10	Administration	200	1700
2	20	Marketing	201	1800
3	40	Human Resources	203	2400
4	70	Public Relations	204	2700
5	110	Accounting	205	1700

| Line 5 Column 1 | Insert | Modified | Windows: C

NOT BETWEEN:

- This operator is used to get specific value not within the given range
- For Example: if I give the condition not between 10 and 20. It will return every value except the given range that 10 and 20.
- Syntax: select * from table_name where column_name not between (range value) and (range value)

Oracle SQL Developer: C:\Users\hp\hr name.sql

File Edit View Navigate Run Source Team Tools Window Help

hr name.sql queries.sql

SQL Worksheet History

Worksheet Query Builder

```

SELECT * FROM departments;

select * from departments where manager_id not BETWEEN 200 and 250;

```

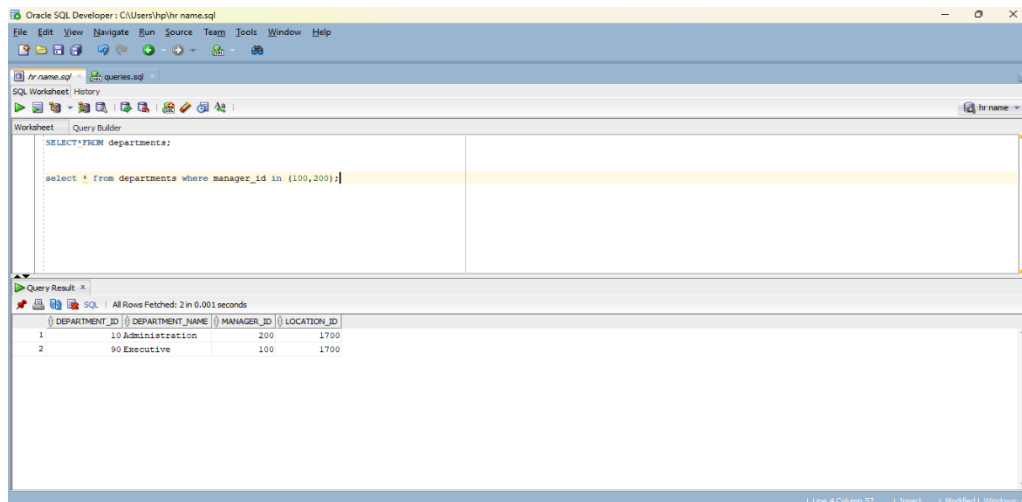
Query Result: All Rows Fetched: 6 in 0.004 seconds

	DEPARTMENT_ID	DEPARTMENT_NAME	MANAGER_ID	LOCATION_ID
1	30	Purchasing	114	1700
2	50	Shipping	121	1800
3	60	IT	103	1400
4	80	Sales	145	2500
5	90	Executive	100	1700
6	100	Finance	108	1700

| Line 5 Column 1 | Insert | Modified | Windows: C

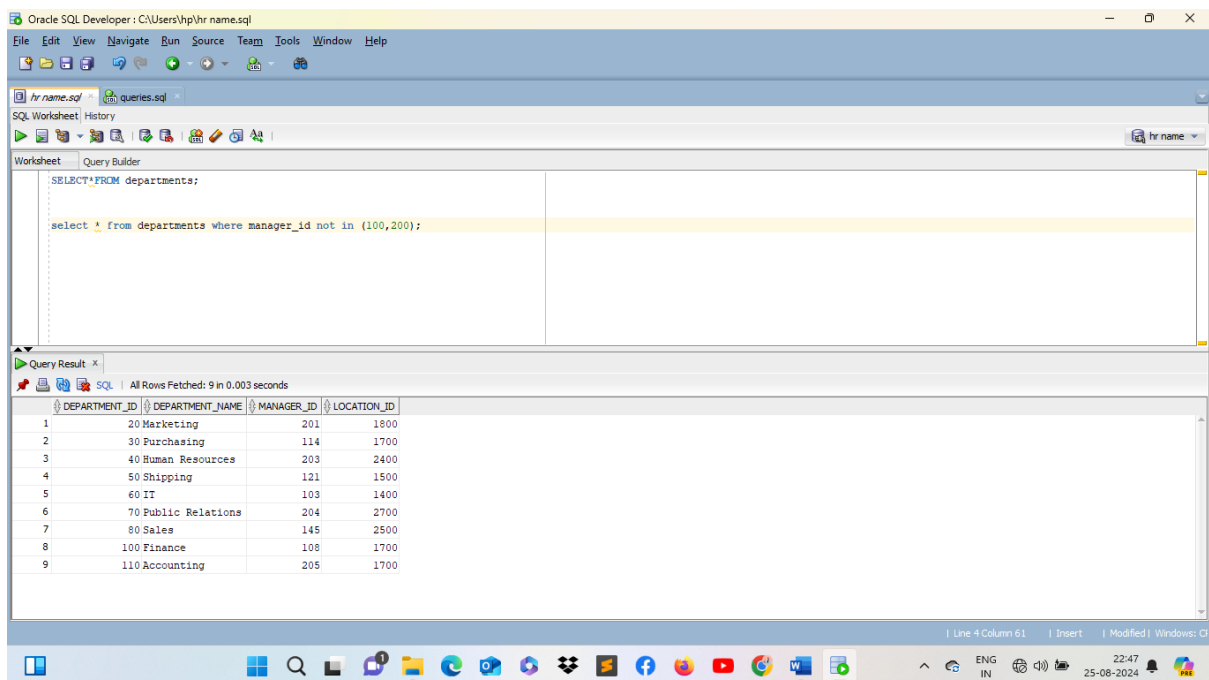
IN and NOT IN operator:

- IN operator is used to when you want to get more than one extract value then we need to use in operator.
- Syntax: select * from table_name where id in (10,20);
- This output will be whose having id is 10 and 20



NOT IN operator:

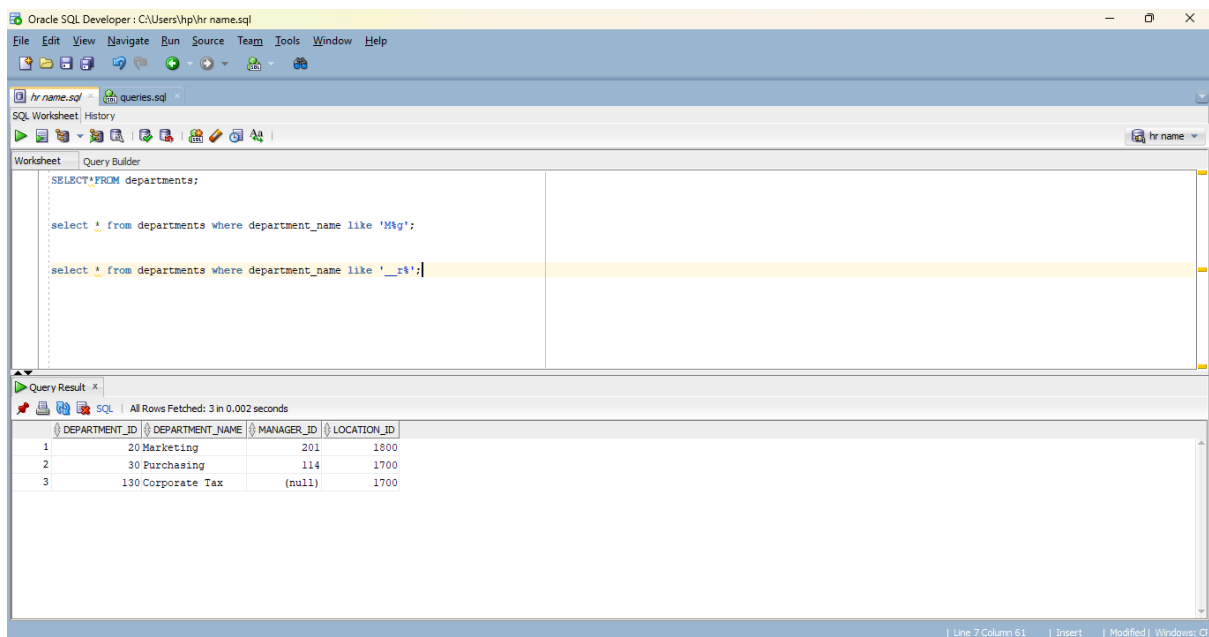
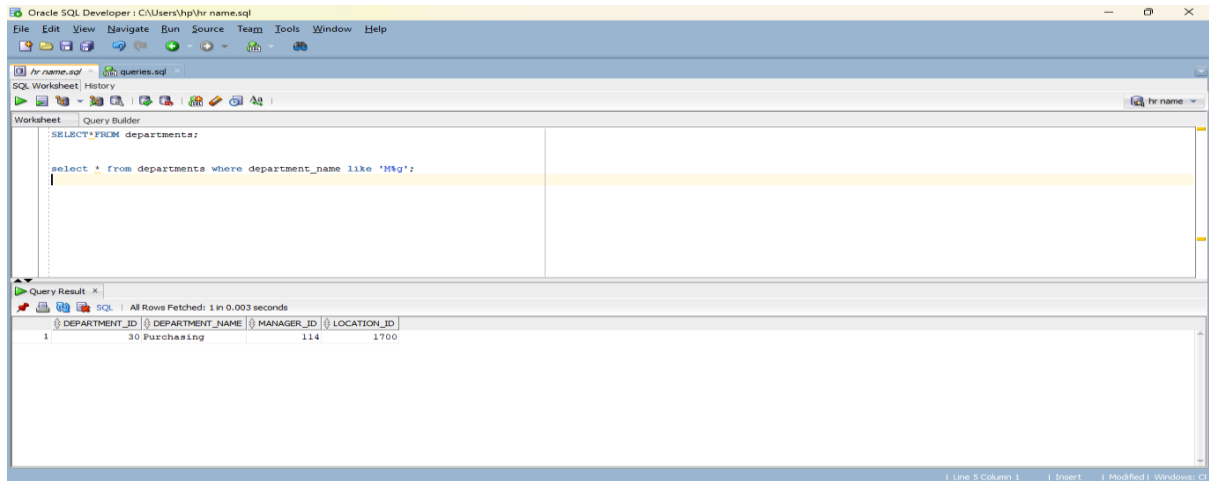
- Syntax: select * from table_name where id not in (10,20);
- It will return all records that are not any of the values in this list.



LIKE operator:(%, _);

- When we don't know the exact values or data then we need to use like operator
- In like operator we have 2 methods
- % - It can be anything
- _ only one position. it could be character or number
- Syntax: select * from table_name where column_name like 'M%';

- If you know first letter and last letter then the syntax would be
- Syntax: select * from table_name where column_name like 'M%g';
- If you know only the middle letter then we need to use '_e';



Logical operator:

- When we want to verify one or more condition in where clause then we need to use logical operator
- In logical operator contains of 3 types
- AND: Combines two or more conditions and returns true only if all conditions are true.

- OR: Combines two or more conditions and returns true if at least one condition is true.
- NOT: Reverses the result of a condition, returning true if the condition is false, and vice versa.

Oracle SQL Developer: C:\Users\hp\hr name.sql

File Edit View Navigate Run Source Team Tools Window Help

hr name.sql queries.sql

SQL Worksheet History

Worksheet Query Builder

```

SELECT * FROM departments;

--AND--

select * from departments where department_id > 50 and location_id <= 1800 and location_id != 1400;

--OR--

--NOT--

```

Query Result: All Rows Fetched: 19 in 0.002 seconds

	DEPARTMENT_ID	DEPARTMENT_NAME	MANAGER_ID	LOCATION_ID
1	90	Executive	100	1700
2	100	Finance	108	1700
3	110	Accounting	205	1700
4	120	Treasury	(null)	1700
5	130	Corporate Tax	(null)	1700
6	140	Control And Credit	(null)	1700
7	150	Shareholder Services	(null)	1700
8	160	Benefits	(null)	1700
9	170	Manufacturing	(null)	1700
10	180	Construction	(null)	1700

36°C Partly sunny

Oracle SQL Developer: C:\Users\hp\hr name.sql

File Edit View Navigate Run Source Team Tools Window Help

hr name.sql queries.sql

SQL Worksheet History

Worksheet Query Builder

```

SELECT * FROM departments;

--OR--

select * from departments where department_id < 50 or location_id != 10;

```

Query Result: All Rows Fetched: 27 in 0.003 seconds

	DEPARTMENT_ID	DEPARTMENT_NAME	MANAGER_ID	LOCATION_ID
1	10	Administration	200	1700
2	20	Marketing	201	1800
3	30	Purchasing	114	1700
4	40	Human Resources	203	2400
5	50	Shipping	121	1500
6	60	IT	103	1400
7	70	Public Relations	204	2700
8	80	Sales	145	2500
9	90	Executive	100	1700
10	100	Finance	108	1700

Oracle SQL Developer: C:\Users\hp\hr name.sql

File Edit View Navigate Run Source Team Tools Window Help

hr name.sql queries.sql

SQL Worksheet History

Worksheet Query Builder

```

SELECT * FROM departments;

--NOT--

select * from departments where location_id != 10;

```

Query Result: All Rows Fetched: 27 in 0.005 seconds

	DEPARTMENT_ID	DEPARTMENT_NAME	MANAGER_ID	LOCATION_ID
1	10	Administration	200	1700
2	20	Marketing	201	1800
3	30	Purchasing	114	1700
4	40	Human Resources	203	2400
5	50	Shipping	121	1500
6	60	IT	103	1400
7	70	Public Relations	204	2700
8	80	Sales	145	2500
9	90	Executive	100	1700
10	100	Finance	108	1700

Delimitres:

- When you want to use apostrophe then we need to use “q”
- q – it is keyword for delimiters
- Syntax : select q' [Happy coding]' from dual;

