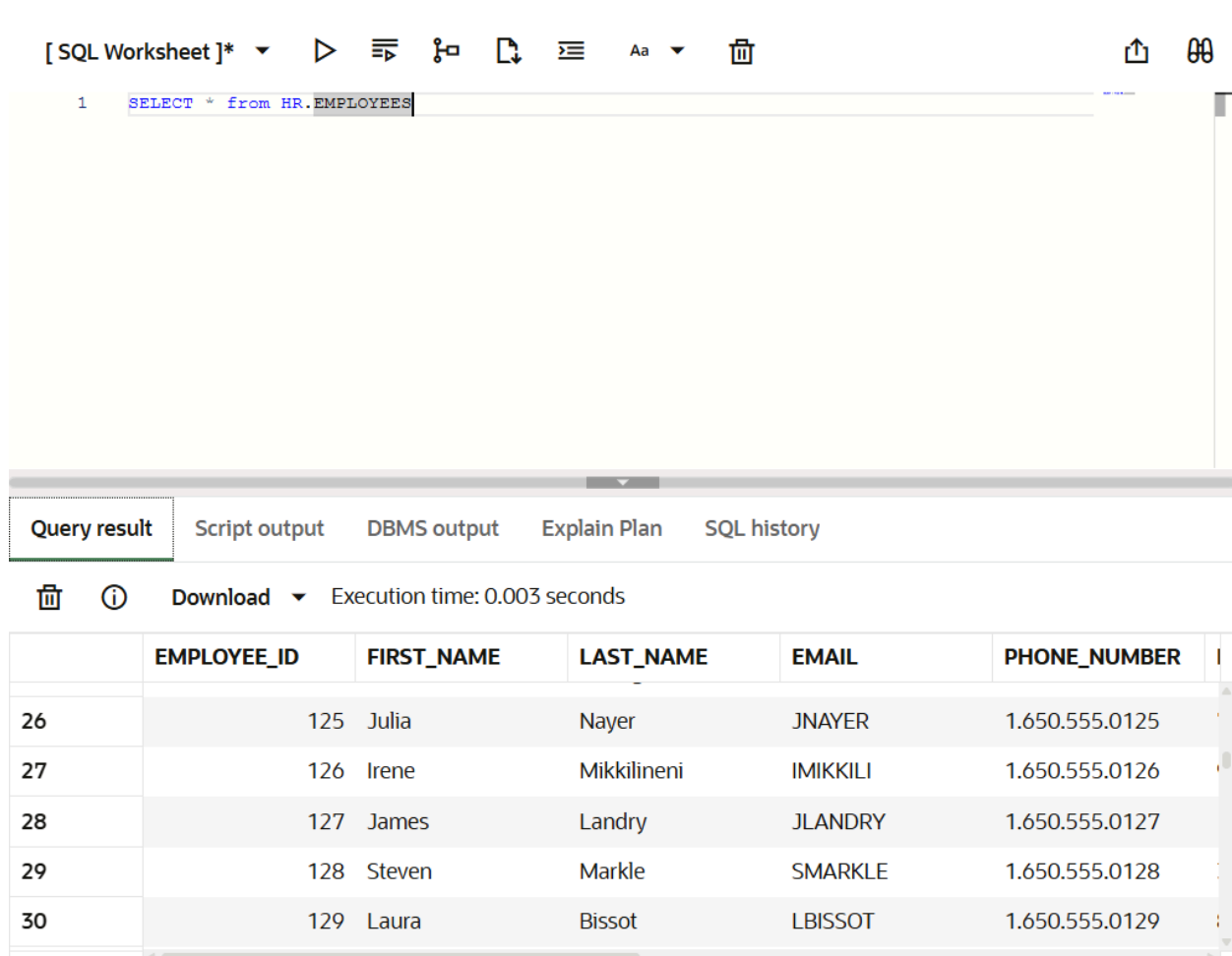


Oracle - Day1

1. SELECT * from HR.EMPLOYEES

This query retrieves all columns and all records from the EMPLOYEES table in the HR schema. The asterisk (*) selects every column like EMPLOYEE_ID, FIRST_NAME, LAST_NAME, EMAIL, etc. It is useful when you want to view complete data, explore the table, or see full details of each employee.



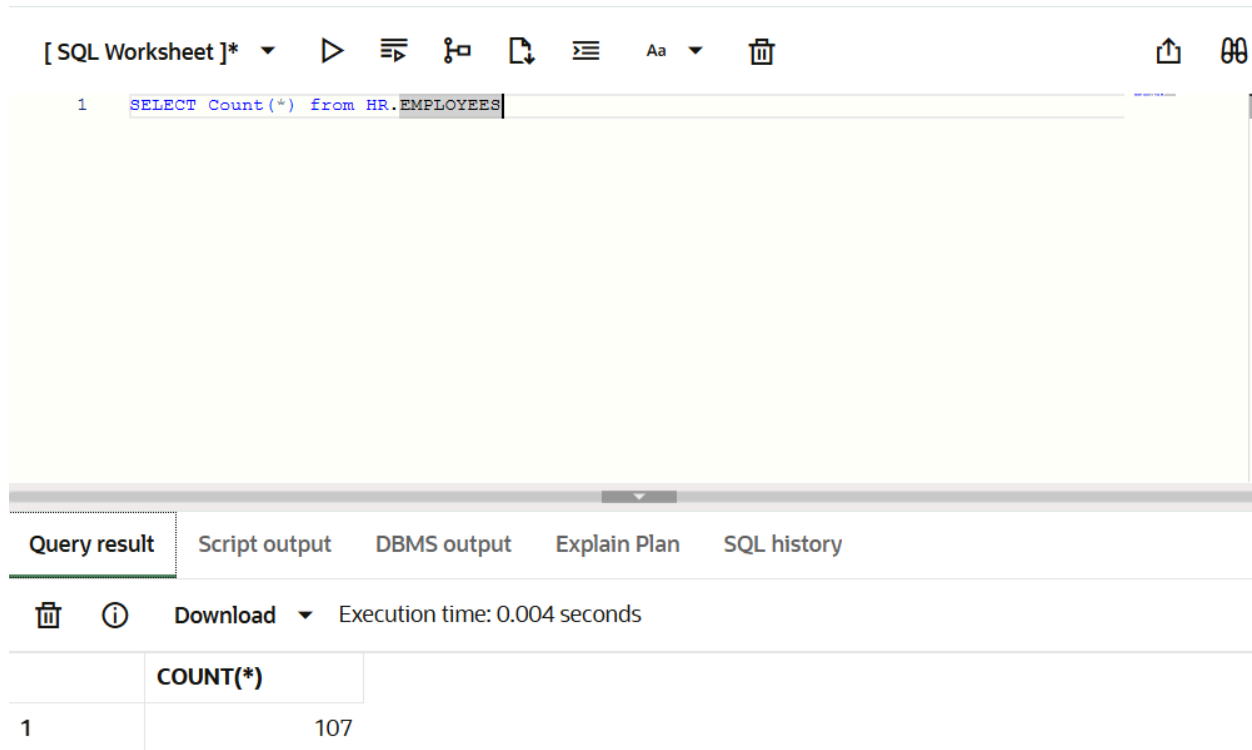
The screenshot displays an Oracle SQL Worksheet interface. At the top, a toolbar contains icons for running the query, saving, and other functions. Below the toolbar, the SQL editor shows the query: `1 SELECT * from HR.EMPLOYEES`. The bottom section of the interface is divided into tabs: "Query result" (selected), "Script output", "DBMS output", "Explain Plan", and "SQL history". Below the tabs, a status bar indicates "Execution time: 0.003 seconds". The main area shows the query results in a table with the following columns: EMPLOYEE_ID, FIRST_NAME, LAST_NAME, EMAIL, and PHONE_NUMBER. The results list five employees with their respective IDs, names, and phone numbers.

	EMPLOYEE_ID	FIRST_NAME	LAST_NAME	EMAIL	PHONE_NUMBER
26	125	Julia	Nayer	JNAYER	1.650.555.0125
27	126	Irene	Mikkilineni	IMIKKILI	1.650.555.0126
28	127	James	Landry	JLANDRY	1.650.555.0127
29	128	Steven	Markle	SMARKLE	1.650.555.0128
30	129	Laura	Bissot	LBISSOT	1.650.555.0129

2. SELECT Count(*) from HR.EMPLOYEES

This query counts the total number of employees in the HR.EMPLOYEES table using COUNT(*), including all rows even with NULL values.

It is useful for getting the total records or for summary reporting.



The screenshot shows an SQL worksheet interface. At the top, there is a toolbar with icons for running, saving, and other functions. Below the toolbar, the SQL query is entered in a text area: `1 SELECT Count(*) from HR.EMPLOYEES`. Below the query area, there are tabs for 'Query result', 'Script output', 'DBMS output', 'Explain Plan', and 'SQL history'. The 'Query result' tab is selected, showing a table with the query result. The table has two columns: 'COUNT(*)' and a value '107'. The execution time is displayed as '0.004 seconds'.

	COUNT(*)
1	107

3. select * from hr.EMPLOYEES where LAST_NAME in ('James','Yang')

This query fetches

all details of employees from the HR.EMPLOYEES table whose **last name is either 'James' or 'Yang'**

It uses the IN clause to match multiple values in the LAST_NAME column.

[SQL Worksheet]*
▶
≡
🔗
📄
≡
Aa
🗑️
📄
🔍

1
select * from hr.EMPLOYEES where LAST_NAME in ('James','Yang')

Query result
Script output
DBMS output
Explain Plan
SQL history

🗑️
🔍
Download
Execution time: 0.007 seconds

	EMPLOYEE_ID	FIRST_NAME	LAST_NAME	EMAIL	PHONE_NUMBER	H
1	103	Alexander	James	AJAMES	1.590.555.0103	1,
2	101	Neena	Yang	NYANG	1.515.555.0101	9,

4. SELECT DISTINCT(LAST_NAME) from HR.EMPLOYEES

This query retrieves all

unique last names of employees from the HR.EMPLOYEES table. The **DISTINCT** keyword removes duplicates, so each last name appears only once in the result. It is useful to see how many different last names are present among employees.









The screenshot shows an SQL Worksheet interface. At the top, there is a toolbar with icons for running queries, saving, and other functions. Below the toolbar, a text area contains the SQL query: `1 SELECT DISTINCT(LAST_NAME) from HR.EMPLOYEES`. Below the text area, there is a tabbed interface with five tabs: "Query result", "Script output", "DBMS output", "Explain Plan", and "SQL history". The "Query result" tab is selected. Below the tabs, there is a section with a trash icon, an information icon, a "Download" button, and the text "Execution time: 0.004 seconds". Below this, a table displays the results of the query. The table has two columns: a row number and the last name. The results are as follows:

	LAST_NAME
1	Abel
2	Ande
3	Atkinson
4	Baida
5	Banda
6	Bates

5. SELECT UNIQUE LAST_NAME FROM HR.EMPLOYEES



This query retrieves all

unique last names from the `HR.EMPLOYEES` table. The `UNIQUE` keyword ensures that duplicate last names are removed from the result. It helps in identifying the **distinct last names** present among employees.

[SQL Worksheet]*      Aa   

```
1 SELECT UNIQUE LAST_NAME FROM HR.EMPLOYEES
```

Query result Script output DBMS output Explain Plan SQL history

  Download Execution time: 0.004 seconds

	LAST_NAME
1	Abel
2	Ande
3	Atkinson
4	Baida
5	Banda
6	Bates

6. SELECT CONCAT(FIRST_NAME, ' ', LAST_NAME) AS FULL_NAME FROM HR.EMPLOYEES









This query combines the

`FIRST_NAME` and `LAST_NAME` columns from the `HR.EMPLOYEES` table into a single column named `FULL_NAME`, with a space in between.

The



`CONCAT` function is used to **join strings** together.

It is useful when you want to display full names of employees in one column.

[SQL Worksheet]*      Aa   

```
1 SELECT CONCAT(FIRST_NAME, ' ', LAST_NAME) AS FULL_NAME FROM HR.EMPLOYEES
```

Query result Script output DBMS output Explain Plan SQL history

  Download Execution time: 0.005 seconds

	FULL_NAME
1	Ellen Abel
2	Sundar Ande
3	Mozhe Atkinson
4	Shelli Baida
5	Amit Banda
6	Elizabeth Bates

7. SELECT CONCAT(FIRST_NAME, ' ', LAST_NAME) AS FULL_NAME FROM HR.EMPLOYEES WHERE DEPARTMENT_ID = 90

This query retrieves the

full names of employees from the `HR.EMPLOYEES` table who belong to **department 90**. The `CONCAT` function joins `FIRST_NAME` and `LAST_NAME` with a space in between, and the `WHERE` clause filters the results based on the `DEPARTMENT_ID`.

The screenshot shows an SQL Worksheet interface. At the top, there is a toolbar with icons for running queries, saving, and other functions. Below the toolbar, a SQL query is entered in a text area:









```
1 SELECT CONCAT (FIRST_NAME, ' ', LAST_NAME) AS FULL_NAME FROM HR.EMPLOYEES WHERE DEPARTMENT
```

Below the query editor, there is a tabbed interface with the following tabs: "Query result", "Script output", "DBMS output", "Explain Plan", and "SQL history". The "Query result" tab is currently selected. Below the tabs, there is a section for the query results. It includes a "Download" button, an information icon, and a status bar indicating "Execution time: 0.001 seconds". The results are displayed in a table with two columns: "FULL_NAME" and an implicit row index column.

	FULL_NAME
1	Steven King
2	Neena Yang
3	Lex Garcia



8. select * from HR.EMPLOYEES where FIRST_NAME like ('Ale%')

This query retrieves all columns for employees from the `HR.EMPLOYEES` table whose **first name starts with 'Ale'**. The `LIKE ('Ale%')` condition matches any name beginning with "Ale", such as **Alexander**, **Alec**, etc. It's useful for filtering names based on patterns.

[SQL Worksheet]*      Aa   

```
1 select * from HR.EMPLOYEES where FIRST_NAME like ('Ale%')
```

Query result Script output DBMS output Explain Plan SQL history










  Download Execution time: 0.006 seconds

	EMPLOYEE_ID	FIRST_NAME	LAST_NAME	EMAIL	PHONE_NUMBER	H
1	185	Alexis	Bull	ABULL	1.650.555.0150	2,
2	103	Alexander	James	AJAMES	1.590.555.0103	1,
3	115	Alexander	Khoo	AKHOO	1.515.555.0115	5,

9. SELECT * FROM HR.EMPLOYEES WHERE FIRST_NAME = 'Alexander' or
LAST_NAME = 'Yang'



This query retrieves all details of the employee from the `HR.EMPLOYEES` table whose **first name is 'Alexander' and last name is 'Yang'**.

It uses the `AND` operator to ensure that **both conditions must be true** for a row to be included in the result.

[SQL Worksheet]*      Aa    

```
1 SELECT * FROM HR.EMPLOYEES WHERE FIRST_NAME = 'Alexander' and LAST_NAME = 'Yang'
```

Query result Script output DBMS output Explain Plan SQL history

  Download Execution time: 0.007 seconds

	EMPLOYEE_ID	FIRST_NAME	LAST_NAME	EMAIL	PHONE_NUMBER	H
1	101	Neena	Yang	NYANG	1.515.555.0101	9,
2	103	Alexander	James	AJAMES	1.590.555.0103	1,
3	115	Alexander	Khoo	AKHOO	1.515.555.0115	5,

10. SELECT * FROM HR.EMPLOYEES WHERE FIRST_NAME = 'Alexander' and
LAST_NAME = 'Yang'

This query retrieves all columns from the HR.EMPLOYEES table for the employee whose **first name is 'Alexander'** and **last name is 'Yang'**

The AND operator ensures that **both conditions** must be true for a record to be selected.

[SQL Worksheet]*

Aa

1SELECT * FROM HR.EMPLOYEES WHERE FIRST_NAME = 'Alexander' and LAST_NAME = 'Yang'

Query resultScript outputDBMS outputExplain PlanSQL history

DownloadExecution time: 0.007 seconds




EMPLOYEE_ID	FIRST_NAME	LAST_NAME	EMAIL	PHONE_NUMBER	HIRE_DATE
No items to display.					

11. SELECT FIRST_NAME AS NAME FROM HR.EMPLOYEES

This query selects the FIRST_NAME column from the HR.EMPLOYEES table and renames it as NAME in the output using the AS keyword. AS is used to give a column a temporary or more readable name.

```
1 SELECT FIRST_NAME AS NAME FROM HR.EMPLOYEES
```

Query result Script output DBMS output Explain Plan SQL history

  Download  Execution time: 0.004 seconds

	NAME
1	Ellen
2	Sundar
3	Mozhe
4	Shelli
5	Amit
6	Elizabeth