SQL

DDL

DML

DCL

TCL

DQL

SELECT query

SELECT query is used to retrieve record(s) from table(s).

syntax:

SELECT <<expressions>>|columns

FROM <<table1>>, <<table2>> | <<table1>> JOIN <<table2>>

[WHERE <<condition1>> AND | OR <<condition2>>]

[GROUP BY <<expression>>]

[HAVING <<condition>>]

[ORDER BY <<column1>>,<<column2>>]

To learn by example, we need some sample schema.

schema (in mysql) is a database.

schema (in oracle) is a private space of an user.

ex: hr schema

sys schema

in livesql

SELECT \* FROM HR.EMPLOYEES;

Basic select query:

SELECT \* FROM HR.EMPLOYEES;

SELECT employee\_id, first\_name FROM HR.EMPLOYEES;

in this above query, what is \* means?

all columns.

\* or column list we mention in that place decides the PROJECTION

What is PROJECTION?

a table may have many columns. But we project only relevant columns.

What is SELECTION?

a table may have many rows, But we select only the rows that pass a criteria.

SELECT \* FROM HR.EMPLOYEES where SALARY>15000;

We got 3 rows only.

UNION

SELECT \* FROM HR.EMPLOYEES where SALARY>15000

union

SELECT \* FROM HR.EMPLOYEES where SALARY>10000 AND SALARY<=15000;

UNION vs UNION ALL

UNION ALL

includes duplicate rows (from both queries)

in that case the number of rows is always equal to sum of both query rows.

INTERSECT

SELECT \* FROM HR.EMPLOYEES where SALARY>15000

intersect

SELECT \* FROM HR.EMPLOYEES where SALARY>10000

we got 3 rows

PRODUCES THE COMMON ROWS amont the 2 select statements

UNION

UNION ALL

MINUS

PRODUCT

MINUS

SELECT \* FROM HR.EMPLOYEES where SALARY>10000

minus

SELECT \* FROM HR.EMPLOYEES where SALARY>15000;

PRODUCT

is done when we use JOINs

Reminder: JOINS

-----------------------------------

SELECT

column list

alias

SELECT EMPLOYEE\_ID, FIRST\_NAME, LAST\_NAME, SALARY

FROM HR.EMPLOYEES --projection

WHERE SALARY>10000; --selection

WHERE:

you can give conditions

multiple conditions using AND OR operators

SELECT EMPLOYEE\_ID, FIRST\_NAME, LAST\_NAME, SALARY

FROM HR.EMPLOYEES

WHERE SALARY>10000

AND SALARY<=12000;

Task:

write select query on HR.EMPLOYEES and display

different set of columns,

and conditions to filter the rows based on different columns like SALARY, DEPARTMENT\_NO etc.

Observation:

When the WHERE condition is based on (=)primary key column (here it is EMPLOYEE\_ID), then we get 1 row. Example: SELECT \* FROM HR.EMPLOYEES WHERE EMPLOYEE\_ID=102

Try to get 1 result using conditions on non-primary key column. (use AND OR operators on multiple columns).

|| for concatenation

SELECT FIRST\_NAME || LAST\_NAME

In WHERE Condition

What are all operators we can use?

COMPARISON OPERATORS

> < >= <= = <> !=

We can modify these operators using ANY ALL (we will use during sub queries).

LOGICAL OPERATORS

NOT AND OR

BETWEEN NOT BETWEEN

IN NOT IN

IS NULL

IS NOT NULL

=NULL will not work

SELECT \* FROM HR.EMPLOYEES WHERE DEPARTMENT\_ID IS NULL;

LIKE operator

Is used to use wild cards. Ex: LIKE ‘A%’

LIKE ‘A\_’

2 wild card characters are used in LIKE operator.

% means 0 or more characters

\_ means only 1 1 and only 1

Task

Find all the employees, whose first name ends with e

SELECT \* FROM HR.EMPLOYEES WHERE FIRST\_NAME LIKE '%e'

Now we have listed the operators, we need to observe them.

Task:

Find the employees where salary is in the range of 5000 to 10000 (inclusive) without using >= <= = etc comparison operator.

Steps for PostgreSQL

1. In software centre, search for “PostgreSQL”
2. Install

After installation:

1. In start menu, type psql and enter

All default values, just press enter

Problem:

Psql is not found in start menu

Solution:

Go to

C:\Program Files (x86)\PostgreSQL\10\bin

In address bar click on empty space and type cmd and enter

You have come to command prompt at the bin location

Type the following command:

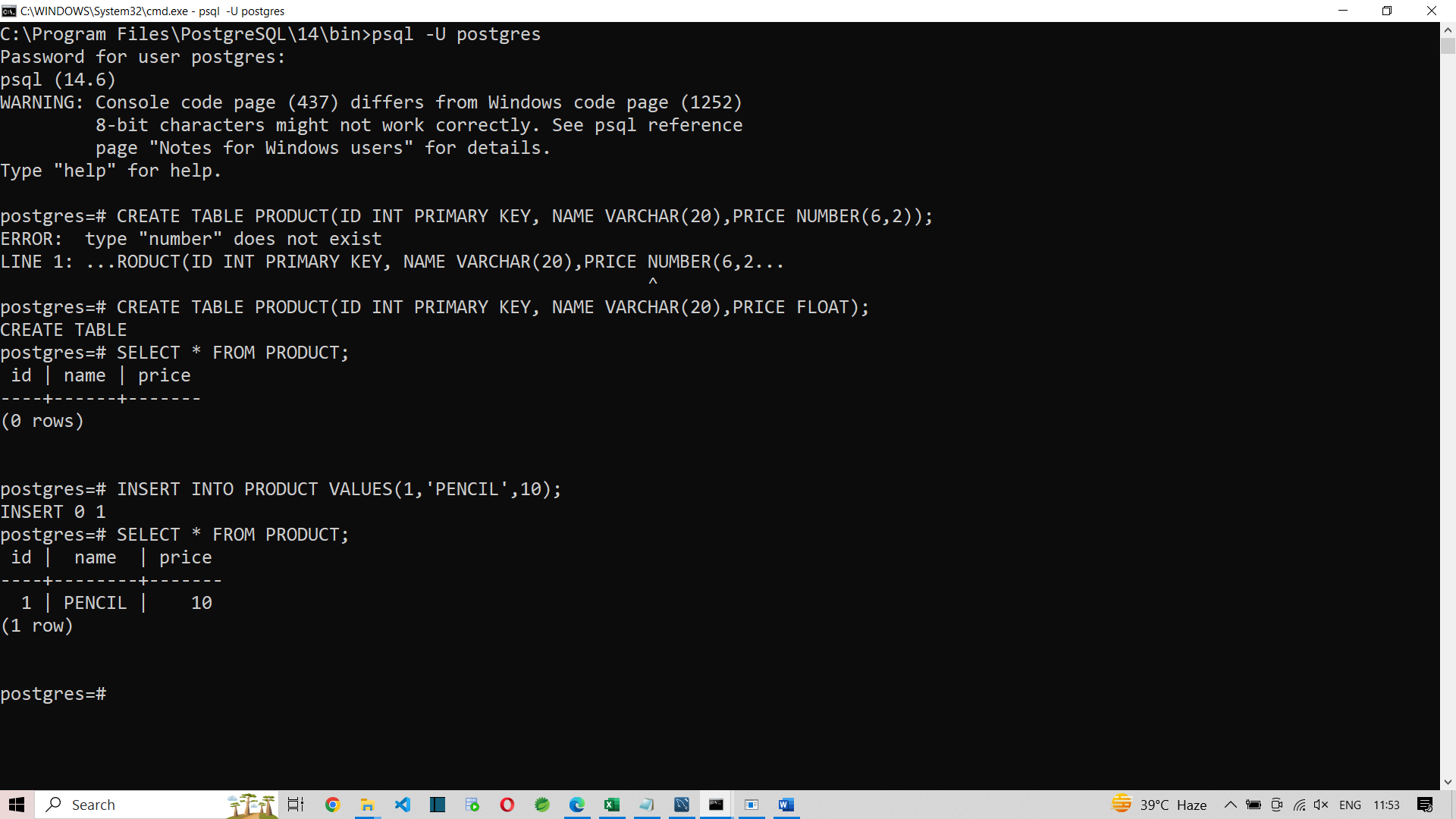
psql -U postgres

<<it will prompt the password of the user “postgres”>>

Type the password as “postgres”. It will not be visible. But type correctly

Now, you can see

postgres=#



Lets continue to learn SELECT statement.

SELECT

WHERE

Lets learn ORDER BY

The result will be sorted based on ORDER BY expression.

SELECT \* FROM HR.EMPLOYEES

ORDER BY FIRST\_NAME DESC;

The above query returns the records in the order of FIRST\_NAME ie in descending order.

ASC means Ascending order. If we do not mention ASC also it is ascending order.

SELECT \* FROM HR.EMPLOYEES

ORDER BY salary;

SELECT \* FROM HR.EMPLOYEES

ORDER BY hire\_date;

So we can sort based on numeric, char, date type of columns.

We can sort records based on multiple columns.

SELECT \* FROM HR.EMPLOYEES

ORDER BY DEPARTMENT\_ID, SALARY;

The above query sorts the employee records based on department\_id and if the department\_ids are same for employees, their records are sorted by their salary.

SELECT \* FROM HR.EMPLOYEES

ORDER BY DEPARTMENT\_ID desc, SALARY asc;

We can also use expressions in ORDER BY clause:

SELECT \* FROM HR.EMPLOYEES

ORDER BY EXTRACT(DAY FROM HIRE\_DATE);

Task:

Practise the ORDER BY class

ASC

DESC

Multiple columns

Expressions

Functions in Oracle