================= CHAP2 . RETRIVING DATA USING SQL SELECT STATEMENTS ==========

--WHAT is in this chap

--1.simple select statements

--2.selecting specific columns

--3.arithmatic operators

--4.NULL values

--5.Column ALIAS(rename col name)

--6.Concatination Operator ||

--7.Alternative Quote Operatore (q)

--8.DISTINCT clause

--9.DESCRIBE command

== 1.BASIC SELECT STATEMNETS ==

--select * from departments;

== 2.SELECTING SPECIFIC COLUMNS ==

--select department_id, location_id from departments;

=== 3.ARITHMATIC OPERATIONS IN SQL STATEMENTS ===

1.+

2.-

3.*

4./

1.-- using add (+)

select last_name , salary , salary +300 **--(if salry is 20,000 it increase to 20,300)**

from employees;

select last_name , salary , 12*salary +100

from employees; **--ex(salry increases to 12*6000+100)**

2.-- using substract (-)

select last_name , salary , salary -300 **--(if salry is 20,000 it decrease to 19,700)**

from employees;

-- AS SAME WE CAN DO MULTIPLY AND DIVIDE OPERATIONS

--ex

SELECT 3+10 FROM DUAL;--dummy table--

SELECT 30+10+90 FROM DUAL;

select JOB_TITLE,MIN_SALARY,MIN_SALARY-10000,MAX_SALARY,MAX_SALARY-MIN_SALARY from jobs;

select MIN_SALARY,2/(MIN_SALARY+5000) from jobs;

select MIN_SALARY,2/MIN_SALARY+5000 from jobs;

=== 4.NULL VALUE ====

--a values that is unavailabel,unsigned or unknown

-- not same as zero or blank space

--here commission_pct is null it not having any values

select last_name , job_id, salary , commission_pct

from employees;

select last_name, 12 * salary * commission_pct

from employees;

==== 5.COLUMN ALIAS (RENAME THE COL. NAME) ====

--Rename column heading

--useful for calaculations

-- *optional (we use AS keyword for between both col name and alias col name)

-- if any space in col it requires double quotation

--[1]

```
select last_name AS Lname, commission_pct comm
from employees;
```

-- here we change (last_name) col name as Lname using AS keyword

-- & commission_pct col name chnaged but without as keyword

--[2]

```
select last_name "L name" , salary*12 "Annual Salary"
from employees;
```

-- here we space is given so our col name having also contain space

--but it required in double quotation

```
SELECT 30+10+90 Addition FROM DUAL;
SELECT 30+10+90 "numbers Addition" FROM DUAL;
SELECT 30+10+90 as Addition FROM DUAL;
select last_name name,SALARY money from employees;
select JOB_TITLE "worker",MAX_SALARY+MIN_SALARY as "annual salary" from jobs;
```

==== 6.CONCATINATION OPERATOR =====

--|| (Two Vertical Bar) is used to concatination

-- it links col or char string to other columns

-- create resultant column that is character expression

```
select job_id || last_name AS "employees1"
from employees;
```

-- here we concat two col in one col.

=Literal Char Strings=

-- literal is char , number or date include in select statemnets

-- must be inclosed in single quotation marks

-- each string is output once fro each row returned

--using literal character string :external character or expression inserted in query--

```
select 'my name is ' ||first_name||' '||last_name ||' joined as
'||JOB_ID from employees;
```

```
select last_name || ' is a ' || job_id  AS "employee2"
from employees;
```

-- here we can see ' is a ' placed in both col name

--ex

select FIRST_NAME,LAST_NAME,FIRST_NAME||' '||LAST_NAME as fullname from employees;

select FIRST_NAME as emp,last_name||EMPLOYEE_ID userid from employees;

=== 7.ALTERNATIVE QUOTE OPERATOR (q) ====

-- it specify your own quotaion mark delimeter

-- select any delimeter

-- inncreasee readability and usability

select department_name || q'[Departments manager ID:]' || manager_id AS "DPM"

from departments;

select last_name || q'[is a]' || job_id AS LJB

from employees;

-- here we use q'[]' for specific msg that we want to add in between both col name

select q'[My name is]' ||first_name,last_name from employees;

select q'[My name is]'||first_name ||q'[joined as]'||JOB_ID from employees;

=== 8.DISTINCT ===

-- it avoid duplicate rows

select department_id from employees;

-- it having allows duplicate rows

-- but when we use distinct it avoid duplicate rows

select DISTINCT department_id from employees;

=== 9.DESCRIBE COMMAND ====

--used to display structure of table

-- it describe details like

--Name

--Null or NOT?

--Type

DESCRIBE employees;

describe DEPARTMENTS;

describe jobs;

describe EMPLOYEES;