



Introduction to SQL

UNIVERSITY OF TEHRAN

NOVEMBER, 2016

What is SQL?

- ▶ **Structured Query Language**
- ▶ SQL is Structured Query Language, which is a computer language for **storing, manipulating** and **retrieving** data stored in relational database.
- ▶ SQL is the standard language for **Relation** Database System. All relational database management systems like "MySQL, MS Access, Oracle, Sybase, Informix, postgres and SQL Server" use SQL as standard database language.

SQL Commands

- ▶ **DDL** - Data Definition Language
- ▶ **DML** - Data Manipulation Language
- ▶ **DCL** - Data Control Language
- ▶ **DQL** - Data Query Language

SQL Commands

DDL - Data Definition Language:

Command	Description
CREATE	Creates a new table, a view of a table, or other object in database
ALTER	Modifies an existing database object, such as a table.
DROP	Deletes an entire table, a view of a table or other object in the database.

DML - Data Manipulation Language:

Command	Description
INSERT	Creates a record
UPDATE	Modifies records
DELETE	Deletes records

SQL Commands

DCL - Data Control Language:

Command	Description
GRANT	Gives a privilege to user
REVOKE	Takes back privileges granted from user

DQL - Data Query Language:

Command	Description
SELECT	Retrieves certain records from one or more tables

SQL RDBMS Concepts

► TABLE

► RECORD

► COLUMN

► CELL

► NULL?

ID	NAME	AGE	ADDRESS	SALARY	ADDRESS
1	Ramesh	32	Ahmedabad	2000.00	Ahmedabad
2	Khilan	25	Delhi	1500.00	Delhi
3	kaushik	23	Kota	2000.00	Kota
4	Chaitali	25	Mumbai	6500.00	Mumbai
5	Hardik	27	Bhopal	8500.00	Bhopal
6	Komal	22	MP	4500.00	MP
7	Muffy	24	Indore	10000.00	Indore

1	Ramesh	32	Ahmedabad	2000.00
---	--------	----	-----------	---------

SQL RDBMS Concepts

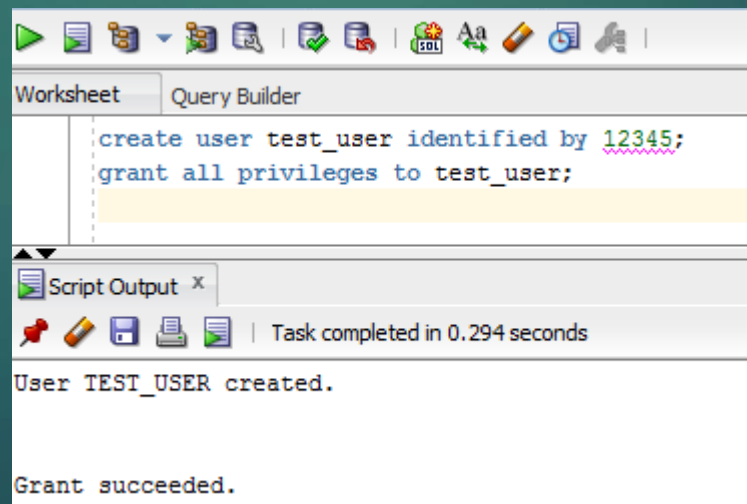
- ▶ SQL Constraints: (applied on columns)
 - ▶ NOT NULL Constraint
 - ▶ UNIQUE Constraint
 - ▶ PRIMARY Key
 - ▶ FOREIGN Key
 - ▶ CHECK Constraint
- ▶ Data Integrity:
 - ▶ Entity Integrity: There are no duplicate rows in a table
 - ▶ Domain Integrity: Enforces valid entries for a given column by restricting the type
 - ▶ Referential Integrity: Rows cannot be deleted which are used by other records
 - ▶ User-Defined Integrity: Enforces some specific business rules

SQL RDBMS Concepts

- ▶ Data Types:
- ▶ Character datatypes:
 - ▶ CHAR
 - ▶ NCHAR
 - ▶ NVARCHAR2
 - ▶ VARCHAR2
- ▶ *Numeric* datatypes:
 - ▶ NUMBER
 - ▶ BINARY_FLOAT
 - ▶ BINARY_DOUBLE
- ▶ *Date time* datatype:
 - ▶ DATE

DCL: CREATE USER

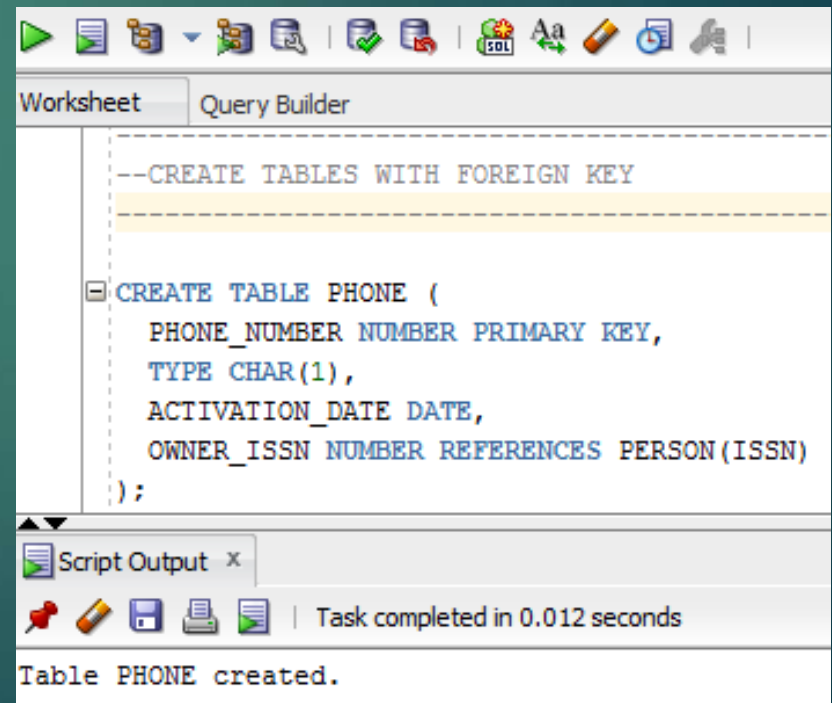
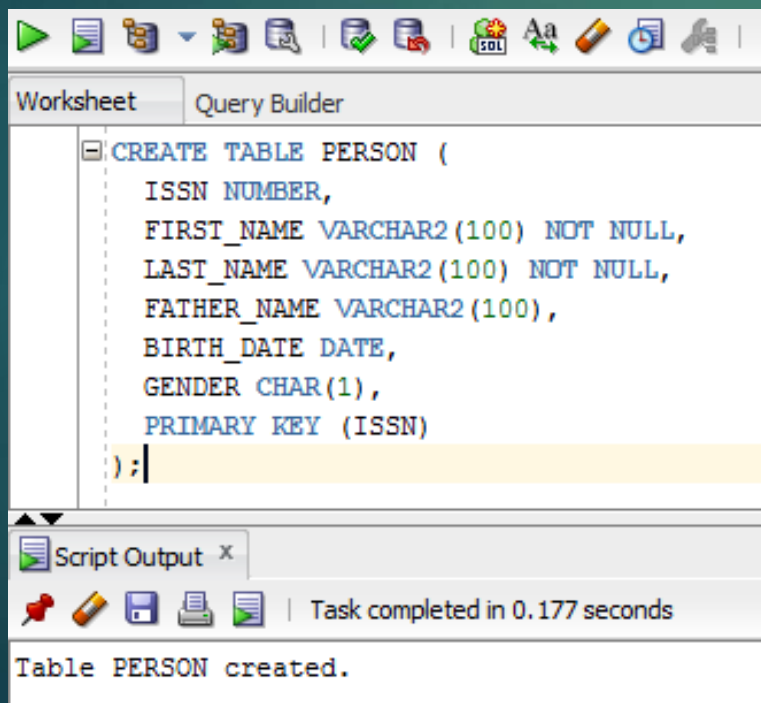
- ▶ 1) Define User with Username/Password
- ▶ 2) Grants Sufficient Privileges.
- ▶ Using SQL Developer (UI, Command)
- ▶ Using SQL *PLUS (Command)



SQL : DDL

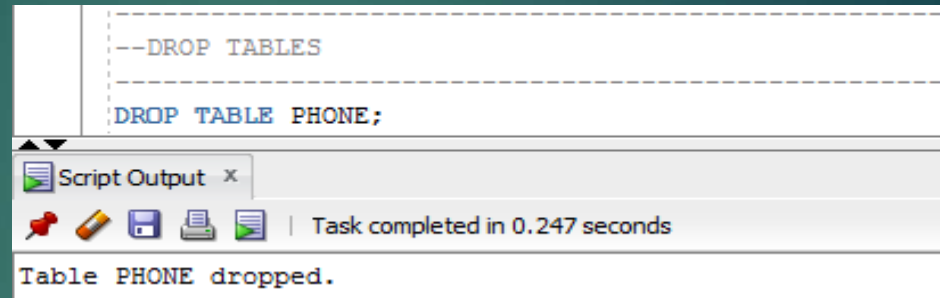
- ▶ Create Table:
 - ▶ COLUMNS
 - ▶ CONSTRAINTS

```
CREATE TABLE table_name(  
    column1 datatype,  
    column2 datatype,  
    column3 datatype,  
    .....  
    columnN datatype,  
    PRIMARY KEY ( one or more columns )  
);
```



SQL: DDL

► DROP TABLE:



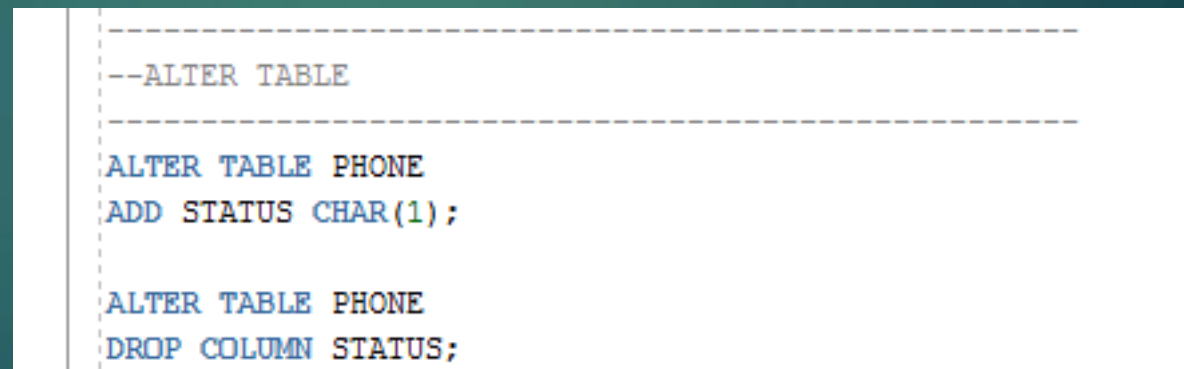
```
--DROP TABLES  
  
DROP TABLE PHONE;
```

Script Output x

Task completed in 0.247 seconds

Table PHONE dropped.

► ALTER TABLE



```
--ALTER TABLE  
  
ALTER TABLE PHONE  
ADD STATUS CHAR(1);  
  
ALTER TABLE PHONE  
DROP COLUMN STATUS;
```

► OTHER DDL COMMANDS!!!!

SQL: DML: INSERT

- ▶ INSERT INTO:
 - ▶ NUMBER, CHAR/VARCHAR2, DATE?

```
INSERT INTO PERSON VALUES (1, 'HOSSEIN', 'ASKARIAN', 'ALI', TO_DATE('12-JUN-1990'), 'M');
INSERT INTO PERSON VALUES (2, 'SADEGH', 'KARIMI', 'HOSSEIN', TO_DATE('05/04/1990', 'DD/MM/YYYY'), 'M');
INSERT INTO PERSON VALUES (3, 'HOSSEIN', 'ASKARIAN', 'ALI', TO_DATE('10-04-1367', 'DD-MM-YYYY', 'NLS_CALENDAR=PERSIAN'), 'F');
```

Script Output x

Task completed in 0.055 seconds

1 row inserted.

1 row inserted.

```
-- WRONG: PK?
INSERT INTO PERSON VALUES (1, 'EHSAN', 'HAMZEI', 'ALIREZA', TO_DATE('26-JUN-1991'), 'M');
```

Script Output x Query Result x

Task completed in 0.096 seconds

Error starting at line : 51 in command -
INSERT INTO PERSON VALUES (1, 'EHSAN', 'HAMZEI', 'ALIREZA', TO_DATE('26-JUN-1991'), 'M')

Error report -
SQL Error: ORA-00001: unique constraint (TEST_USER.SYS_C0010353) violated
00001. 00000 - "unique constraint (%s.%s) violated"

*Cause: An UPDATE or INSERT statement attempted to insert a duplicate key.
For Trusted Oracle configured in DBMS MAC mode, you may see
this message if a duplicate entry exists at a different level.

*Action: Either remove the unique restriction or do not insert the key.

SQL: DML: INSERT

- ▶ INSERT INTO:
- ▶ SPECIFIC COLUMNS

```
-- SPECIFIC COLUMNS
INSERT INTO PERSON (BIRTH_DATE, FATHER_NAME, LAST_NAME, FIRST_NAME, ISSN)
VALUES (TO_DATE('12021990', 'DDMMYYYY'), 'HOSSEIN', 'HEMMATI', 'ALIREZA', 4);
```

```
-- WRONG: NOT NULL?
INSERT INTO PERSON (BIRTH_DATE, FATHER_NAME, FIRST_NAME, ISSN)
VALUES (TO_DATE('12021990', 'DDMMYYYY'), 'SADEGH', 'SETAREH', 5);
```

Script Output x Query Result x

Task completed in 0.011 seconds

Error starting at line : 57 in command -
INSERT INTO PERSON (BIRTH_DATE, FATHER_NAME, FIRST_NAME, ISSN)
VALUES (TO_DATE('12021990', 'DDMMYYYY'), 'SADEGH', 'SETAREH', 5)
Error report -
SQL Error: ORA-01400: cannot insert NULL into ("TEST_USER"."PERSON"."LAST_NAME")
01400. 00000 - "cannot insert NULL into (%s)"
*Cause: An attempt was made to insert NULL into previously listed objects.
*Action: These objects cannot accept NULL values.

SQL: DML: INSERT

- ▶ INSERT INTO:
- ▶ FOREIGN KEY

```
-- FOREIGN KEY
INSERT INTO PHONE VALUES (09324545321,'M',TO_DATE('01-01-1395','DD-MM-YYYY','NLS_CALENDAR = PERSIAN'),1);
-- WRONG: FK?
INSERT INTO PHONE VALUES (02134324533,'T',TO_DATE('03-05-1392','DD-MM-YYYY','NLS_CALENDAR = PERSIAN'),11)
```

Script Output x Query Result x

Task completed in 0.03 seconds

1 row inserted.

Error starting at line : 64 in command -

```
INSERT INTO PHONE VALUES (02134324533,'T',TO_DATE('03-05-1392','DD-MM-YYYY','NLS_CALENDAR = PERSIAN'),11)
```

Error report -

SQL Error: ORA-02291: integrity constraint (TEST_USER.SYS_C0010357) violated - parent key not found

02291. 00000 - "integrity constraint (%s.%s) violated - parent key not found"

*Cause: A foreign key value has no matching primary key value.

*Action: Delete the foreign key or add a matching primary key.

- ▶ ALL DML COMMANDS NEED COMMIT

SQL: DML: UPDATE

► UPDATE

```
UPDATE PERSON
SET FATHER_NAME = 'AAAAAA';
-- ROLLBACK!
ROLLBACK;
```

Script Output x Query Result x

Task completed in 0.003 seconds

4 rows updated.

Rollback complete.

► WHERE CLAUSE

```
--USING WHERE CLAUSE
UPDATE PERSON
SET FATHER_NAME = 'ALIREZA'
WHERE ISSN = 1;
COMMIT;
```

Script Output x Query Result x

Task completed in 0.001 seconds

1 row updated.

Commit complete.

SQL: DML: DELETE

► DELETE (DELETE VS DROP????)

```
DELETE PERSON;
```

Script Output x Query Result x

Task completed in 0.01 seconds

Error starting at line : 85 in command -
DELETE PERSON

Error report -

SQL Error: ORA-02292: integrity constraint (TEST_USER.SYS_C0010357) violated - child record found
02292. 00000 - "integrity constraint (%s.%s) violated - child record found"

*Cause: attempted to delete a parent key value that had a foreign dependency.

*Action: delete dependencies first then parent or disable constraint.

```
DELETE PERSON WHERE ISSN = 2;  
ROLLBACK;
```

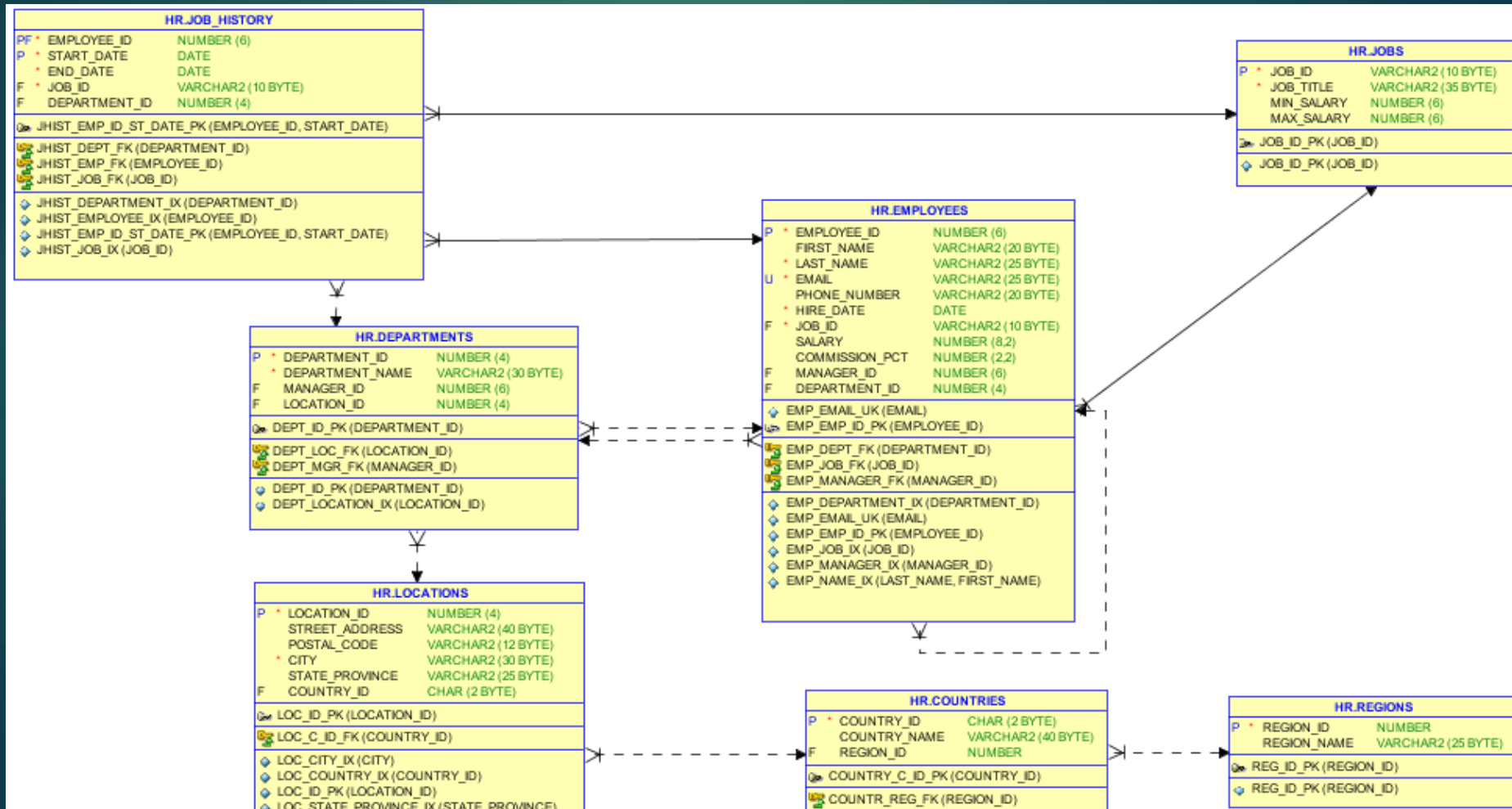
Script Output x Query Result x

Task completed in 0.001 seconds

1 row deleted.

Rollback complete.

HR-Schema

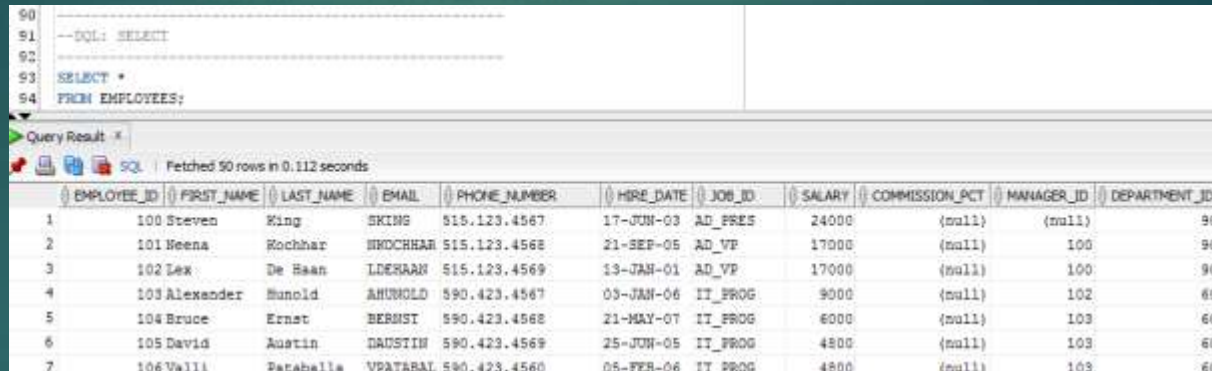


SQL: DQL: SELECT

▶ SELECT

```
SELECT column1, column2....columnN
FROM   table_name;
```

▶ *

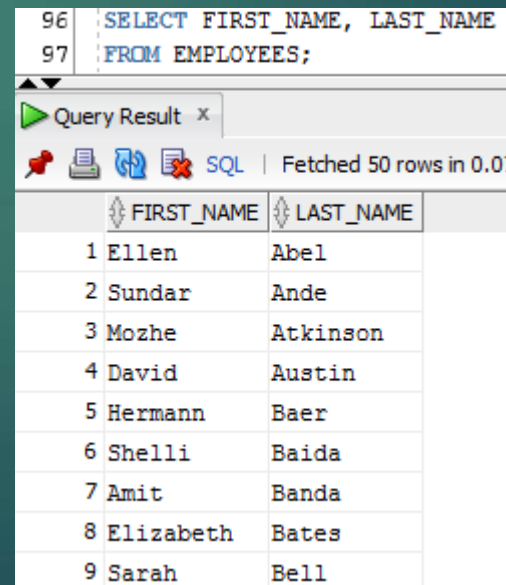


Query Result x

SQL | Fetched 50 rows in 0.112 seconds

	EMPLOYEE_ID	FIRST_NAME	LAST_NAME	EMAIL	PHONE_NUMBER	HIRE_DATE	JOB_ID	SALARY	COMMISSION_PCT	MANAGER_ID	DEPARTMENT_ID
1	100	Steven	King	SKING	515.123.4567	17-JUN-03	AD_PRES	24000	(null)	(null)	90
2	101	Neena	Kochhar	NKOCHHAR	515.123.4568	21-SEP-05	AD_VP	17000	(null)	100	90
3	102	Lex	De Haan	LDEHAAN	515.123.4569	13-JAN-01	AD_VP	17000	(null)	100	90
4	103	Alexander	Burns	ABURNS	590.423.4567	05-JAN-06	IT_PROG	9000	(null)	102	60
5	104	Bruce	Ernst	BERNST	590.423.4568	21-MAY-07	IT_PROG	6000	(null)	103	60
6	105	David	Austin	DAUSTIN	590.423.4569	25-JUN-05	IT_PROG	4800	(null)	103	60
7	106	Valli	Pataballa	VPATABAL	590.423.4560	05-FEB-06	IT_PROG	4800	(null)	103	60

▶ SPECIFIC COLUMNS



Query Result x

SQL | Fetched 50 rows in 0.07

	FIRST_NAME	LAST_NAME
1	Ellen	Abel
2	Sundar	Ade
3	Mozhe	Atkinson
4	David	Austin
5	Hermann	Baer
6	Shelli	Baida
7	Amit	Banda
8	Elizabeth	Bates
9	Sarah	Bell

SQL: DQL: SELECT

► DISTINCT CLAUSE

```
SELECT DISTINCT column1, column2....columnN  
FROM   table_name;
```

99	SELECT DEPARTMENT_ID
100	FROM EMPLOYEES

Query Result x	
SQL Fetched 50	
	DEPARTMENT_ID
1	90
2	90
3	90
4	60
5	60
6	60
7	60
8	60
9	100
10	100

99	SELECT DISTINCT DEPARTMENT_ID
100	FROM EMPLOYEES

Query Result x	
SQL All Rows Fetched: 12 in	
	DEPARTMENT_ID
1	100
2	30
3	(null)
4	90
5	20
6	70
7	110
8	50
9	80
10	40

SQL: DQL: SELECT

► WHERE CLAUSE

```
SELECT column1, column2....columnN
FROM   table_name
WHERE  CONDITION;
```

```
102 SELECT *
103 FROM EMPLOYEES
104 WHERE DEPARTMENT_ID = 100;
```

Query Result x

SQL | All Rows Fetched: 6 in 0.198 seconds

	EMPLOYEE_ID	FIRST_NAME	LAST_NAME	EMAIL	PHONE_NUMBER	HIRE_DATE	JOB_ID	SALARY	COMMISSION_PCT	MANAGER_ID	DEPARTMENT_ID
1	108	Nancy	Greenberg	NGREENBE	515.124.4569	17-AUG-02	FI_MGR	12008	(null)	101	100
2	109	Daniel	Faviet	DFAVIET	515.124.4169	16-AUG-02	FI_ACCOUNT	9000	(null)	108	100
3	110	John	Chen	JCHEN	515.124.4269	28-SEP-05	FI_ACCOUNT	8200	(null)	108	100
4	111	Ismael	Sciarra	ISCIARRA	515.124.4369	30-SEP-05	FI_ACCOUNT	7700	(null)	108	100
5	112	Jose Manuel	Urman	JMURMAN	515.124.4469	07-MAR-06	FI_ACCOUNT	7800	(null)	108	100
6	113	Luis	Popp	LPOPP	515.124.4567	07-DEC-07	FI_ACCOUNT	6900	(null)	108	100

```
SELECT column1, column2....columnN
FROM   table_name
WHERE  CONDITION-1 {AND|OR} CONDITION-2;
```

```
106 SELECT *
107 FROM EMPLOYEES
108 WHERE DEPARTMENT_ID = 100 AND FIRST_NAME LIKE 'J%';
```

Query Result x

SQL | All Rows Fetched: 2 in 0.004 seconds

	EMPLOYEE_ID	FIRST_NAME	LAST_NAME	EMAIL	PHONE_NUMBER	HIRE_DATE	JOB_ID	SALARY	COMMISSION_PCT	MANAGER_ID	DEPARTMENT_ID
1	110	John	Chen	JCHEN	515.124.4269	28-SEP-05	FI_ACCOUNT	8200	(null)	108	100
2	112	Jose Manuel	Urman	JMURMAN	515.124.4469	07-MAR-06	FI_ACCOUNT	7800	(null)	108	100

SQL: DQL: SELECT

▶ IN CLAUSE

▶ NOT IN

```
SELECT column1, column2....columnN
FROM   table_name
WHERE  column_name IN (val-1, val-2,...val-N);
```

```
110 | SELECT *
111 | FROM EMPLOYEES
112 | WHERE DEPARTMENT_ID IN (100,90);
```

Query Result x

SQL | All Rows Fetched: 9 in 0.003 seconds

	EMPLOYEE_ID	FIRST_NAME	LAST_NAME	EMAIL	PHONE_NUMBER	HIRE_DATE	JOB_ID	SALARY	COMMISSION_PCT	MANAGER_ID	DEPARTMENT_ID
1	100	Steven	King	SKING	515.123.4567	17-JUN-03	AD_PRES	24000	(null)	(null)	90
2	101	Neena	Kochhar	NKOCHHAR	515.123.4568	21-SEP-05	AD_VP	17000	(null)	100	90
3	102	Lex	De Haan	LDEHAAN	515.123.4569	13-JAN-01	AD_VP	17000	(null)	100	90
4	108	Nancy	Greenberg	NGREENBE	515.124.4569	17-AUG-02	FI_MGR	12008	(null)	101	100
5	109	Daniel	Faviet	DFAVIET	515.124.4169	16-AUG-02	FI_ACCOUNT	9000	(null)	108	100
6	110	John	Chen	JCHEN	515.124.4269	28-SEP-05	FI_ACCOUNT	8200	(null)	108	100
7	111	Ismael	Sciarra	ISCIARRA	515.124.4369	30-SEP-05	FI_ACCOUNT	7700	(null)	108	100

▶ NULL IN WHERE?

▶ IS NULL

▶ IS NOT NULL

```
114 | -- WRONG: USE IS NULL/ IS NOT NULL
115 | SELECT *
116 | FROM EMPLOYEES
117 | WHERE DEPARTMENT_ID = NULL;
118 |
119 | -- WRONG: USE IS NULL/ IS NOT NULL
120 | SELECT *
121 | FROM EMPLOYEES
122 | WHERE DEPARTMENT_ID <> NULL;
123 |
124 | SELECT *
125 | FROM EMPLOYEES
126 | WHERE DEPARTMENT_ID IS NULL;
127 |
128 | SELECT *
129 | FROM EMPLOYEES
130 | WHERE DEPARTMENT_ID IS NOT NULL;
```

SQL: DQL: SELECT

► ORDER BY

```
SELECT column1, column2....columnN
FROM   table_name
WHERE  CONDITION
ORDER BY column_name {ASC|DESC};
```

```
143 SELECT *
144 FROM EMPLOYEES
145 WHERE DEPARTMENT_ID = 50
146 ORDER BY SALARY;
```

Explain Plan x Query Result x

SQL | All Rows Fetched: 45 in 0.014 seconds

	EMPLOYEE_ID	FIRST_NAME	LAST_NAME	EMAIL	PHONE_NUMBER	HIRE_DATE	JOB_ID	SALARY	COMMISSION_PCT	MANAGER_ID	DEPARTMENT_ID
1	132	TJ	Olson	TJOLSON	650.124.8234	10-APR-07	ST_CLERK	2100	(null)	121	50
2	136	Hazel	Philtanker	HPHILTAN	650.127.1634	06-FEB-08	ST_CLERK	2200	(null)	122	50
3	128	Steven	Markle	SMARKLE	650.124.1434	08-MAR-08	ST_CLERK	2200	(null)	120	50
4	135	Ki	Gee	KGEE	650.127.1734	12-DEC-07	ST_CLERK	2400	(null)	122	50
5	127	James	Landry	JLANDRY	650.124.1334	14-JAN-07	ST_CLERK	2400	(null)	120	50
6	140	Joshua	Patel	JPATEL	650.121.1834	06-APR-06	ST_CLERK	2500	(null)	123	50

- ## ▶ HAVING CLAUSE

[illegible]

SQL: DQL: SELECT

- ▶ SELECT FROM MULTIPLE TABLES
 - ▶ CARTESIAN MULTIPLICATION

```
212 --CARTESIAN MULTIPLICATIONS
213 SELECT E.*, J.JOB_ID AS N_JOB_ID
214 FROM EMPLOYEES E, JOBS J;
215
216 SELECT COUNT(*) AS TABLE_DIM, COUNT(DISTINCT E.EMPLOYEE_ID) AS EMP_DIM, COUNT(DISTINCT J.JOB_ID) AS JOB_DIM
217 SELECT COUNT(*) AS TABLE_DIM, COUNT(DISTINCT E.EMPLOYEE_ID) AS EMP_DIM, COUNT(DISTINCT J.JOB_ID) AS JOB_DIM
218 FROM EMPLOYEES E, JOBS J;
```

- ▶ JOIN

```
--JOIN
SELECT COUNT(*)
FROM EMPLOYEES E, JOBS J
WHERE E.JOB_ID = J.JOB_ID;

SELECT COUNT(*)
FROM EMPLOYEES E
JOIN JOBS J ON E.JOB_ID = J.JOB_ID;
```


About Me

- ▶ Ehsan Hamzei
- ▶ Graduate Student of Geospatial Information System
- ▶ University of Tehran
- ▶ Git-Hub: <https://github.com/ehsan-hamzei/>
- ▶ Source-Codes: <https://github.com/ehsan-hamzei/Oracle-SQL/>