Fair Use Notice:

The material used in this presentation i.e., pictures/graphs/text, etc. is solely intended for educational/teaching purpose, offered free of cost to the students for use under special circumstances of Online Education due to COVID-19 Lockdown situation and may include copyrighted material - the use of which may not have been specifically authorised by Copyright Owners. It's application constitutes Fair Use of any such copyrighted material as provided in globally accepted law of many countries. The contents of presentations are intended only for the attendees of the class being conducted by the presenter.

DATABASE SYSTEMS (SW215)

JOINS

By : HIRA NOMAN

EQUI-JOIN FORMATION THROUGH USING CLAUSE

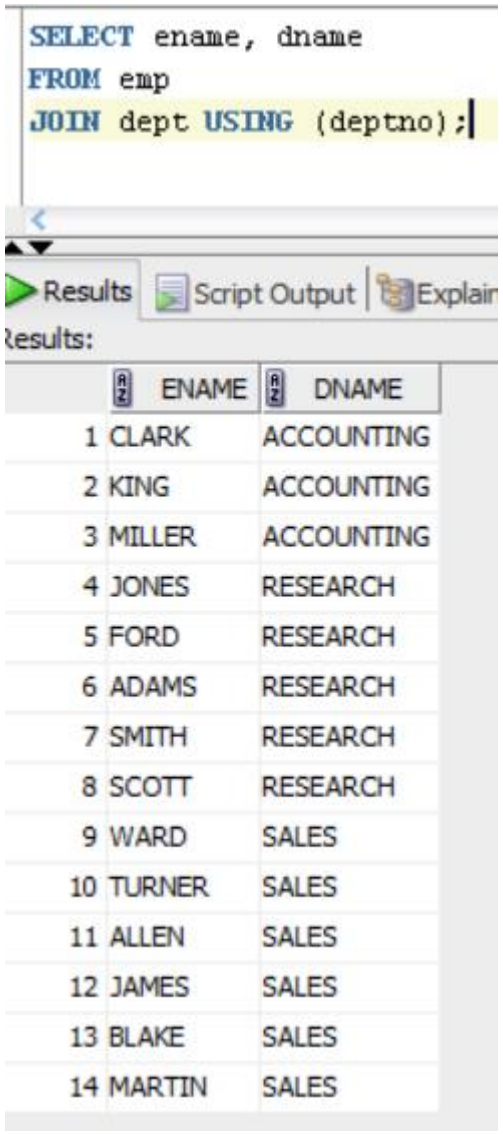
- **NATURAL JOIN** uses all the columns with matching names and datatypes to join the tables. The USING Clause can be used to specify only those columns that should be used for an EQUIJOIN.
- If several columns have the same names but the datatypes do not match, the NATURAL JOIN clause can be modified with the **USING** clause to specify the columns that should be used for an EQUIJOIN.
- Use the USING clause to specify the columns for the equijoin where several columns have the same names but not same data types.
- Use the USING clause to match only one column when more than one column matches.
- The NATURAL JOIN and USING clauses are mutually exclusive and error occurs if the NATURAL and USING keywords occur in the same join clause.
- When we use the USING clause in a join statement, the join column is not qualified with table aliases. Do not use alias even if the same column is used elsewhere in the SQL statement.
- The columns that are common in both the tables, but not used in the USING clause, must be prefixed with a table alias.

SYNTAX:

```
SELECT table1.column, table2.column
FROM table1
JOIN table2 USING ( join_column1 [ , join_column2…] );
```

EXAMPLE:

```
SELECT ename, dname
FROM emp
JOIN dept USING ( deptno);
```



```
SELECT ename, dname
FROM emp
JOIN dept USING (deptno);
```

Results | Script Output | Explain

Results:

	ENAME	DNAME
1	CLARK	ACCOUNTING
2	KING	ACCOUNTING
3	MILLER	ACCOUNTING
4	JONES	RESEARCH
5	FORD	RESEARCH
6	ADAMS	RESEARCH
7	SMITH	RESEARCH
8	SCOTT	RESEARCH
9	WARD	SALES
10	TURNER	SALES
11	ALLEN	SALES
12	JAMES	SALES
13	BLAKE	SALES
14	MARTIN	SALES

	EMPNO	ENAME	JOB	MGR	HIREDATE	SAL	COMM	DEPTNO	DNAME
1	7369	SMITH	CLERK	7902	17-DEC-80	800	(null)	20	RESEARCH
2	7499	ALLEN	SALESMAN	7698	20-FEB-81	1600	300	30	SALES
3	7521	WARD	SALESMAN	7698	22-FEB-81	1250	500	30	SALES
4	7566	JONES	MANAGER	7839	02-APR-81	2975	(null)	20	RESEARCH
5	7654	MARTIN	SALESMAN	7698	28-SEP-81	1250	1400	30	SALES
6	7698	BLAKE	MANAGER	7839	01-MAY-81	2850	(null)	30	SALES
7	7782	CLARK	MANAGER	7839	09-JUN-81	2450	(null)	10	ACCOUNTING
8	7788	SCOTT	ANALYST	7566	19-APR-87	3000	(null)	20	RESEARCH
9	7839	KING	PRESIDENT	(null)	17-NOV-81	5000	(null)	10	ACCOUNTING
10	7844	TURNER	SALESMAN	7698	08-SEP-81	1500	0	30	SALES
11	7876	ADAMS	CLERK	7788	23-MAY-87	1100	(null)	20	RESEARCH
12	7900	JAMES	CLERK	7698	03-DEC-81	950	(null)	30	SALES
13	7902	FORD	ANALYST	7566	03-DEC-81	3000	(null)	20	RESEARCH
14	7934	MILLER	CLERK	7782	23-JAN-82	1300	(null)	10	ACCOUNTING

```
SELECT sal
FROM emp4
JOIN dept USING (dname);
```

Results | Script Output | Exp

Results:

	SAL
1	800
2	1600
3	1250
4	2975
5	1250
6	2850
7	2450
8	3000
9	5000
10	1500
11	1100
12	950
13	3000
14	1300

Use the USING clause to match only one column when more than one column matches

```
Name                          Null     Type
----------------------------- -------- ----------------
EMPNO                                  NUMBER(4)
ENAME                                  VARCHAR2(10)
JOB                                    VARCHAR2(9)
MGR                                    NUMBER(4)
HIREDATE                               DATE
SAL                                    NUMBER(7,2)
COMM                                   NUMBER(7,2)
DEPTNO                                 NUMBER(2)
DNAME                                  VARCHAR2(10)

9 rows selected

desc emp4
Name                          Null     Type
----------------------------- -------- ----------------
EMPNO                                  NUMBER(4)
ENAME                                  VARCHAR2(10)
JOB                                    VARCHAR2(9)
MGR                                    NUMBER(4)
HIREDATE                               DATE
SAL                                    NUMBER(7,2)
COMM                                   NUMBER(7,2)
DEPTNO                                 NUMBER(2)
DNAME                                  CHAR(10)
```

```
SELECT SAL , ENAME , DNAME
FROM EMP4 JOIN DEPT USING(DNAME)
```

Results | Script Output | Explain | Autotrace

Results:

	SAL	ENAME	DNAME
1	2450	CLARK	ACCOUNTING
2	5000	KING	ACCOUNTING
3	1300	MILLER	ACCOUNTING

```
SELECT sal,ename,emp4.dname
FROM emp4
JOIN dept USING (DEPTNO);
```

Results | Script Output | Explain | Auto

Results:

	SAL	ENAME	DNAME
1	800	SMITH	RESEARCH
2	1600	ALLEN	SALES
3	1250	WARD	SALES
4	2975	JONES	RESEARCH
5	1250	MARTIN	SALES
6	2850	BLAKE	SALES
7	2450	CLARK	ACCOUNTING
8	3000	SCOTT	RESEARCH
9	5000	KING	ACCOUNTING
10	1500	TURNER	SALES
11	1100	ADAMS	RESEARCH
12	950	JAMES	SALES
13	3000	FORD	RESEARCH
14	1300	MILLER	ACCOUNTING

```
SELECT sal,ename,dname
FROM emp4
NATURAL JOIN dept;
```

Results | Script Output | Explain | Autotr

Results:

	SAL	ENAME	DNAME
1	2450	CLARK	ACCOUNTING
2	5000	KING	ACCOUNTING
3	1300	MILLER	ACCOUNTING

Use the USING clause to specify the columns for the equijoin where several columns have the same names but not same data types.

```
SELECT * FROM EMP4 JOIN DEPT USING (DEPTNO,DNAME)
```

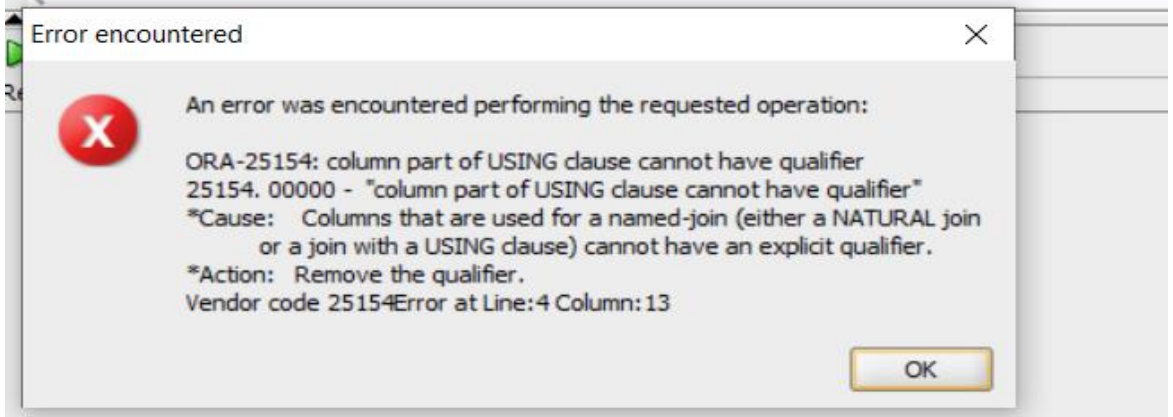
Results | Script Output | Explain | Autotrace | DBMS Output | OWA Output

esults:

	DEPTNO	DNAME	EMPNO	ENAME	JOB	MGR	HIREDATE	SAL	COMM	LOC
1	20	RESEARCH	7369	SMITH	CLERK	7902	17-DEC-80	800	(null)	DALLAS
2	30	SALES	7499	ALLEN	SALESMAN	7698	20-FEB-81	1600	300	CHICAGO
3	30	SALES	7521	WARD	SALESMAN	7698	22-FEB-81	1250	500	CHICAGO
4	20	RESEARCH	7566	JONES	MANAGER	7839	02-APR-81	2975	(null)	DALLAS
5	30	SALES	7654	MARTIN	SALESMAN	7698	28-SEP-81	1250	1400	CHICAGO
6	30	SALES	7698	BLAKE	MANAGER	7839	01-MAY-81	2850	(null)	CHICAGO
7	10	ACCOUNTING	7782	CLARK	MANAGER	7839	09-JUN-81	2450	(null)	NEW YORK
8	20	RESEARCH	7788	SCOTT	ANALYST	7566	19-APR-87	3000	(null)	DALLAS
9	10	ACCOUNTING	7839	KING	PRESIDENT	(null)	17-NOV-81	5000	(null)	NEW YORK
10	30	SALES	7844	TURNER	SALESMAN	7698	08-SEP-81	1500	0	CHICAGO
11	20	RESEARCH	7876	ADAMS	CLERK	7788	23-MAY-87	1100	(null)	DALLAS
12	30	SALES	7900	JAMES	CLERK	7698	03-DEC-81	950	(null)	CHICAGO
13	20	RESEARCH	7902	FORD	ANALYST	7566	03-DEC-81	3000	(null)	DALLAS
14	10	ACCOUNTING	7934	MILLER	CLERK	7782	23-JAN-82	1300	(null)	NEW YORK
15	30	SALES	3214	TIM	CLERK	7782	25-FEB-21	6000	600	CHICAGO

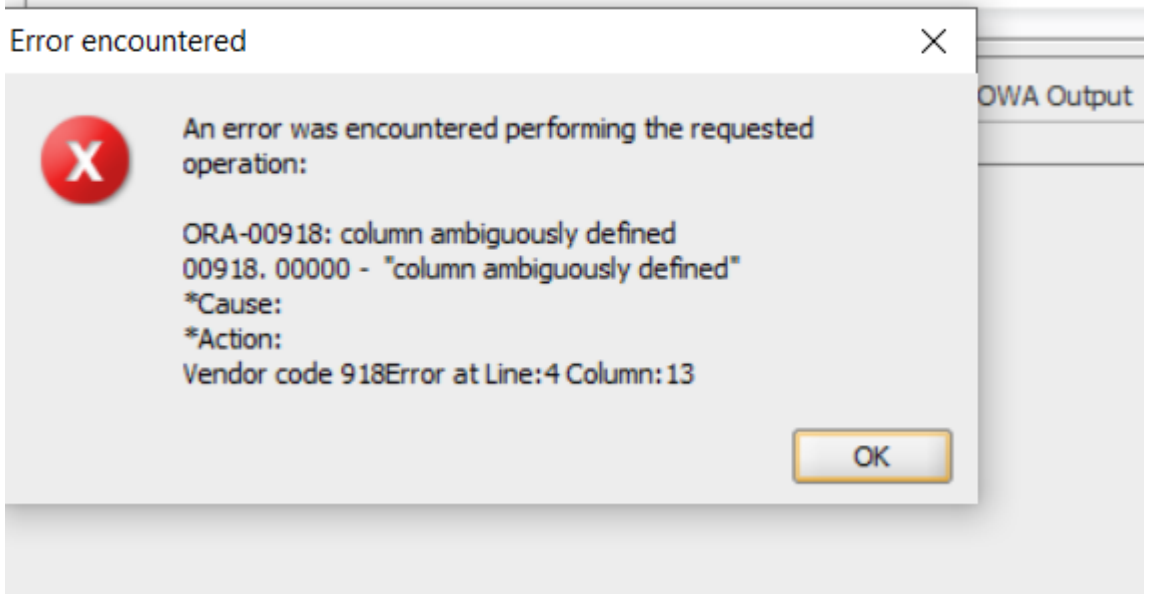
USING with multiple common columns behaves like AND operator as well

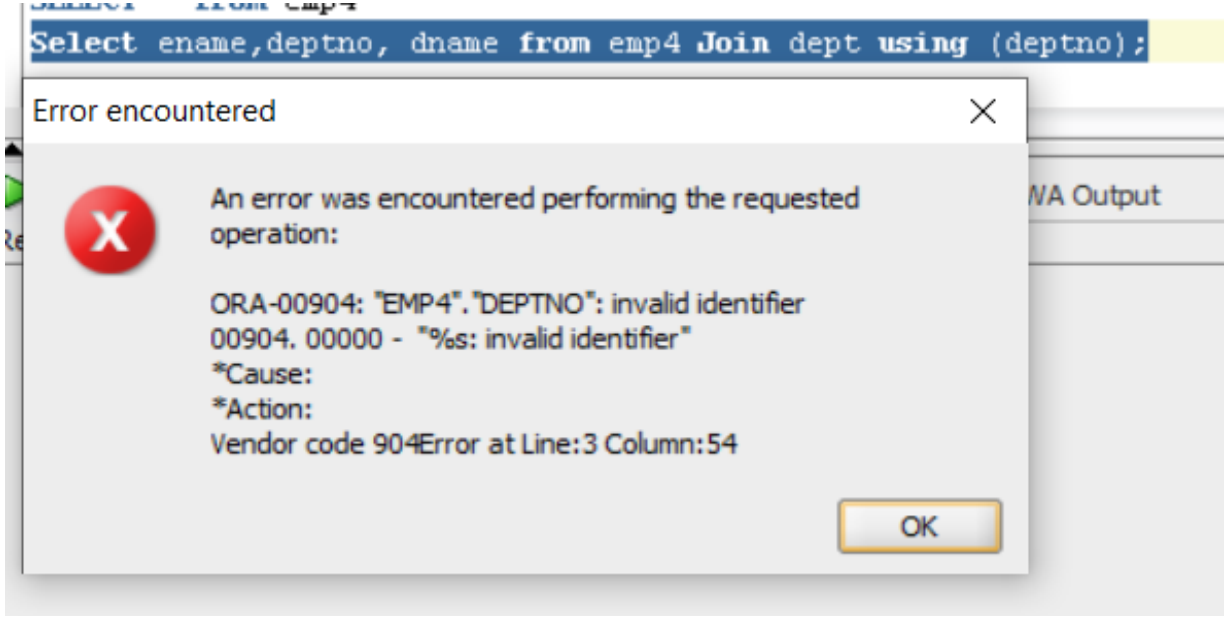
```
Select ename,emp4.deptno,emp4.dname from emp4 Join dept using (deptno);
```



The join column is not qualified with table aliases. Do not use alias even if the same column is used elsewhere in the SQL statement.

```
Select ename,dname from emp4 Join dept using (deptno);
```





The columns that are common in both the tables, but not used in the USING clause, must be prefixed with a table alias.

TASK A

Write SQL query to find the working location of the employees. Also give their respective employee_id and last_name.

```
SELECT e.EMPLOYEE_ID , e.LAST_NAME,
d.LOCATION_ID
FROM Employees e JOIN Departments d
USING(DEPARTMENT_ID);
```

```
SELECT e.EMPLOYEE_ID, e.LAST_NAME, d.LOCATION_ID
FROM Employees e JOIN Departments d
USING(DEPARTMENT_ID)
```

Results | Script Output | Explain | Autotrace | DBMS Output

Results:

	EMPLOYEE_ID	LAST_NAME	LOCATION_ID
1	200	Whalen	1700
2	201	Hartstein	1800
3	202	Fay	1800
4	114	Raphaely	1700
5	119	Colmenares	1700
6	115	Khoo	1700
7	116	Baida	1700
8	117	Tobias	1700
9	118	Himuro	1700
10	203	Mavris	2400
11	198	OConnell	1500
12	199	Grant	1500
13	120	Weiss	1500
14	121	Fripp	1500
15	122	Kaufling	1500
16	123	Vollman	1500
17	124	Mourgos	1500
18	125	Nayer	1500
19	126	Mikkilineni	1500
20	127	Landry	1500
21	128	Markle	1500
22	129	Bissot	1500
23	130	Atkinson	1500
24	131	Marlow	1500
25	132	Olson	1500

TASK B

For the job title MANAGER display the following details:

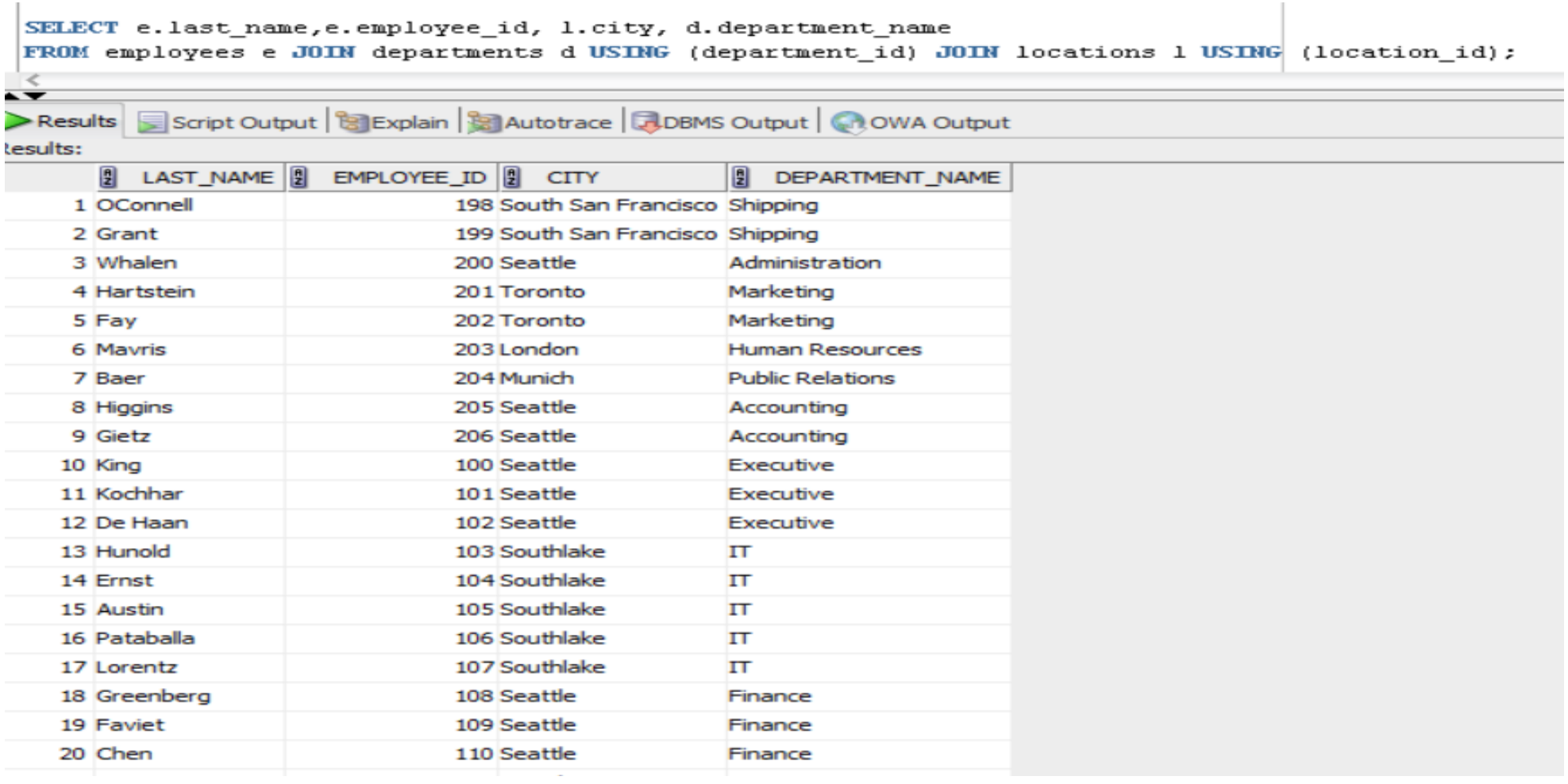
Employee number ,names ,job ,manager ,salary ,commission , department number ,department name and location.

```
SELECT EMP.* , DEPT.DNAME , DEPT.LOC
FROM EMP , DEPT
WHERE EMP.DEPTNO = DEPT.DEPTNO AND EMP.JOB = 'MANAGER'
```

	EMPNO	ENAME	JOB	MGR	HIREDATE	SAL	COMM	DEPTNO	DNAME	LOC
1	7782	CLARK	MANAGER	7839	09-JUN-81	2450	(null)	10	ACCOUNTING	NEW YORK
2	7566	JONES	MANAGER	7839	02-APR-81	2975	(null)	20	RESEARCH	DALLAS
3	7698	BLAKE	MANAGER	7839	01-MAY-81	2850	(null)	30	SALES	CHICAGO

Creating Three-Way Joins with the USING Clause

```
SELECT e.last_name,e.employee_id, l.city, d.department_name
FROM employees e JOIN departments d USING (department_id)
                 JOIN locations l   USING (location_id);
```



```
SELECT e.last_name,e.employee_id, l.city, d.department_name
FROM employees e JOIN departments d USING (department_id) JOIN locations l USING (location_id);
```

Results | Script Output | Explain | Autotrace | DBMS Output | OWA Output

Results:

	LAST_NAME	EMPLOYEE_ID	CITY	DEPARTMENT_NAME
1	OConnell	198	South San Francisco	Shipping
2	Grant	199	South San Francisco	Shipping
3	Whalen	200	Seattle	Administration
4	Hartstein	201	Toronto	Marketing
5	Fay	202	Toronto	Marketing
6	Mavris	203	London	Human Resources
7	Baer	204	Munich	Public Relations
8	Higgins	205	Seattle	Accounting
9	Gietz	206	Seattle	Accounting
10	King	100	Seattle	Executive
11	Kochhar	101	Seattle	Executive
12	De Haan	102	Seattle	Executive
13	Hunold	103	Southlake	IT
14	Ernst	104	Southlake	IT
15	Austin	105	Southlake	IT
16	Pataballa	106	Southlake	IT
17	Lorentz	107	Southlake	IT
18	Greenberg	108	Seattle	Finance
19	Faviet	109	Seattle	Finance
20	Chen	110	Seattle	Finance

EQUI-JOIN FORMATION THROUGH ON CLAUSE

- The join condition for the natural join is basically an equijoin of identical column names.
- ON clause can be used to join columns that have different names.
- Use the ON clause to specify conditions or specify columns to join.
- The join condition is separated from other search conditions.
- This is the easiest and widely used form of the join clauses.
- An error occurs if the NATURAL and ON keywords occur in the same join clause.
- The JOIN…ON clause allows one or more equijoin columns to specify in brackets after the ON keyword.
- The equijoin columns are fully qualified as table1.column_name = table2.column_name (optionally specified in brackets) after the ON keyword.

SYNTAX:

```
SELECT select_list
FROM table_1 JOIN table_2
ON ( table_1.column_name  =  table_2.column_name )
```

```
Select ename,d.dname, d.deptno from emp4 e Join dept d ON e.deptno=d.deptno;
```

Results | Script Output | Explain | Autotrace | DBMS Output | OWA Output

esults:

	ENAME	DNAME	DEPTNO
1	CLARK	ACCOUNTING	10
2	KING	ACCOUNTING	10
3	MILLER	ACCOUNTING	10
4	JONES	RESEARCH	20
5	FORD	RESEARCH	20
6	ADAMS	RESEARCH	20
7	SMITH	RESEARCH	20
8	SCOTT	RESEARCH	20
9	WARD	SALES	30
10	TURNER	SALES	30
11	ALLEN	SALES	30
12	JAMES	SALES	30
13	BLAKE	SALES	30
14	MARTIN	SALES	30

QUALIFIER CAN BE USED WITH THE COLUMN USED TO CREATE THE JOIN.

```
ALTER TABLE EMP4 RENAME COLUMN deptno to departmentno
SELECT * from emp4
```

Results | Script Output | Explain | Autotrace | DBMS Output | OWA Output

Results:

	EMPNO	ENAME	JOB	MGR	HIREDATE	SAL	COMM	DEPARTMENTNO	DNAME
1	7369	SMITH	CLERK	7902	17-DEC-80	800	(null)	20	(null)
2	7499	ALLEN	SALESMAN	7698	20-FEB-81	1600	300	30	(null)
3	7521	WARD	SALESMAN	7698	22-FEB-81	1250	500	30	(null)
4	7566	JONES	MANAGER	7839	02-APR-81	2975	(null)	20	(null)

```
ALTER TABLE EMP4 RENAME COLUMN deptno to departmentno
SELECT * from emp4
Select ename, d.deptno from emp4 e Join dept d ON e.departmentno = d.deptno;
```

Results | Script Output | Explain | Autotrace | DBMS Output | OWA Output

esults:

	ENAME	DEPTNO
1	SMITH	20
2	ALLEN	30
3	WARD	30
4	JONES	20
5	MARTIN	30
6	BLAKE	30
7	CLARK	10
8	SCOTT	20
9	KING	10
10	TURNER	30
11	ADAMS	20
12	JAMES	30
13	FORD	20
14	MILLER	10

ON clause can be used to join columns that have different names

INNER JOIN

- Inner join and natural join are almost same but there is a slight difference between them. The difference is in natural join no need to specify condition but in inner join condition is obligatory. If we do specify the condition in inner join , it resultant tables is like a cartesian product.

SR.NO.	NATURAL JOIN	INNER JOIN
1.	Natural Join joins two tables based on same attribute name and datatypes.	Inner Join joins two table on the basis of the column which is explicitly specified in the ON clause.
2.	In Natural Join, The resulting table will contain all the attributes of both the tables but keep only one copy of each common column	In Inner Join, The resulting table will contain all the attribute of both the tables including duplicate columns also

EQUI-JOIN FORMATION THROUGH INNER JOIN

- EQUI-JOIN can be formed through INNER JOIN by using the ON clause.

SYNTAX:

```
SELECT column_name(s)
FROM table1
INNER JOIN table2
ON (table1.column_name = table2.column_name) ;


SELECT column_name(s)
FROM (( table1
INNER JOIN table2 ON table1.column_name = table2.column_name )
INNER JOIN table3 ON table2.column_name = table3.column_name ) ;
```

```
SELECT * FROM EMP INNER JOIN DEPT ON (EMP.DEPTNO = DEPT.DEPTNO)
```

Results | Script Output | Explain | Autotrace | DBMS Output | OWA Output

Results:

	EMPNO	ENAME	JOB	MGR	HIREDATE	SAL	COMM	DEPTNO	DEPTNO_1	DNAME	LOC
1	7782	CLARK	MANAGER	7839	09-JUN-81	2450	(null)	10	10	ACCOUNTING	NEW YORK
2	7839	KING	PRESIDENT	(null)	17-NOV-81	5000	(null)	10	10	ACCOUNTING	NEW YORK
3	7934	MILLER	CLERK	7782	23-JAN-82	1300	(null)	10	10	ACCOUNTING	NEW YORK
4	7566	JONES	MANAGER	7839	02-APR-81	2975	(null)	20	20	RESEARCH	DALLAS
5	7902	FORD	ANALYST	7566	03-DEC-81	3000	(null)	20	20	RESEARCH	DALLAS
6	7876	ADAMS	CLERK	7788	23-MAY-87	1100	(null)	20	20	RESEARCH	DALLAS
7	7369	SMITH	CLERK	7902	17-DEC-80	800	(null)	20	20	RESEARCH	DALLAS
8	7788	SCOTT	ANALYST	7566	19-APR-87	3000	(null)	20	20	RESEARCH	DALLAS
9	7521	WARD	SALESMAN	7698	22-FEB-81	1250	500	30	30	SALES	CHICAGO
10	7844	TURNER	SALESMAN	7698	08-SEP-81	1500	0	30	30	SALES	CHICAGO
11	7499	ALLEN	SALESMAN	7698	20-FEB-81	1600	300	30	30	SALES	CHICAGO
12	7900	JAMES	CLERK	7698	03-DEC-81	950	(null)	30	30	SALES	CHICAGO
13	7698	BLAKE	MANAGER	7839	01-MAY-81	2850	(null)	30	30	SALES	CHICAGO
14	7654	MARTIN	SALESMAN	7698	28-SEP-81	1250	1400	30	30	SALES	CHICAGO