Historical Joins 8i

6-1

Equi join

Department Table

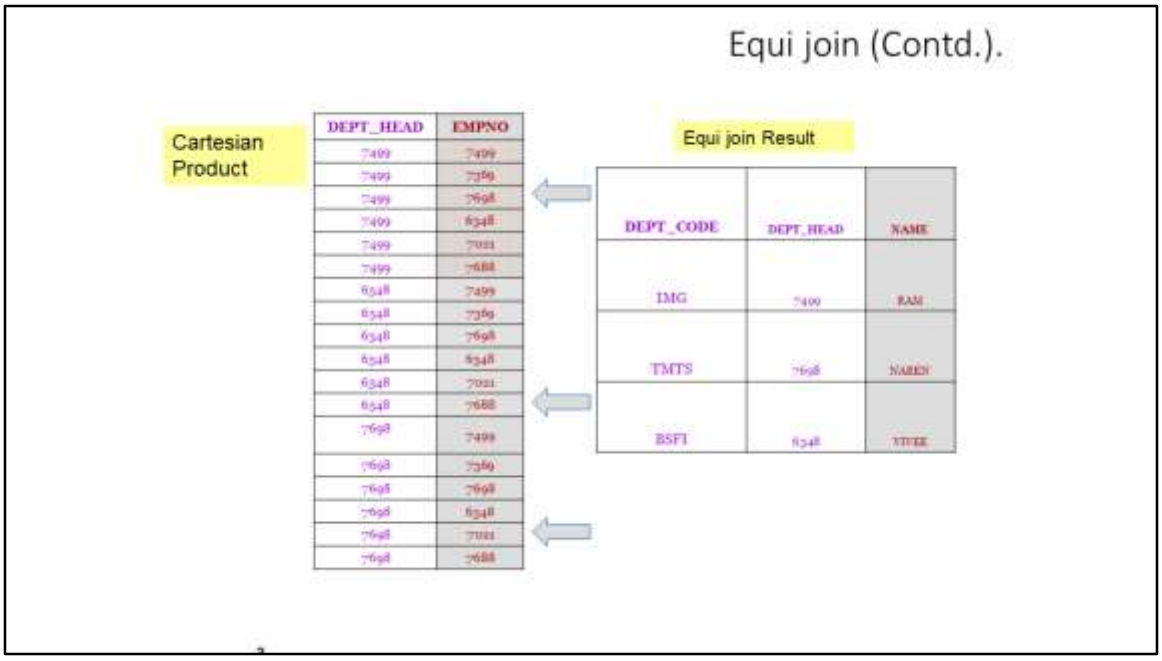
DEPT_CODE	DEPT_HEAD
IMG	7499
BSFI	6348
TMTS	7698
NEW1	
NEW2	

Employees Table

EMPNO	NAME	DEPT_CODE	LOC_CODE
7499	RAM	IMG	BDC
7369	GOPAL	BSFI	BDC
7698	NAREN	TMTS	CDC
6348	VIVEK	BSFI	CDC
7021	JOSEPH	IMG	PDC
7688	RAHEEM	IMG	HDC

Joining Purpose: to List out Department heads and their names

Joining Condition : Department.dept_Head = Employees.EMPNO



Note: NULL Values are ignored during join process.

If NO joining condition is present in the query, Oracle will give Cartesian product of two table rows in the select statement.

Non Equi join

Income Tax (Alias T)

Low_Ann_Sal	High_Ann_Sal	IT_Slab
10000	12000	1
12001	16000	2
16001	22000	3
22001	99999	4

Employees Table (Alias E)

EMPNO	NAME	DEPT_CODE	Ann_SAL
7499	RAM	IMG	12000
7369	GOPAL	BSFI	14000
7698	NAREN	TMTS	17000
6348	VIVEK	BSFI	12000
7021	JOSEPH	IMG	15000
7688	RAHEEM	IMG	28000

Joining Purpose: to List out Employees and their respective Income Tax Slab

Joining Condition: E.Ann_Sal T.Low_Ann_Sal <= E.Ann_Sal T.High_Ann_Sal

Non Equi join Result

EMPNO	ENAME	DEPT_CODE	Ann_SAL
7499	RAM	IMG	12000
7369	GOPAL	BSFI	14000
7698	NAREN	TMTS	17000
6348	VIVEK	BSFI	12000
7021	JOSEPH	IMG	15000
7688	RAHEEM	IMG	28000

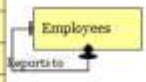
Low_Ann_Sal	High_Ann_Sal	IT_Slab
10000	12000	1
12001	16000	2
16001	22000	3
10000	12000	1
12001	16000	2
22001	99999	4

Select Statement

```
SELECT EMPNO,ENAME,DEPT_CODE,ANN_SAL,Low_Ann_Sal,High_Ann_Sal,IT_SLAB
FROM EMPLOYEES E , ITAX_SLAB T
WHERE E.ANN_SAL BETWEEN T.LOW_ANN_SAL AND T.HIGH_ANN_SAL
```

Self Join

EMPNO	ENAME	JOB	DEPTNO	HIREDATE	MGR	SAL	COMM
7788	ARUN	ANALYST	10	19-Apr-87	7566	2200	
7844	TURNER	SALESMAN	10	8-Sep-81	7698	3000	0
7934	MILLER	CLERK	20	23-Jan-82	7782	2500	
7900	CAPTAIN	CLERK	20	3-Dec-81	7698	2100	
7654	RAVI	SALESMAN	20	28-Sep-81	7698	2500	1400
7839	RAJ	PRESIDENT	20	17-Nov-81		1200	
7782	JOSEPH	MANAGER	30	9-Jun-81	7839	2500	
7566	RAGHU	MANAGER	30	2-Apr-81	7839	900	
7902	CHANDRAN	ANALYST	30	3-Dec-81	7566	2200	
7499	RAM	SALESMAN	30	20-Feb-81	7698	2100	300
7876	AKBAR	CLERK	30	23-May-87	7788	2100	
7698	ARJUN	MANAGER	30	1-May-81	7839	1600	
7521	SHYAM	SALESMAN	30	22-Feb-81	7698	800	300



Joining Purpose: to List out Employee details empno,ename,Job,deptno,hiredate,mgr, Employee's Manager's Name (who is also one among the Employees).

Self Join Result

EMPNO	ENAME	JOB	DEPTNO	HIREDATE	MGR	MANAGER
7788	ARUN	ANALYST	10	19-Apr-87	7566	RAGHU
7844	TURNER	SALESMAN	10	8-Sep-81	7698	ARJUN
7934	MILLER	CLERK	20	23-Jan-82	7782	JOSEPH
7900	CAPTAIN	CLERK	20	3-Dec-81	7698	ARJUN
7654	RAVI	SALESMAN	20	28-Sep-81	7698	ARJUN
7839	RAJ	PRESIDENT	20	17-Nov-81		
7782	JOSEPH	MANAGER	30	9-Jun-81	7839	RAJ
7566	RAGHU	MANAGER	30	2-Apr-81	7839	RAJ
7902	CHANDRAN	ANALYST	30	3-Dec-81	7566	RAGHU
7499	RAM	SALESMAN	30	20-Feb-81	7698	ARJUN
7876	AKBAR	CLERK	30	23-May-87	7788	ARUN
7698	ARJUN	MANAGER	30	1-May-81	7839	RAJ
7521	SHYAM	SALESMAN	30	22-Feb-81	7698	ARJUN

Employees table has been imitated as two different tables to be joined.

MGR column assumes the same domain of values as that of EMPNO Column.

Select Statement

```
SELECT E.EMPNO,E.ENAME,E.JOB, E.DEPTNO,E.HIREDATE,E.MGR,M.ENAME as MANAGER
FROM EMPLOYEES E , EMPLOYEES M
WHERE E.MGR = M.EMPNO
```

7

To understand the above Self Join SELECT statement, pretend that the column aliases given to employee table represent the following roles.

E – Employees (Employee role of Employees table)
M – Managers (Manager role of Employees table)

Left Outer Join

Department Table

DEPT_CODE	DEPT_HEAD
IMG	7499
BSFI	6348
TMTS	7698
NEW1	
NEW2	

EmployeesTable

EMPNO	NAME	DEPT_CODE	LOC_CODE
7499	RAM	IMG	BDC
7369	GOPAL	BSFI	BDC
7698	NAREN	TMTS	CDC
6348	VIVEK	BSFI	CDC
7021	JOSEPH	IMG	PDC
7688	RAHEEM	IMG	HDC

Joining Purpose: List department wise employee details, including the departments without any employees in it too.

Left Outer Join Result

Dept Left outer join Employees

DEPT_CODE	EMPNO	NAME	LOC_CODE
BSFI	7369	GOPAL	BDC
BSFI	6348	VIVEK	CDC
IMG	7499	RAM	BDC
IMG	7021	JOSEPH	PDC
IMG	7688	RAHEEM	HDC
NEW1	Null	Null	Null
NEW2	Null	Null	Null
TMTS	7698	NAREN	CDC


```
Select D.Dept_Code,Empno,Name,Loc_Code
from Dept Left Outer join Employees E
on D.dept_code = E.dept_code
Order by D.Dept_Code
```

Note: In the above query, if Employee table's deptcode column is selected instead of Department.deptcode, you will get Null even in the deptcode column for NEW1 and NEW2 departments and may not serve the purpose.

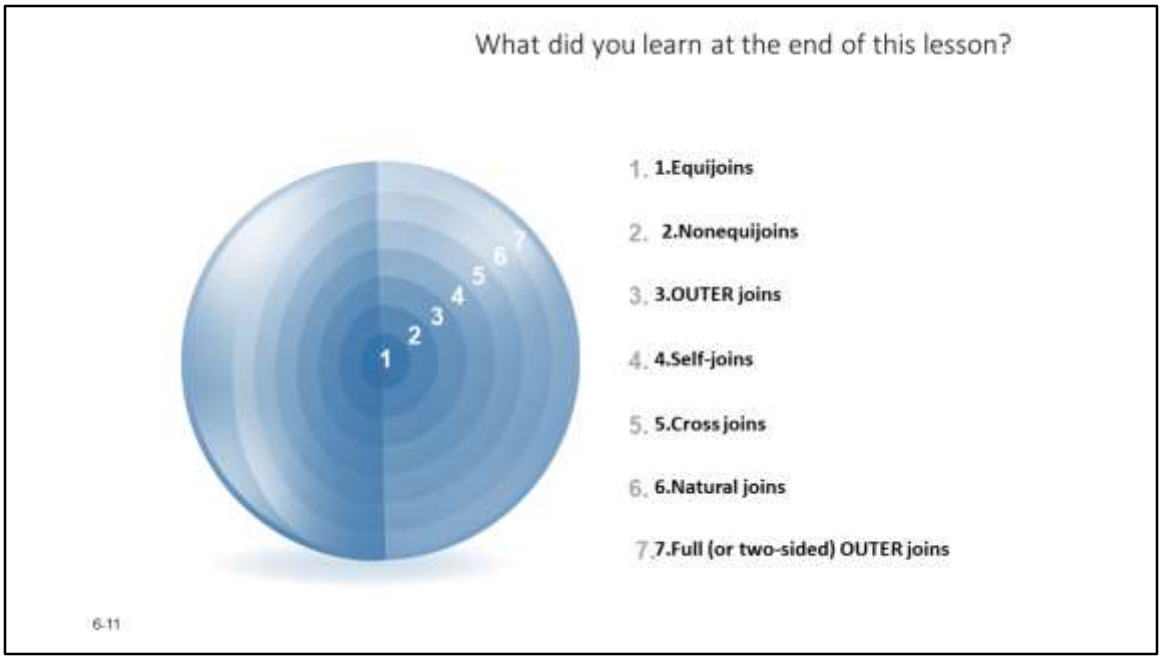
Quiz

The SQL:1999 standard join syntax supports the following types of joins. Which of these join types does Oracle join syntax support?

1. Equijoins
2. Nonequijoins
3. Left OUTER join
4. Right OUTER join
5. Full OUTER join
6. Self joins
7. Natural joins
8. Cartesian products

6-10

Answer: 1, 2, 3, 4, 6, 8



Summary

There are multiple ways to join tables.

Types of Joins

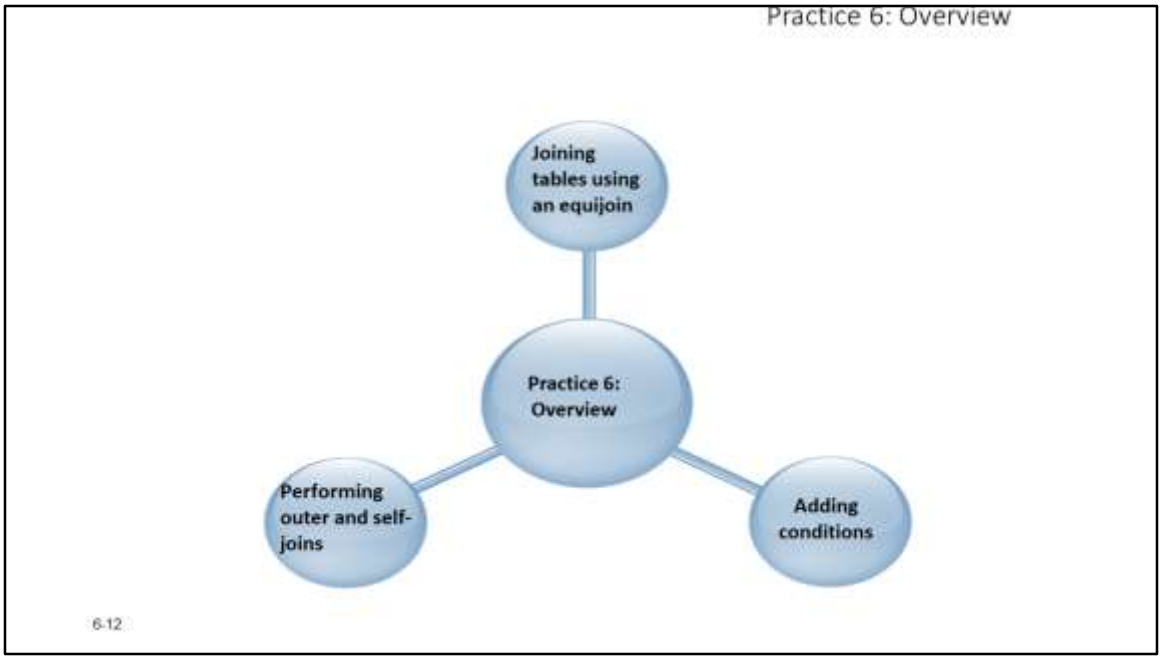
Equijoins
Nonequijoins
`OUTER` joins
Self-joins
Cross joins
Natural joins
Full (or two-sided) `OUTER` joins

Cartesian Products

A Cartesian product results in the display of all combinations of rows. This is done by either omitting the `WHERE` clause or by specifying the `CROSS JOIN` clause.

Table Aliases

Table aliases speed up database access.
Table aliases can help to keep SQL code smaller by conserving memory.
Table aliases are sometimes mandatory to avoid column ambiguity.



Practice 6: Overview

This practice is intended to give you experience in extracting data from more than one table using the SQL:1999–compliant joins.