**SELF JOIN**

**Joining a table to itself** is called self join

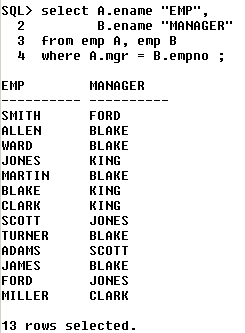
The **FROM** clause looks like this,

FROM emp A, emp B

Or

FROM emp A join emp B - *ANSI style*

For ex, - **Display employee name along with their manager name**

****

**Now, let us see how this i.e the logic (the above query) works,**

|  |  |  |
| --- | --- | --- |
| **Emp (A)** | | |
| **EmpNo** | **Ename** | **Mgr** |
| 101 | Scott | 102 |
| 102 | Blake | 103 |
| 103 | King | - |
| 104 | Smith | 103 |
| 105 | Jones | 104 |

|  |  |  |
| --- | --- | --- |
| **Emp (B)** | | |
| **EmpNo** | **Ename** | **Mgr** |
| 101 | Scott | 102 |
| 102 | Blake | 103 |
| 103 | King | - |
| 104 | Smith | 103 |
| 105 | Jones | 104 |

Now, when we give the above query – in Oracle – it starts matching the ‘**mgr**’ column of **emp A** with the ‘**empno**’ of **emp b** – we get two tables because in **self join** – a duplicate of the table required is created.

Now let us consider the **first employee Scott** – it starts the **mgrid** of **Scott** with the **empno** of all the records in **emp B** – when two **ids** match, then the **empno** in **emp B** becomes the **mgr** of the **empno** in **emp A**. Thus, we can see that – **mgr id** 102 is matching with **empno** 102 **Blake** in **emp B**. Therefore, Blake is the manager of Scott.

Similarly we do the same for all the other records of **emp A** and thus find the employees and their respective managers.

**Display the employees who are getting the same salary**

