

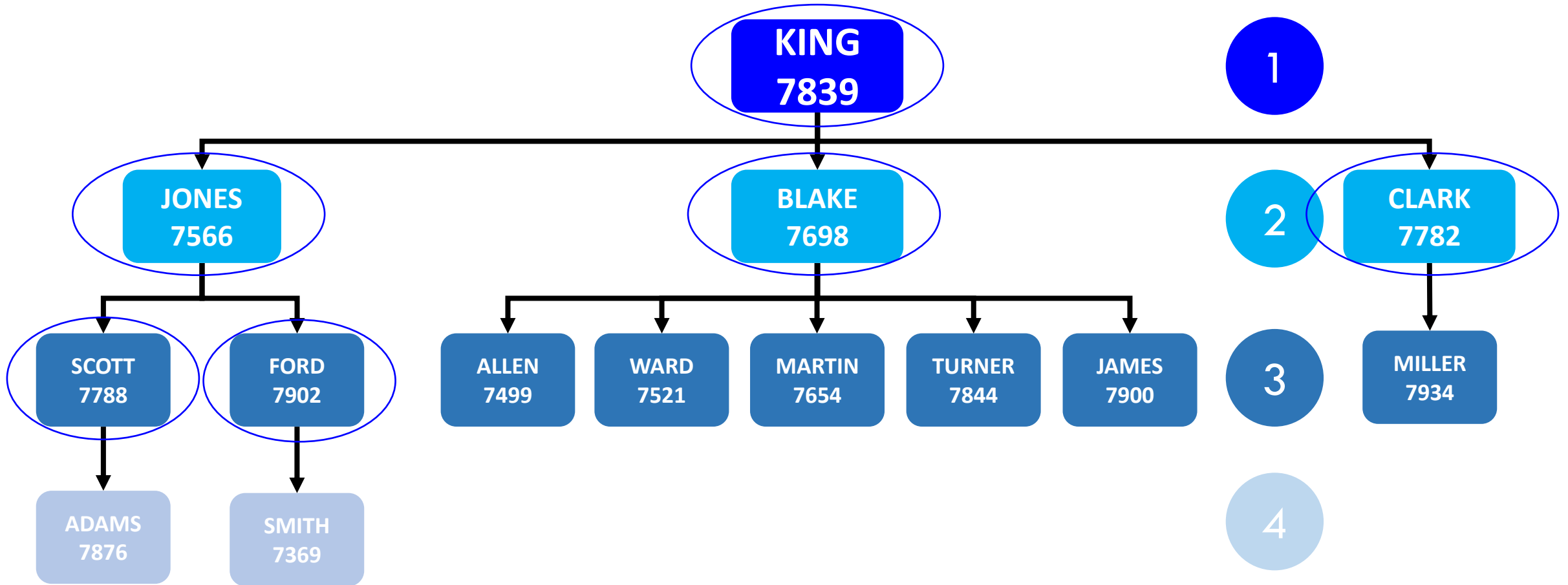
Hierarchical Queries (HQ)

- If a table contains hierarchical data then you can arrange the rows of this table in hierarchical order using HQ
- Hierarchical data in this context is where you have columns that are related to each other via a parent-child relationship, for example having an employee id and a manager id
- Examples of hierarchical data
 - Organizational
 - General Ledger
 - Product Categories
 - ...

Syntax

```
SELECT  
[COLUMNS],  
LEVEL,  
CONNECT BY ROOT [COLUMN],  
SYS_CONNECT_BY_PATH(PARAMETERS)  
FROM TABLE  
START WITH [CONDITION]  
CONNECT BY PRIOR [CONDITION]  
ORDER SIBLINGS BY [COLUMNS];
```

EMP Table Hierarchy



CONNECT BY PRIOR & START WITH

- CONNECT BY PRIOR specifies the relationship between parent and child rows
- PRIOR specifies the child column and hence the direction of the query
- START WITH specifies the root row(s) of the hierarchy, if this is not specified all roots will be returned

CONNECT BY ROOT

- This returns the value of the root row
- Used in the SELECT statement

CONNECT_BY_ROOT COLUMN

SYS CONNECT BY PATH

- This function shows you the path up to the root for each node, you just define a separator
- Used in the SELECT statement

SYS_CONNECT_BY_PATH(COLUMN,SEPERATOR)

LEVEL

- “Level” is a pseudo column that can be used in the SELECT statement only when a CONNECT BY CLAUSE is used
- It shows the level in the hierarchy for each returned row

ORDER BY SIBLINGS

- If we use the order by clause we no longer preserve the hierarchy
- ORDER BY SIBLINGS preserves the hierarchy primarily and then orders the results by the specified columns
- Can only be used in hierarchical queries