<https://docs.microsoft.com/en-us/sql/relational-databases/tables/lesson-1-1-examining-the-current-structure-of-the-employee-table>

Lesson 1-1 - Examining the Current Structure of the Employee Table

2017-3-1 1 min to read Contributors

* [BYHAM](https://github.com/BYHAM)

* [Craig Guyer](https://github.com/craigg-msft)

The sample **AdventureWorks2012** database contains an **Employee** table in the **HumanResources** schema. To avoid changing the original table, this step makes a copy of the **Employee** table named **EmployeeDemo**. To simplify the example, you only copy five columns from the original table. Then, you query the **HumanResources.EmployeeDemo** table to review how the data is structured in a table without using the **hierarchyid** data type.

To copy the Employee table

1. In a Query Editor window, run the following code to copy the table structure and data from the **Employee** table into a new table named **EmployeeDemo**.

Copy

USE AdventureWorks ;

GO

SELECT EmployeeID, LoginID, ManagerID, Title, HireDate

INTO HumanResources.EmployeeDemo

FROM HumanResources.Employee ;

GO

To examine the structure and data of the EmployeeDemo table

* This new **EmployeeDemo** table represents a typical table in an existing database that you might want to migrate to a new structure. In a Query Editor window, run the following code to show how the table uses a self join to display the employee/manager relationships:

Copy

SELECT

Mgr.EmployeeID AS MgrID, Mgr.LoginID AS Manager,

Emp.EmployeeID AS E\_ID, Emp.LoginID, Emp.Title

FROM HumanResources.EmployeeDemo AS Emp

LEFT JOIN HumanResources.EmployeeDemo AS Mgr

ON Emp.ManagerID = Mgr.EmployeeID

ORDER BY MgrID, E\_ID

Here is the result set.

Copy

MgrID Manager E\_ID LoginID Title

NULL NULL 109 adventure-works\ken0 Chief Executive Officer

3 adventure-works\roberto0 4 adventure-works\rob0 Senior Tool Designer

3 adventure-works\roberto0 9 adventure-works\gail0 Design Engineer

3 adventure-works\roberto0 11 adventure-works\jossef0 Design Engineer

3 adventure-works\roberto0 158 adventure-works\dylan0 Research and Development Manager

3 adventure-works\roberto0 263 adventure-works\ovidiu0 Senior Tool Designer

3 adventure-works\roberto0 267 adventure-works\michael8 Senior Design Engineer

3 adventure-works\roberto0 270 adventure-works\sharon0 Design Engineer

6 adventure-works\david0 2 adventure-works\kevin0 Marketing Assistant

...

The results continue for a total of 290 rows.

Notice that the **ORDER BY** clause caused the output to list the direct reports of each management level together. For instance, all seven of the direct reports of **MgrID** 3 (roberto0) are listed adjacent to each other. Although not impossible, it is much more difficult to group all those who eventually report to **MgrID** 3.

In the next task, we will create a new table with a **hierarchyid** data type, and move the data into the new table.