<https://docs.microsoft.com/en-us/sql/relational-databases/tables/lesson-2-1-creating-a-table-using-the-hierarchyid-data-type>

# Lesson 2-1 - Creating a Table Using the hierarchyid Data Type

2017-3-1 1 min to read Contributors

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

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

The following example creates a table named EmployeeOrg, which includes employee data together with their reporting hierarchy. The example creates the table in the **AdventureWorks2012** database, but that is optional. To keep the example simple, this table includes only five columns:

* OrgNode is a **hierarchyid** column that stores the hierarchical relationship.
* OrgLevel is a computed column, based on the OrgNode column that stores each nodes level in the hierarchy. It will be used for a breadth-first index.
* EmployeeID contains the typical employee identification number that is used for applications such as payroll. In new application development, applications can use the OrgNode column and this separate EmployeeID column is not needed.
* EmpName contains the name of the employee.
* Title contains the title of the employee.

### To create the EmployeeOrg table

1. In a Query Editor window, run the following code to create the EmployeeOrg table. Specifying the OrgNode column as the primary key with a clustered index will create a depth-first index:

Copy

USE AdventureWorks2012 ;

GO

CREATE TABLE HumanResources.EmployeeOrg

(

OrgNode hierarchyid PRIMARY KEY CLUSTERED,

OrgLevel AS OrgNode.GetLevel(),

EmployeeID int UNIQUE NOT NULL,

EmpName varchar(20) NOT NULL,

Title varchar(20) NULL

) ;

GO

1. Run the following code to create a composite index on the OrgLevel and OrgNode columns to support efficient breadth-first searches:

Copy

CREATE UNIQUE INDEX EmployeeOrgNc1

ON HumanResources.EmployeeOrg(OrgLevel, OrgNode) ;

GO

The table is now ready for data. The next task will populate the table by using hierarchical methods.

## Next Task in Lesson

[Populating a Hierarchical Table Using Hierarchical Methods](https://docs.microsoft.com/en-us/sql/relational-databases/tables/lesson-2-2-populating-a-hierarchical-table-using-hierarchical-methods)