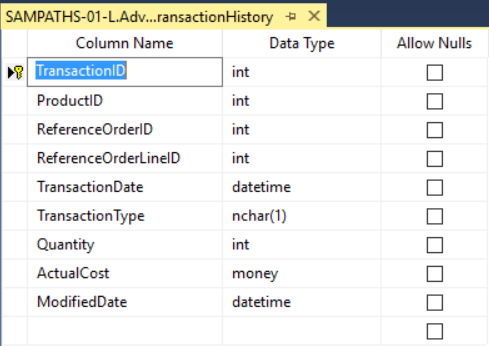
**JOINS PERFORMED ON TWO TABLES:**

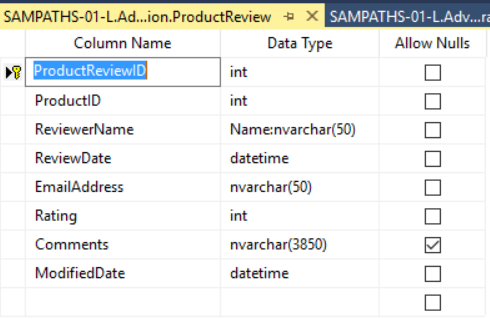
**DATABASE USED:** AdventureWorks2016CTP3

**SCHEMA DEFINITION**

**Production.TransactionHistory**



**Production.ProductReview**

****

**Set of all products with their reviews and comments.**

USE AdventureWorks2016CTP3

SELECT

t.ProductID,

r.ProductReviewID,

r.ReviewerName,

t.TransactionType,

r.comments,

t.ActualCost

FROM Production.TransactionHistory t

LEFT JOIN Production.ProductReview r

ON r.ProductID = t.ProductID

**Set of only those products that are commented and reviewed by the reviewers.**

USE AdventureWorks2016CTP3

SELECT

t.ProductID,

r.ProductReviewID,

r.ReviewerName,

t.TransactionType,

r.comments,

t.ActualCost

FROM Production.TransactionHistory t

INNER JOIN Production.ProductReview r

ON r.ProductID = t.ProductID

**Product list with both the products that has a review and the products that does not have a review.**

USE AdventureWorks2016CTP3

SELECT

t.ProductID,

r.ProductReviewID,

r.ReviewerName,

t.TransactionType,

r.comments,

t.ActualCost

FROM Production.TransactionHistory t

FULL OUTER JOIN Production.ProductReview r

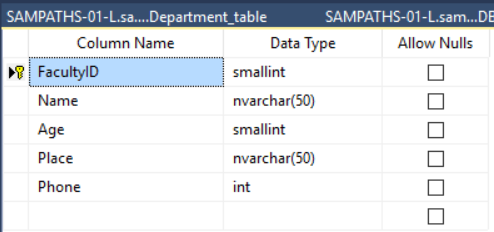
ON r.ProductID = t.ProductID

**JOINS PERFORMED ON 3 TABLES:**

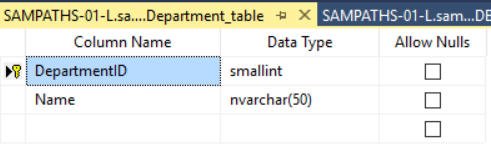
**DATABASE USED:** SAMPATHS-01-L (Local Database)

**SCHEMA DEFINITION**

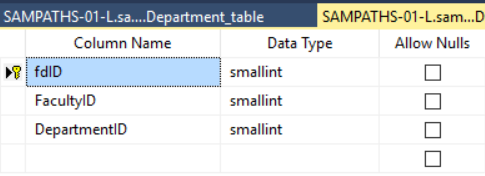
**dbo.Faculty\_Table**

****

**dbo.Department\_Table**

****

**dbo.fb\_Table**

****

**Employees who belong to same locality.**

USE sampleDB

SELECT DISTINCT

f1.FacultyID,

f1.Name,

f1.Age,

f1.Place,

f1.Phone

FROM Faculty\_table f1

INNER JOIN Faculty\_table f2

ON f1.Place = f2.Place

WHERE f1.FacultyID <> f2.FacultyID

ORDER BY f1.Place

**The employee list that holds the employee and the department details**

USE sampleDB

SELECT

f.FacultyID,

f.Name,

fd.DepartmentID,

d.Name

FROM dbo.Faculty\_table f

INNER JOIN dbo.fd\_table fd

ON f.FacultyID = fd.FacultyID

INNER JOIN dbo.Department\_table d

ON d.DepartmentID = fd.DepartmentID

**List of employees who belongs to a certain department and also the employees who haven’t been assigned any department.**

USE sampleDB

SELECT

f.FacultyID,

f.Name,

f.Place,

fd.fdid,

d.DepartmentID,

d.Name AS faculty\_department\_id

FROM dbo.Faculty\_table f

LEFT JOIN dbo.fd\_table fd

ON f.FacultyID = fd.FacultyID

LEFT JOIN dbo.Department\_table d

ON d.DepartmentID = fd.DepartmentID

**List of departments with the details of faculty assigned to that department.**

USE sampleDB

SELECT

f.FacultyID,

f.Name,

f.Place,

d.DepartmentID,

d.Name

FROM dbo.Department\_table d

JOIN dbo.fd\_table fd

ON d.DepartmentID = fd.DepartmentID

LEFT JOIN dbo.Faculty\_table f

ON f.FacultyID = fd.FacultyID