

---

# Acquisition of Adventure Works Announcement

---

Bill Lan, Lillian Lei, Chloe Wang.

## Abstract

Good news everyone! We, Wide World Importers, are happy to announce that we have just brought out a small company called “Adventure Works”! Now that bike shop is our sub-company. The first thing of all works pending would be to merge AdventureWorks user logon information, person information, and products into our database. This is a process that will be a collaborative effort between our Enterprise Data Analytics team and the Data Engineering team over At Adventure Works as Well!

The Data Engineering Team at AdventureWorks has left us a description of their database, storage methods, and detailed documentation of their database. They have even been kind enough to help us merge our databases, leaving their code and documentation so that we can begin to generate new product categories, changes to the way their data is stored, and their customers to become Wide World Importer customers by making an online account!

## 1 Application.People

Most data in the Application.People are updated except columns IsSystemUser, IsEmployee and IsSalesperson. We plan to do it in the future by checking whether the person is an employee or a customer or a vendor. If the person is an employee, IsSystemUser and IsEmployee are likely to be 1, otherwise 0. IsEmployee will be 1 if AW.BusinessEntityID is in the Employee Table, otherwise 0. Likewise, IsSalesperson will be 1 if AW.BusinessEntityID is in the SalesPerson Table.

```
ALTER TABLE WideWorldImporters.Application.People
ALTER COLUMN IsSystemUser DROP NOT NULL;
ALTER TABLE WideWorldImporters.Application.People
ALTER COLUMN IsEmployee DROP NOT NULL;
ALTER TABLE WideWorldImporters.Application.People
ALTER COLUMN IsSalesPerson DROP NOT NULL;

WITH TempTable as (
    SELECT * FROM AdventureWorks2019.Person.Person Person
    LEFT JOIN (
        SELECT * FROM AdventureWorks 2019.Person.EmailAddress
    ) AS PEA
    ON Person.BusinessEntityID = PEA.BusinessEntityID
    LEFT JOIN (
        SELECT * FROM AdventureWorks2019.Person.PersonPhone
    ) PPP ON Person.BusinessEntityID = PPP.BusinessEntityID
)
SELECT
    ROW_NUMBER() + 3261 AS PersonID,
    FirstName AS PreferredName,
    FirstName + ' ' + MiddleName + ' ' + LastName AS FullName,
    FirstName + ' ' + LastName AS SearchName,
```

```

EmailAddress AS LogonName,
PersonPhone AS PhoneNumber,
1 AS IsPermittedToLogon,
1 AS IsExternalLogonProvider ,
NULL AS HashedPassword,
NULL AS IsSystemUser,
NULL AS IsEmployee,
NULL AS IsSalesPerson,
NULL AS UserPreferences,
NULL AS FaxNumber,
EmailAddress AS EmailAddress,
NULL AS Photo,
NULL AS CustomFields,
NULL AS OtherLanguages,
NULL AS LastEditedBy,
GetDate() AS ValidFrom,
'2099/12/31' AS ValidTo
FROM TempTable

INSERT INTO WideWorldImporters.Application.People (
PersonID, FullName, PreferredName,
SearchName, IsPermittedToLogon, LogonName,
IsExternalLogonProvider, HashedPassword, IsSystemUser,
IsEmployee, IsSalesperson, UserPreferences,
PhoneNumber, FaxNumber, EmailAddress,
Photo, CustomFields, OtherLanguages,
LastEditedBy, ValidFrom, ValidTo
);

```

## 1.1 Warehouse.Colors

The Warehouse.Colors Table is up-to-date. We migrated new colors in the AW database and created a new ColorID for each new color. We assume that as the data engineer, we're essentially PersonID 1, so all the values in the LastEditedBy column is set to 1.

```

INSERT INTO WideWorldImporters.Warehouse.Colors
SELECT ROW_NUMBER() OVER() + 36 AS ColorID,
P.Color AS ColorName, 1 AS LastEditedBy,
GetDate() AS ValidFrom,
CAST('12/31/9999 23:59:59.999' as datetime) AS ValidTo
FROM AdventureWorks2019.Production.Product P
WHERE P.Color NOT IN (SELECT C.ColorName
FROM WideWorldImporters.Warehouse.Colors C)

```

## 1.2 Warehouse.StockGroups

For the Warehouse.StockGroup, we updated the database with the AW.ProductCategory table.

```

INSERT INTO WideWorldImporters.Warehouse.StockGroups
SELECT ROW_NUMBER() OVER(ORDER BY C.ProductCategoryID) + 10 AS StockGroupID,
C.Name AS StockGroupName,
1 AS LastEditedBy,
GETDATE() AS ValidFrom,
CAST('12/31/9999 23:59:59.999' as datetime) AS ValidTo
FROM AdventureWorks2019.Production.ProductCategory C
WHERE C.Name NOT IN (SELECT SG.StockGroupName
FROM WideWorldImporters.Warehouse.StockGroups SG)

```

### 1.3 Warehouse.StockItemGroup

For the Warehouse.StockItemStockGroup, we updated the database with the AW.ProductSubcategory table.

```
INSERT INTO WideWorldImporters.Warehouse.StockGroups
SELECT ROW_NUMBER() OVER(ORDER BY C.ProductCategoryID) + 10 AS StockGroupID,
C.Name AS StockGroupName,
1 AS LastEditedBy,
GETDATE() AS ValidFrom,
CAST('12/31/9999 23:59:59.999' as datetime) AS ValidTo
FROM AdventureWorks2019.Production.ProductCategory C
WHERE C.Name NOT IN (SELECT SG.StockGroupName
FROM WideWorldImporters.Warehouse.StockGroups SG)
```

### 1.4 Warehouse.StockItemStockGroup

For the Warehouse.StockItemStockGroup, we updated the database with the AW.ProductSubcategory table.

```
INSERT INTO WideWorldImporters.StockItemStockGroup
SELECT ROW_NUMBER() OVER(ORDER BY C.Name) + 10 AS StockItemStockGroupID,
P.ProductID + 227 AS StockItemID,
C.ProductCategoryID + 10 as StockGroupID,
1 AS LastEditedBy,
CAST('12/31/9999 23:59:59.997' as datetime) AS ValidTo
FROM AdventureWorks2019.Production.Product P
JOIN AdventureWorks2019.Production.ProductSubcategory SC
ON P.ProductSubcategoryID = SC.ProductSubcategoryID
JOIN AdventureWorks2019.Production.ProductCategory C
ON SC.ProductCategoryID = C.ProductCategoryID
ORDER BY P.ProductID
SELECT
    P AS StockItemID,
    ROW_NUMBER() OVER(ORDER BY PI.Product) + 443 AS StockItemGroupID,
    1 AS LastEditedBy,
    '2022/08/04 AS LstEditedWhen
FROM AdventureWorks2019.Production.Product P
JOIN AdventureWorks2019.Productions.ProductSubcategory S
P.ProductionCategoryID = S.ProductCategoryID
```

### 1.5 Warehouse.StockItems

For this stage of data migration, we are only updating the product information for Warehouse.StockItems for the StockItemID, StockItemName, ColorID, as well as ValidFrom and ValidTo. The next stage will be filling in the rest of the columns once we get more information with AdventureWorks.

```
ALTER TABLE WideWorldImporters.Warehouse.StockItems
ALTER COLUMN SupplierID DROP NOT NULL;
ALTER TABLE WideWorldImporters.Warehouse.StockItems
ALTER COLUMN UnitPackageID DROP NOT NULL;
ALTER TABLE WideWorldImporters.Warehouse.StockItems
ALTER COLUMN OuterPackageID DROP NOT NULL;
ALTER TABLE WideWorldImporters.Warehouse.StockItems
ALTER COLUMN LeadTimeDays DROP NOT NULL;
ALTER TABLE WideWorldImporters.Warehouse.StockItems
ALTER COLUMN QuantityPerOuter DROP NOT NULL;
```

```

ALTER TABLE WideWorldImporters.Warehouse.StockItems
ALTER COLUMN IsChillerStock DROP NOT NULL;
ALTER TABLE WideWorldImporters.Warehouse.StockItems
ALTER COLUMN TaxRate DROP NOT NULL;
ALTER TABLE WideWorldImporters.Warehouse.StockItems
ALTER COLUMN UnitPrice DROP NOT NULL;
ALTER TABLE WideWorldImporters.Warehouse.StockItems
ALTER COLUMN SearchDetails DROP NOT NULL;

INSERT INTO WideWorldImporters.Warehouse.StockItems (StockItemID,
    StockItemName, ColorID, LastEditedBy, ValidFrom, ValidTo)
SELECT
APP.ProductID + 227 AS StockItemID,
APP.Name AS StockItemName,
WWC.ColorID,
1 AS LastEditedBy,
GetDate() AS ValidFrom,
CAST('12/31/9999 23:59:59.999' as datetime) AS ValidTo
FROM AdventureWorks2019.Production.Product APP
LEFT JOIN WideWorldImporters.Warehouse.Colors WWC
ON APP.Color = WWC.ColorName

```