/\*

Cleaning Data in SQL

\*/

SELECT \*

FROM PortfolioProject.dbo.NashvilleHousing

--------------------------------------------------------------------------------------------------------------------------

-- Standardize Date Format

SELECT SaleDate

FROM PortfolioProject.dbo.NashvilleHousing

SELECT SaleDate, CONVERT(DATE,SaleDate)

FROM PortfolioProject.dbo.NashvilleHousing

ALTER TABLE PortfolioProject.dbo.NashvilleHousing

ADD SaleDateConverted DATE

UPDATE PortfolioProject.dbo.NashvilleHousing

SET SaleDateConverted = CONVERT(DATE,SaleDate)

--------------------------------------------------------------------------------------------------------------------------

-- Populate Property Address data

/\*

We have found out that PropertyAddress column has some NULL values

\*/

SELECT \*

FROM PortfolioProject.dbo.NashvilleHousing

WHERE PropertyAddress IS NULL

ORDER BY ParcelID

/\*

PropertyAddress will be populated from the matching ParcelID

If the UniqueID is different and ParcelID is same then PropertyAddress will be populated

\*/

SELECT a.ParcelID, a.PropertyAddress, b.ParcelID, b.PropertyAddress, ISNULL(a.PropertyAddress,b.PropertyAddress)

FROM PortfolioProject.dbo.NashvilleHousing a

JOIN PortfolioProject.dbo.NashvilleHousing b

ON a.ParcelID = b.ParcelID

AND a.[UniqueID ] <> b.[UniqueID ]

WHERE a.PropertyAddress IS NULL

Update a

SET PropertyAddress = ISNULL(a.PropertyAddress,b.PropertyAddress)

FROM PortfolioProject.dbo.NashvilleHousing a

JOIN PortfolioProject.dbo.NashvilleHousing b

ON a.ParcelID = b.ParcelID

AND a.[UniqueID ] <> b.[UniqueID ]

WHERE a.PropertyAddress IS NULL

--------------------------------------------------------------------------------------------------------------------------

-- Breaking out Address into Individual Columns (Address, City, State)

/\*

Breaking out Property Address

\*/

SELECT PropertyAddress

FROM PortfolioProject.dbo.NashvilleHousing

WHERE PropertyAddress is null

ORDER BY ParcelID

SELECT

SUBSTRING(PropertyAddress, 1, CHARINDEX(',', PropertyAddress) -1) as Address

, SUBSTRING(PropertyAddress, CHARINDEX(',', PropertyAddress) +1 , LEN(PropertyAddress)) as Address

FROM PortfolioProject.dbo.NashvilleHousing

ALTER TABLE PortfolioProject.dbo.NashvilleHousing

ADD PropertySplitAddress NVARCHAR(255);

UPDATE PortfolioProject.dbo.NashvilleHousing

SET PropertySplitAddress = SUBSTRING(PropertyAddress, 1, CHARINDEX(',', PropertyAddress) -1 )

ALTER TABLE PortfolioProject.dbo.NashvilleHousing

ADD PropertySplitCity NVARCHAR(255);

UPDATE PortfolioProject.dbo.NashvilleHousing

SET PropertySplitCity = SUBSTRING(PropertyAddress, CHARINDEX(',', PropertyAddress) + 1 , LEN(PropertyAddress))

SELECT \*

FROM PortfolioProject.dbo.NashvilleHousing

/\*

Breaking out Owner Address

\*/

SELECT OwnerAddress

FROM PortfolioProject.dbo.NashvilleHousing

SELECT

PARSENAME(REPLACE(OwnerAddress, ',', '.') , 3)

,PARSENAME(REPLACE(OwnerAddress, ',', '.') , 2)

,PARSENAME(REPLACE(OwnerAddress, ',', '.') , 1)

FROM PortfolioProject.dbo.NashvilleHousing

ALTER TABLE PortfolioProject.dbo.NashvilleHousing

ADD OwnerSplitAddress NVARCHAR(255)

Update PortfolioProject.dbo.NashvilleHousing

SET OwnerSplitAddress = PARSENAME(REPLACE(OwnerAddress, ',', '.') , 3)

ALTER TABLE PortfolioProject.dbo.NashvilleHousing

ADD OwnerSplitCity NVARCHAR(255)

UPDATE PortfolioProject.dbo.NashvilleHousing

SET OwnerSplitCity = PARSENAME(REPLACE(OwnerAddress, ',', '.') , 2)

ALTER TABLE PortfolioProject.dbo.NashvilleHousing

ADD OwnerSplitState NVARCHAR(255)

UPDATE PortfolioProject.dbo.NashvilleHousing

SET OwnerSplitState = PARSENAME(REPLACE(OwnerAddress, ',', '.') , 1)

SELECT \*

FROM PortfolioProject.dbo.NashvilleHousing

--------------------------------------------------------------------------------------------------------------------------

-- Change Y and N to Yes and No in "Sold as Vacant" field

SELECT DISTINCT(SoldAsVacant), Count(SoldAsVacant)

FROM PortfolioProject.dbo.NashvilleHousing

GROUP BY SoldAsVacant

ORDER BY 2

SELECT SoldAsVacant

, CASE WHEN SoldAsVacant = 'Y' THEN 'Yes'

WHEN SoldAsVacant = 'N' THEN 'No'

ELSE SoldAsVacant

END

FROM PortfolioProject.dbo.NashvilleHousing

Update PortfolioProject.dbo.NashvilleHousing

SET SoldAsVacant = CASE WHEN SoldAsVacant = 'Y' THEN 'Yes'

WHEN SoldAsVacant = 'N' THEN 'No'

ELSE SoldAsVacant

END

-----------------------------------------------------------------------------------------------------------------------------------------------------------

-- Remove Duplicates

WITH RowNumCTE AS(

SELECT \*,

ROW\_NUMBER() OVER (

PARTITION BY ParcelID,

PropertyAddress,

SalePrice,

SaleDate,

LegalReference

ORDER BY

UniqueID

) row\_num

FROM PortfolioProject.dbo.NashvilleHousing

--order by ParcelID

)

SELECT \*

--DELETE

FROM RowNumCTE

WHERE row\_num > 1

ORDER BY PropertyAddress

SELECT \*

FROM PortfolioProject.dbo.NashvilleHousing

---------------------------------------------------------------------------------------------------------

-- Delete Unused Columns

SELECT \*

FROM PortfolioProject.dbo.NashvilleHousing

ALTER TABLE PortfolioProject.dbo.NashvilleHousing

DROP COLUMN OwnerAddress, TaxDistrict, PropertyAddress, SaleDate