

/\*

Cleaning Data in SQL Queries

\*/

Select \*

From PortfolioProject.dbo.NashvilleHousing

---

-- Standardize Date Format

Select saleDateConverted, CONVERT(Date,SaleDate)

From PortfolioProject.dbo.NashvilleHousing

Update NashvilleHousing

SET SaleDate = CONVERT(Date,SaleDate)

-- If it doesn't Update properly

ALTER TABLE NashvilleHousing

Add SaleDateConverted Date;

Update NashvilleHousing

SET SaleDateConverted = CONVERT(Date,SaleDate)

---

-- Populate Property Address data

Select \*

From PortfolioProject.dbo.NashvilleHousing

--Where PropertyAddress is null

order by ParcelID

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)

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 NashvilleHousing

Add PropertySplitAddress Nvarchar(255);

Update NashvilleHousing

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

ALTER TABLE NashvilleHousing

Add PropertySplitCity Nvarchar(255);

Update NashvilleHousing

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

Select \*

From PortfolioProject.dbo.NashvilleHousing

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 NashvilleHousing

Add OwnerSplitAddress Nvarchar(255);

Update NashvilleHousing

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

ALTER TABLE NashvilleHousing

Add OwnerSplitCity Nvarchar(255);

Update NashvilleHousing

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

ALTER TABLE NashvilleHousing

Add OwnerSplitState Nvarchar(255);

Update 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 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 \*

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