/\*

Step 1.Cleaning Data in SQL queries

\*/

SELECT \*

FROM NashvilleHousing;

/\*

Step 2. Standardize Date format

\*/

SELECT SaleDate

FROM NashvilleHousing;

Select SaleDate, CONVERT(Date,SaleDate)

from dbo.NashvilleHousing

UPDATE NashvilleHousing

SET SaleDate = CONVERT(Date,SaleDate)

ALTER TABLE NashvilleHousing

Add SaleDateConverted Date;

UPDATE NashvilleHousing

SET SaleDateConverted = CONVERT(Date,SaleDate)

SELECT SaleDateConverted

From dbo.NashvilleHousing

/\*

Step 3. Property Address (Populate Null value)

\*/

Select \*

From [Project Porfolio].dbo.NashvilleHousing

where PropertyAddress Is Null

order by ParcelID

Select a.ParcelID, a.PropertyAddress, b.ParcelID, b.PropertyAddress, isnull(a.PropertyAddress, b.PropertyAddress)

From [Project Porfolio].dbo.NashvilleHousing a

JOIN [Project Porfolio].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 [Project Porfolio].dbo.NashvilleHousing a

JOIN [Project Porfolio].dbo.NashvilleHousing b

on a.ParcelID = b.ParcelID

and a.UniqueID <> b.UniqueID

where a.PropertyAddress is null

/\*

Step 4. Address breakdown (Address, City, State)

\*/

Select PropertyAddress

From [Project Porfolio].dbo.NashvilleHousing

order by ParcelID

Select

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

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

from [Project Porfolio].dbo.NashvilleHousing

ALTER TABLE NashvilleHousing

Add PropertySplit\_Address Nvarchar(255);

Update NashvilleHousing

Set PropertySplit\_Address = SUBSTRING(PropertyAddress, 1,CHARINDEX(',',PropertyAddress)-1)

ALTER TABLE NashvilleHousing

Add PropertySplit\_City Nvarchar(255);

Update NashvilleHousing

Set PropertySplit\_City = SUBSTRING(PropertyAddress, CHARINDEX(',',PropertyAddress)+1, lEN(PropertyAddress))

SELECT \*

FROM NashvilleHousing;

/\* Step 5: PARSENAME - Owner Address Cleansing\*/

SELECT

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

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

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

FROM NashvilleHousing;

ALTER TABLE NashvilleHousing

ADD OwnerSplit\_Address NVARCHAR (255);

UPDATE NashvilleHousing

SET OwnerSplit\_Address = PARSENAME(REPLACE(OwnerAddress, ',', '.'), 3)

ALTER TABLE NashvilleHousing

ADD OwnerSplit\_City NVARCHAR (255);

UPDATE NashvilleHousing

SET OwnerSplit\_City = PARSENAME(REPLACE(OwnerAddress, ',', '.'), 2)

ALTER TABLE NashvilleHousing

ADD OwnerSplit\_State NVARCHAR (255);

UPDATE NashvilleHousing

SET OwnerSplit\_State = PARSENAME(REPLACE(OwnerAddress, ',', '.'), 1)

SELECT \*

FROM NashvilleHousing;

/\*Step 6: Cleansing Change Y/N to Yes/No \*/

SELECT DISTINCT SoldAsVacant, COUNT (SoldAsVacant)

FROM NashvilleHousing

GROUP BY SoldAsVacant

ORDER BY 2;

SELECT SoldAsVacant

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

WHEN SoldAsVacant = 'N' THEN 'No'

ELSE SoldAsVacant

END

FROM NashvilleHousing;

UPDATE NashvilleHousing

SET SoldAsVacant =

CASE WHEN SoldAsVacant = 'Y' THEN 'Yes'

WHEN SoldAsVacant = 'N' THEN 'No'

ELSE SoldAsVacant

END

/\* Step 7: Remove Duplicates \*/

WITH RowNumCTE AS (

SELECT \*,

ROW\_NUMBER () OVER (

PARTITION BY ParcelID,

SalePrice,

SaleDate,

LegalReference

ORDER BY UniqueID

) row\_num

FROM NashvilleHousing

)

DELETE

FROM RowNumCTE

WHERE row\_num >1

/\* Delete Unused Columns \*/

SELECT \*

FROM NashvilleHousing

ALTER TABLE NashvilleHousing

DROP COLUMN OwnerAddress, PropertyAddress, TaxDistrict, SaleDate