Select \*

from [part 3].dbo.NashvilleHousing

-- sale date

Select SaleDate

from [part 3].dbo.NashvilleHousing

Select Saledatecoverted, CONVERT(Date, Saledate)

from [part 3].dbo.NashvilleHousing

Update NashvilleHousing

set Saledate = CONVERT(Date, Saledate)

Alter table NashvilleHousing

add Saledatecoverted Date;

Update NashvilleHousing

set Saledatecoverted = CONVERT(Date, Saledate)

-- Property address

Select PropertyAddress

from [part 3].dbo.NashvilleHousing

where PropertyAddress is null

Select \*

from [part 3].dbo.NashvilleHousing

where PropertyAddress is null

Select \*

from [part 3].dbo.NashvilleHousing

--where PropertyAddress is null

order by ParcelID

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

from [part 3].dbo.NashvilleHousing a

join [part 3].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 [part 3].dbo.NashvilleHousing a

join [part 3].dbo.NashvilleHousing b

on a.ParcelID = b.ParcelID

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

where a.PropertyAddress is null

-- breaking address

Select PropertyAddress

from [part 3].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 [part 3].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 [part 3].dbo.NashvilleHousing

-- owner address

Select OwnerAddress

from [part 3].dbo.NashvilleHousing

Select

PARSENAME(replace(Owneraddress, ',' , '.'), 3)

,PARSENAME(replace(Owneraddress, ',' , '.'), 2)

,PARSENAME(replace(Owneraddress, ',' , '.'), 1)

from [part 3].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 [part 3].dbo.NashvilleHousing

-- change y and n to yes and no in sold vacant

select Distinct (SoldAsVacant), Count(SoldAsVacant)

from [part 3].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 [part 3].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 [part 3].dbo.NashvilleHousing

)

--Order by ParcelID

Select \*

from RownumCTE

where Row\_num > 1

order by PropertyAddress

--Delete

With RownumCTE AS(

select \*,

ROW\_NUMBER() Over (

Partition by ParcelID,

PropertyAddress,

SalePrice,

SaleDate,

LegalReference

Order by

UniqueId

) Row\_num

from [part 3].dbo.NashvilleHousing

)

--Order by ParcelID

Delete

from RownumCTE

where Row\_num > 1

--order by PropertyAddress

-- delete columns

Select \*

from [part 3].dbo.NashvilleHousing

--Alter table NashvilleHousing

--Drop Column Owneraddress, taxedistrict, Propertyaddress,saledate