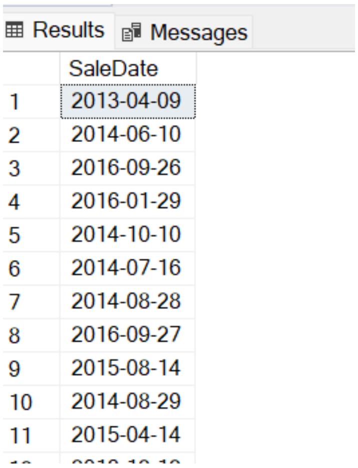
NASHVILLE HOUSING DATA

SQL PROJECT FOR DATA CLEANING

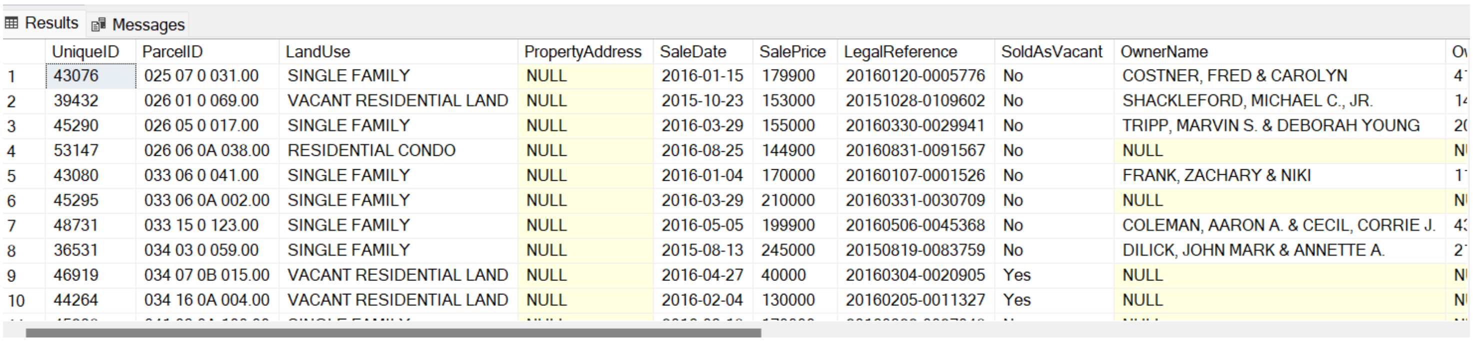
BY GAURAV CHANDRA

```
--here original data had the saledate column as datetime datatype ,so I changed the datatype to date
--to change datetime into date for saledate column
alter table dbo.housingdata alter column SaleDate Date;

select  SaleDate from dbo.housingdata;
```



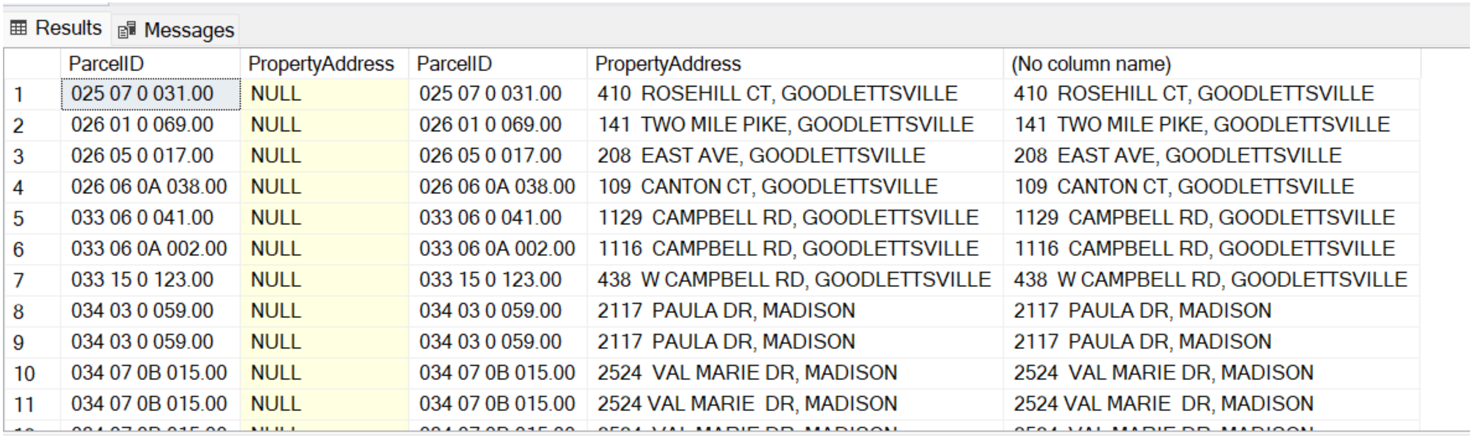
	SaleDate
1	2013-04-09
2	2014-06-10
3	2016-09-26
4	2016-01-29
5	2014-10-10
6	2014-07-16
7	2014-08-28
8	2016-09-27
9	2015-08-14
10	2014-08-29
11	2015-04-14

```
--In our dataset, PropertyAddress column had nulls evenif it has a valid parcelID ,
--To fix this,I noticed that a parcelID is unique to the PropertyAddress, so if two rows have different uniqueID but it has same parcelID,
--then both must have the same Address too.
select* from dbo.housingdata
where PropertyAddress is null
order by ParcelID
```



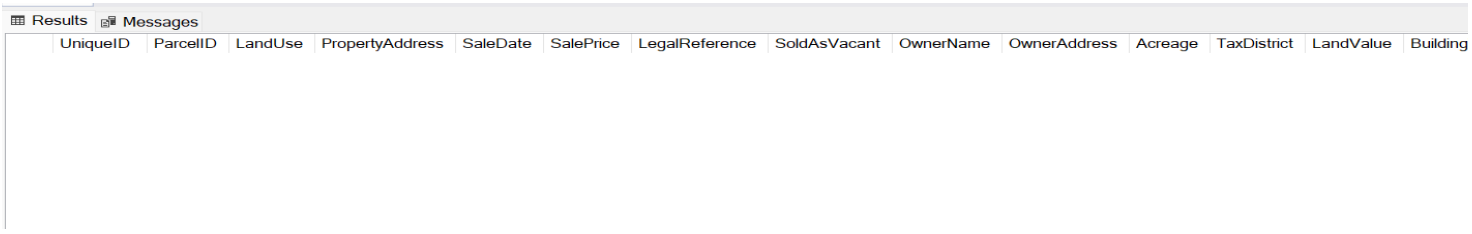
	UniqueID	ParcelID	LandUse	PropertyAddress	SaleDate	SalePrice	LegalReference	SoldAsVacant	OwnerName	Ow
1	43076	025 07 0 031.00	SINGLE FAMILY	NULL	2016-01-15	179900	20160120-0005776	No	COSTNER, FRED & CAROLYN	4
2	39432	026 01 0 069.00	VACANT RESIDENTIAL LAND	NULL	2015-10-23	153000	20151028-0109602	No	SHACKLEFORD, MICHAEL C., JR.	14
3	45290	026 05 0 017.00	SINGLE FAMILY	NULL	2016-03-29	155000	20160330-0029941	No	TRIPP, MARVIN S. & DEBORAH YOUNG	2
4	53147	026 06 0A 038.00	RESIDENTIAL CONDO	NULL	2016-08-25	144900	20160831-0091567	No	NULL	N
5	43080	033 06 0 041.00	SINGLE FAMILY	NULL	2016-01-04	170000	20160107-0001526	No	FRANK, ZACHARY & NIKI	1
6	45295	033 06 0A 002.00	SINGLE FAMILY	NULL	2016-03-29	210000	20160331-0030709	No	NULL	N
7	48731	033 15 0 123.00	SINGLE FAMILY	NULL	2016-05-05	199900	20160506-0045368	No	COLEMAN, AARON A. & CECIL, CORRIE J.	4
8	36531	034 03 0 059.00	SINGLE FAMILY	NULL	2015-08-13	245000	20150819-0083759	No	DILICK, JOHN MARK & ANNETTE A.	2
9	46919	034 07 0B 015.00	VACANT RESIDENTIAL LAND	NULL	2016-04-27	40000	20160304-0020905	Yes	NULL	N
10	44264	034 16 0A 004.00	VACANT RESIDENTIAL LAND	NULL	2016-02-04	130000	20160205-0011327	Yes	NULL	N

```
--lets now select the rows with different uniqueID but same parcelID and if one of them has null
propertyAddress,
--we'd change it into the corresponding address

select h1.ParcelID,h1.PropertyAddress ,
h2.ParcelID,h2.PropertyAddress,isnull(h1.PropertyAddress,h2.PropertyAddress)
from dbo.housingdata h1 join
dbo.housingdata h2
on h1.ParcelID = h2.ParcelID
and h1.UniqueID <> h2.[UniqueID ]
where h1.PropertyAddress is null
order by h1.ParcelID
```



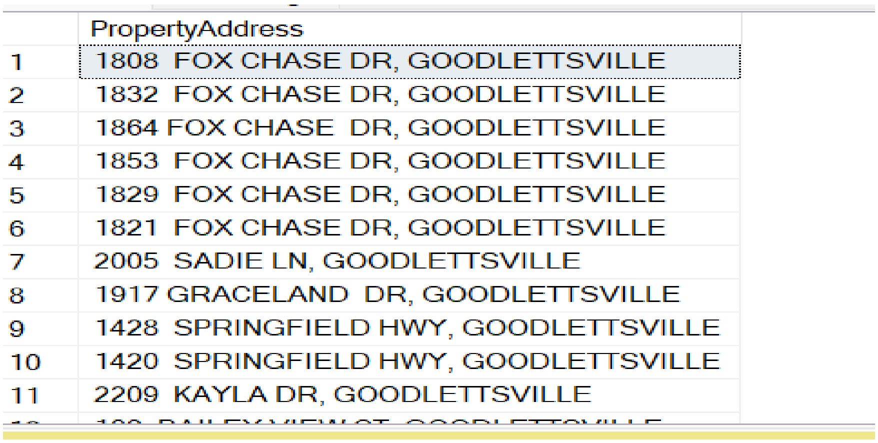
	ParcelID	PropertyAddress	ParcelID	PropertyAddress	(No column name)
1	025 07 0 031.00	NULL	025 07 0 031.00	410 ROSEHILL CT, GOODLETTSVILLE	410 ROSEHILL CT, GOODLETTSVILLE
2	026 01 0 069.00	NULL	026 01 0 069.00	141 TWO MILE PIKE, GOODLETTSVILLE	141 TWO MILE PIKE, GOODLETTSVILLE
3	026 05 0 017.00	NULL	026 05 0 017.00	208 EAST AVE, GOODLETTSVILLE	208 EAST AVE, GOODLETTSVILLE
4	026 06 0A 038.00	NULL	026 06 0A 038.00	109 CANTON CT, GOODLETTSVILLE	109 CANTON CT, GOODLETTSVILLE
5	033 06 0 041.00	NULL	033 06 0 041.00	1129 CAMPBELL RD, GOODLETTSVILLE	1129 CAMPBELL RD, GOODLETTSVILLE
6	033 06 0A 002.00	NULL	033 06 0A 002.00	1116 CAMPBELL RD, GOODLETTSVILLE	1116 CAMPBELL RD, GOODLETTSVILLE
7	033 15 0 123.00	NULL	033 15 0 123.00	438 W CAMPBELL RD, GOODLETTSVILLE	438 W CAMPBELL RD, GOODLETTSVILLE
8	034 03 0 059.00	NULL	034 03 0 059.00	2117 PAULA DR, MADISON	2117 PAULA DR, MADISON
9	034 03 0 059.00	NULL	034 03 0 059.00	2117 PAULA DR, MADISON	2117 PAULA DR, MADISON
10	034 07 0B 015.00	NULL	034 07 0B 015.00	2524 VAL MARIE DR, MADISON	2524 VAL MARIE DR, MADISON
11	034 07 0B 015.00	NULL	034 07 0B 015.00	2524 VAL MARIE DR, MADISON	2524 VAL MARIE DR, MADISON

```
UPDATE h1
set h1.PropertyAddress = isnull(h1.PropertyAddress,h2.PropertyAddress)
from dbo.housingdata h1 join
dbo.housingdata h2
on h1.ParcelID = h2.ParcelID
and h1.UniqueID <> h2.[UniqueID ]
where h1.PropertyAddress is null

select * from dbo.housing
where PropertyAddress is null

--we can notice there are no nulls addresses left
```



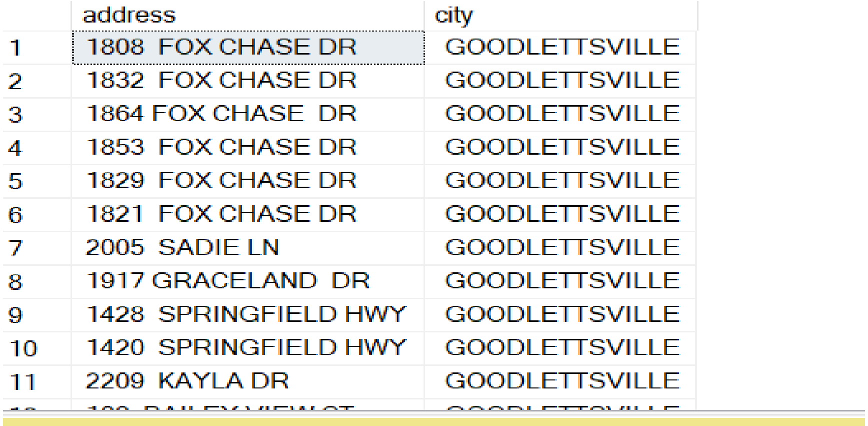
	UniqueID	ParcelID	LandUse	PropertyAddress	SaleDate	SalePrice	LegalReference	SoldAsVacant	OwnerName	OwnerAddress	Acreage	TaxDistrict	LandValue	Building

```
--Breaking out Address into Individual Columns (Address,City ,State)
select PropertyAddress from dbo.housing
```



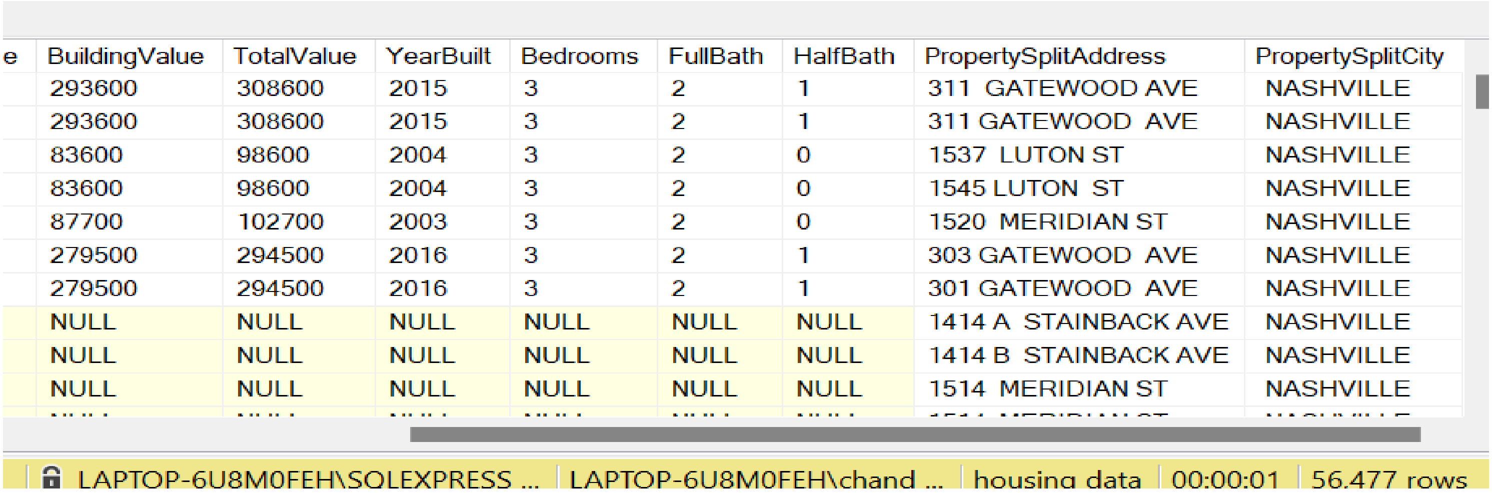
	PropertyAddress
1	1808 FOX CHASE DR, GOODLETTSVILLE
2	1832 FOX CHASE DR, GOODLETTSVILLE
3	1864 FOX CHASE DR, GOODLETTSVILLE
4	1853 FOX CHASE DR, GOODLETTSVILLE
5	1829 FOX CHASE DR, GOODLETTSVILLE
6	1821 FOX CHASE DR, GOODLETTSVILLE
7	2005 SADIE LN, GOODLETTSVILLE
8	1917 GRACELAND DR, GOODLETTSVILLE
9	1428 SPRINGFIELD HWY, GOODLETTSVILLE
10	1420 SPRINGFIELD HWY, GOODLETTSVILLE
11	2209 KAYLA DR, GOODLETTSVILLE

```
--Here I have selected the 2 substrings of PropertyAddress separated by commas for address and city of
property
select substring(PropertyAddress,1,charindex(',',PropertyAddress)-1) as address
,substring(PropertyAddress,charindex(',',PropertyAddress)+1,len(PropertyAddress)) as city
from dbo.housingdata
```



	address	city
1	1808 FOX CHASE DR	GOODLETTSVILLE
2	1832 FOX CHASE DR	GOODLETTSVILLE
3	1864 FOX CHASE DR	GOODLETTSVILLE
4	1853 FOX CHASE DR	GOODLETTSVILLE
5	1829 FOX CHASE DR	GOODLETTSVILLE
6	1821 FOX CHASE DR	GOODLETTSVILLE
7	2005 SADIE LN	GOODLETTSVILLE
8	1917 GRACELAND DR	GOODLETTSVILLE
9	1428 SPRINGFIELD HWY	GOODLETTSVILLE
10	1420 SPRINGFIELD HWY	GOODLETTSVILLE
11	2209 KAYLA DR	GOODLETTSVILLE

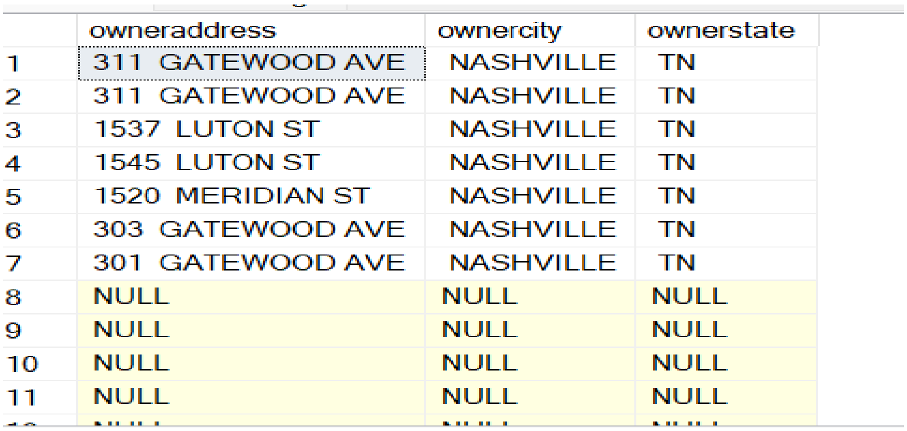
```
--Now , lets add two new columns for splitted-address and splitted-city and update these columns with the
substrings
alter table dbo.housingdata
add PropertySplitAddress nvarchar(255)

update dbo.housingdata
set PropertySplitAddress = substring(PropertyAddress,1,charindex(',',PropertyAddress)-1)

alter table dbo.housingdata
add PropertySplitCity nvarchar(255)

update dbo.housingdata
set PropertySplitCity = substring(PropertyAddress,charindex(',',PropertyAddress)+1,len(PropertyAddress))


select * from dbo.housingdata

--we can see that two columns have been added in the end of table.
```



e	BuildingValue	TotalValue	YearBuilt	Bedrooms	FullBath	HalfBath	PropertySplitAddress	PropertySplitCity
	293600	308600	2015	3	2	1	311 GATEWOOD AVE	NASHVILLE
	293600	308600	2015	3	2	1	311 GATEWOOD AVE	NASHVILLE
	83600	98600	2004	3	2	0	1537 LUTON ST	NASHVILLE
	83600	98600	2004	3	2	0	1545 LUTON ST	NASHVILLE
	87700	102700	2003	3	2	0	1520 MERIDIAN ST	NASHVILLE
	279500	294500	2016	3	2	1	303 GATEWOOD AVE	NASHVILLE
	279500	294500	2016	3	2	1	301 GATEWOOD AVE	NASHVILLE
	NULL	NULL	NULL	NULL	NULL	NULL	1414 A STAINBACK AVE	NASHVILLE
	NULL	NULL	NULL	NULL	NULL	NULL	1414 B STAINBACK AVE	NASHVILLE
	NULL	NULL	NULL	NULL	NULL	NULL	1514 MERIDIAN ST	NASHVILLE

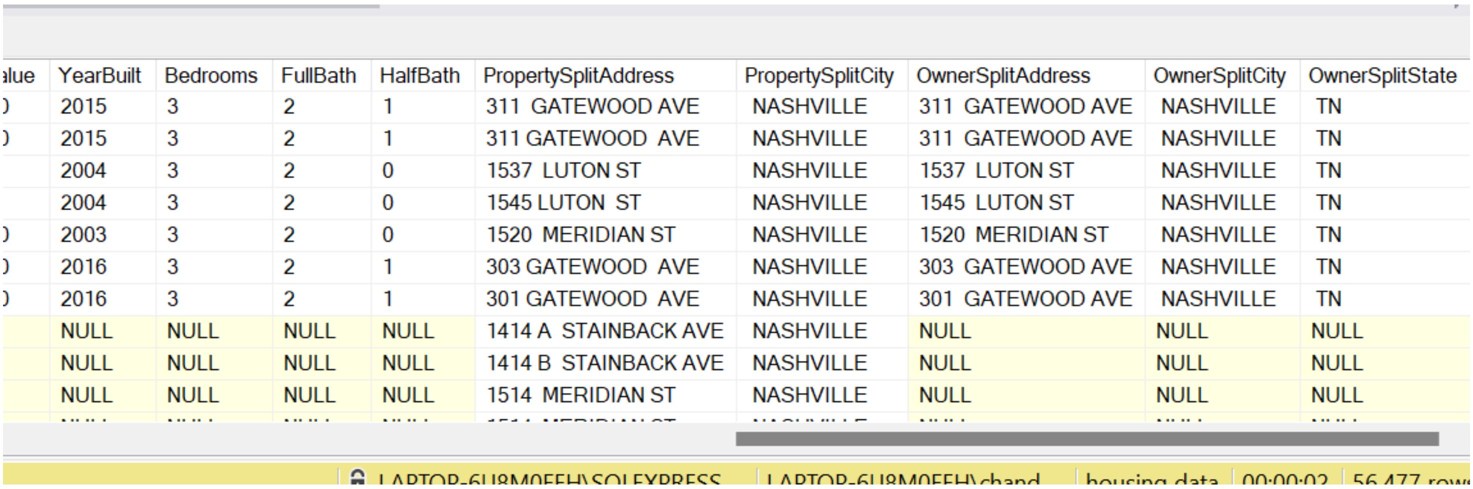
LAPTOP-6U8M0FEH\SQLEXPRESS ... | LAPTOP-6U8M0FEH\chand ... | housing data | 00:00:01 | 56,477 rows

```
--Here again,I created splitted address and city column from the OwnerAddress column,but using a simpler
method i.e parsename.
select parsename(replace(OwnerAddress,',','.'),3) as owneraddress,
parsename(replace(OwnerAddress,',','.'),2) as ownercity,
parsename(replace(OwnerAddress,',','.'),1) as ownerstate
from dbo.housingdata
```



	owneraddress	ownercity	ownerstate
1	311 GATEWOOD AVE	NASHVILLE	TN
2	311 GATEWOOD AVE	NASHVILLE	TN
3	1537 LUTON ST	NASHVILLE	TN
4	1545 LUTON ST	NASHVILLE	TN
5	1520 MERIDIAN ST	NASHVILLE	TN
6	303 GATEWOOD AVE	NASHVILLE	TN
7	301 GATEWOOD AVE	NASHVILLE	TN
8	NULL	NULL	NULL
9	NULL	NULL	NULL
10	NULL	NULL	NULL
11	NULL	NULL	NULL

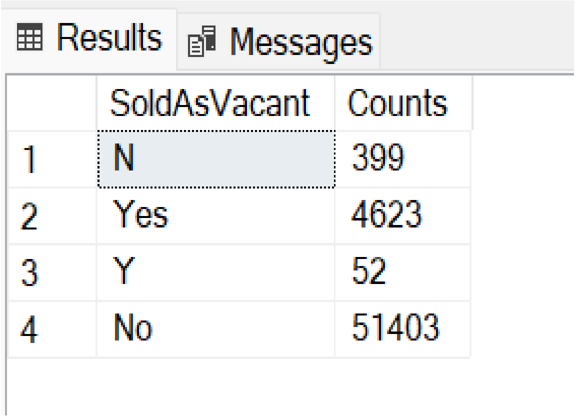
```
--Now , lets add three new columns for splitted-address , splitted-city and splitted-state & update these
columns with the substrings
alter table dbo.housingdata
add OwnerSplitAddress nvarchar(255)

update dbo.housingdata
set OwnerSplitAddress = parsename(replace(OwnerAddress,',','.'),3)


alter table dbo.housingdata
add OwnerSplitCity nvarchar(255)

update dbo.housingdata
set OwnerSplitCity = parsename(replace(OwnerAddress,',','.'),2)

alter table dbo.housingdata
add OwnerSplitState nvarchar(255)

update dbo.housingdata
set OwnerSplitState = parsename(replace(OwnerAddress,',','.'),1)


select * from dbo.housingdata

--here we can see that three new columns have been added in the end of the table.
```



alue	YearBuilt	Bedrooms	FullBath	HalfBath	PropertySplitAddress	PropertySplitCity	OwnerSplitAddress	OwnerSplitCity	OwnerSplitState
)	2015	3	2	1	311 GATEWOOD AVE	NASHVILLE	311 GATEWOOD AVE	NASHVILLE	TN
)	2015	3	2	1	311 GATEWOOD AVE	NASHVILLE	311 GATEWOOD AVE	NASHVILLE	TN
	2004	3	2	0	1537 LUTON ST	NASHVILLE	1537 LUTON ST	NASHVILLE	TN
	2004	3	2	0	1545 LUTON ST	NASHVILLE	1545 LUTON ST	NASHVILLE	TN
)	2003	3	2	0	1520 MERIDIAN ST	NASHVILLE	1520 MERIDIAN ST	NASHVILLE	TN
)	2016	3	2	1	303 GATEWOOD AVE	NASHVILLE	303 GATEWOOD AVE	NASHVILLE	TN
)	2016	3	2	1	301 GATEWOOD AVE	NASHVILLE	301 GATEWOOD AVE	NASHVILLE	TN
	NULL	NULL	NULL	NULL	1414 A STAINBACK AVE	NASHVILLE	NULL	NULL	NULL
	NULL	NULL	NULL	NULL	1414 B STAINBACK AVE	NASHVILLE	NULL	NULL	NULL
	NULL	NULL	NULL	NULL	1514 MERIDIAN ST	NASHVILLE	NULL	NULL	NULL

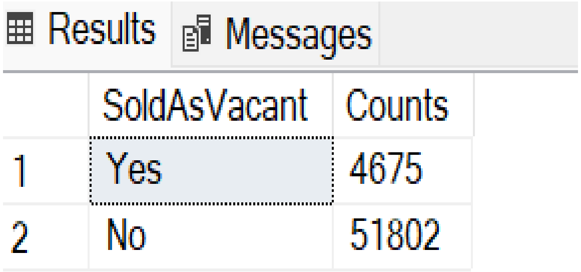
LAPTOP-6U8M0FEH\SQLEXPRESS | LAPTOP-6U8M0FEH\chand | housing data | 00:00:02 | 56,477 rows

```sql
--In our data set, column SoldAsVacant had some mixed values ,So we updated all the Ys and Ns to Yes and
No respectively.
select distinct(SoldAsVacant),count(SoldAsVacant) as Counts from dbo.housingdata
group by SoldAsVacant
```



	SoldAsVacant	Counts
1	N	399
2	Yes	4623
3	Y	52
4	No	51403

```sql
select SoldAsVacant ,
case SoldAsVacant
when 'Y' then 'Yes'
when 'N' then 'No'
else SoldAsVacant
end
from dbo.housingdata


update dbo.housingdata
set SoldAsVacant =
case SoldAsVacant
when 'Y' then 'Yes'
when 'N' then 'No'
else SoldAsVacant
end

select distinct(SoldAsVacant),count(SoldAsVacant) as Counts from dbo.housingdata
group by SoldAsVacant

--here we can see that we now only have Yes and No in SoldAsVacant column
```



	SoldAsVacant	Counts
1	Yes	4675
2	No	51802