Walkthrough: Wauwatosa Data Cleaning Project

In this project, I used Microsoft SQL Server to clean data that was scraped from Zillow for home sales in Wauwatosa, WI between February 2021 and March 2023.

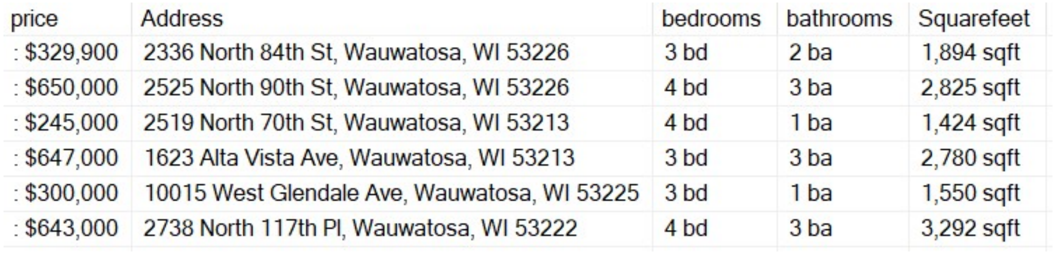
For full project and code, please see: https://github.com/bebenjam/Wauwatosa_Data_Cleaning_SQL

Functions and Skills Used

UNION	CTEs
Window Functions	Removing Duplicates
SUBSTRING	LEN
CHARINDEX	ALTER and UPDATE Tables
PARSENAME	CAST
TRY_CAST	REPLACE
WHERE	CREATE VIEW
PATINDEX	

The Data

The dirty data consisted of strings (VARCHAR) that contained extra characters, letters, etc. that were not formatted well for further analysis.



price	Address	bedrooms	bathrooms	Squarefeet
: $329,900	2336 North 84th St, Wauwatosa, WI 53226	3 bd	2 ba	1,894 sqft
: $650,000	2525 North 90th St, Wauwatosa, WI 53226	4 bd	3 ba	2,825 sqft
: $245,000	2519 North 70th St, Wauwatosa, WI 53213	4 bd	1 ba	1,424 sqft
: $647,000	1623 Alta Vista Ave, Wauwatosa, WI 53213	3 bd	3 ba	2,780 sqft
: $300,000	10015 West Glendale Ave, Wauwatosa, WI 53225	3 bd	1 ba	1,550 sqft
: $643,000	2738 North 117th Pl, Wauwatosa, WI 53222	4 bd	3 ba	3,292 sqft

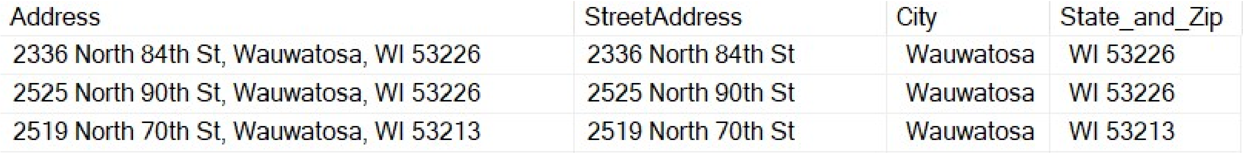
The objective was to clean each column for better analysis. For example, the "Address" column would be separated into address number, directional and street name, city, state, and ZIP code columns.

Code Example

Here is an example of the code to extract components from the "Address" column:

```sql
 SELECT Address,
        SUBSTRING (Address, 1, CHARINDEX(',',Address) -1) AS StreetAddress,
        PARSENAME(REPLACE(Address, ',', '.'), 2) AS City,
        PARSENAME(REPLACE(Address, ',', '.'), 1) AS State_and_Zip
FROM Wauwatosa_Housing_Market.dbo.Wauwa_Sold_2021_2023
```

And the result:

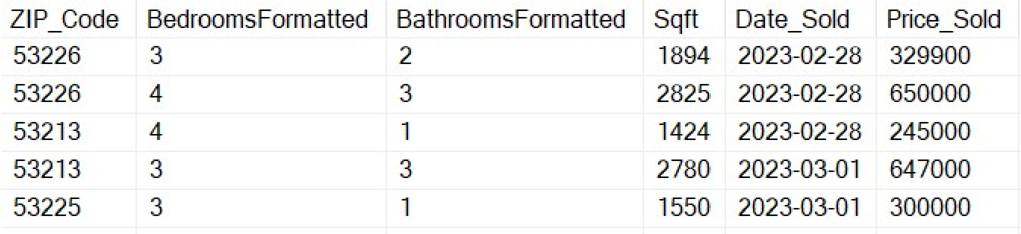


Address	StreetAddress	City	State_and_Zip
2336 North 84th St, Wauwatosa, WI 53226	2336 North 84th St	Wauwatosa	WI 53226
2525 North 90th St, Wauwatosa, WI 53226	2525 North 90th St	Wauwatosa	WI 53226
2519 North 70th St, Wauwatosa, WI 53213	2519 North 70th St	Wauwatosa	WI 53213

This step used SUBSTRING and PARSENAME to extract information between the comma delimiter. This would require further parsing.

__Project Outcome__

At the end of the project, we have separate columns for many of the components seen below. These formats can now be queried upon for further analysis including characteristics of houses sold in each ZIP code, bedroom and bathroom combinations, when they were sold, and trends in prices.



ZIP_Code	BedroomsFormatted	BathroomsFormatted	Sqft	Date_Sold	Price_Sold
53226	3	2	1894	2023-02-28	329900
53226	4	3	2825	2023-02-28	650000
53213	4	1	1424	2023-02-28	245000
53213	3	3	2780	2023-03-01	647000
53225	3	1	1550	2023-03-01	300000

This table was used for Tableau visualizations, see here:

https://public.tableau.com/app/profile/ben.beutler/viz/WauwatosaRecentlySold-April2023/Dashboard1

__Project Lessons Learned__

- PARSENAME and REPLACE used together can be powerful! SUBSTRING had some limitations for me in trying to extract the "Address" column mostly because street numbers, names, and city lengths were all different.
- TRY_CAST was useful in pushing through table updates for columns I was changing to INT data type. Some records contained strings label as a string 'null'. Instead of parsing out or excluding those records, I could transform them into NULL and keep other information from the record.
- This was overall a fun project! I had to look up different ways to extract data with CHARINDEX and PATINDEX. I had to get creative using the WHERE and LIKE statements in order to parse the correct data.
- I was able to create a clean view from dirty data that was scraped from Zillow that could be used for further data analysis on homes sold in Wauwatosa, WI between February 2021 and March 2023.