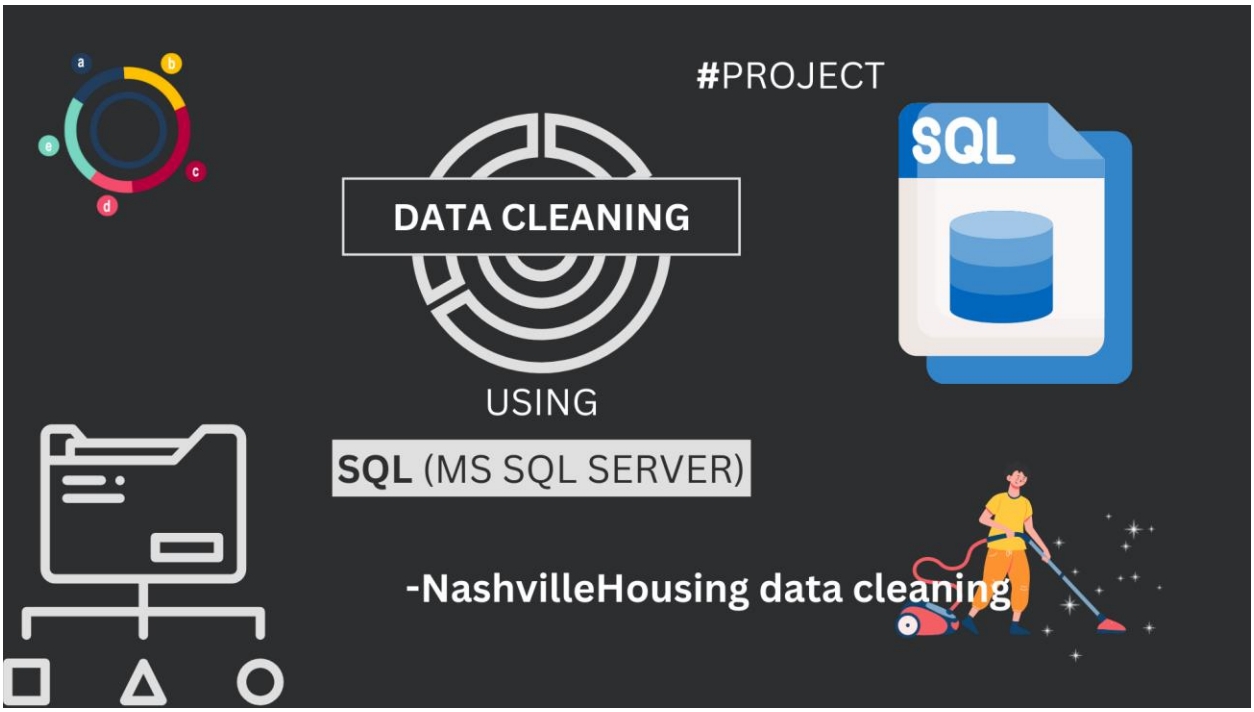
Data Cleaning Mastery: SQL Project for Efficient Data Cleansing



Description:

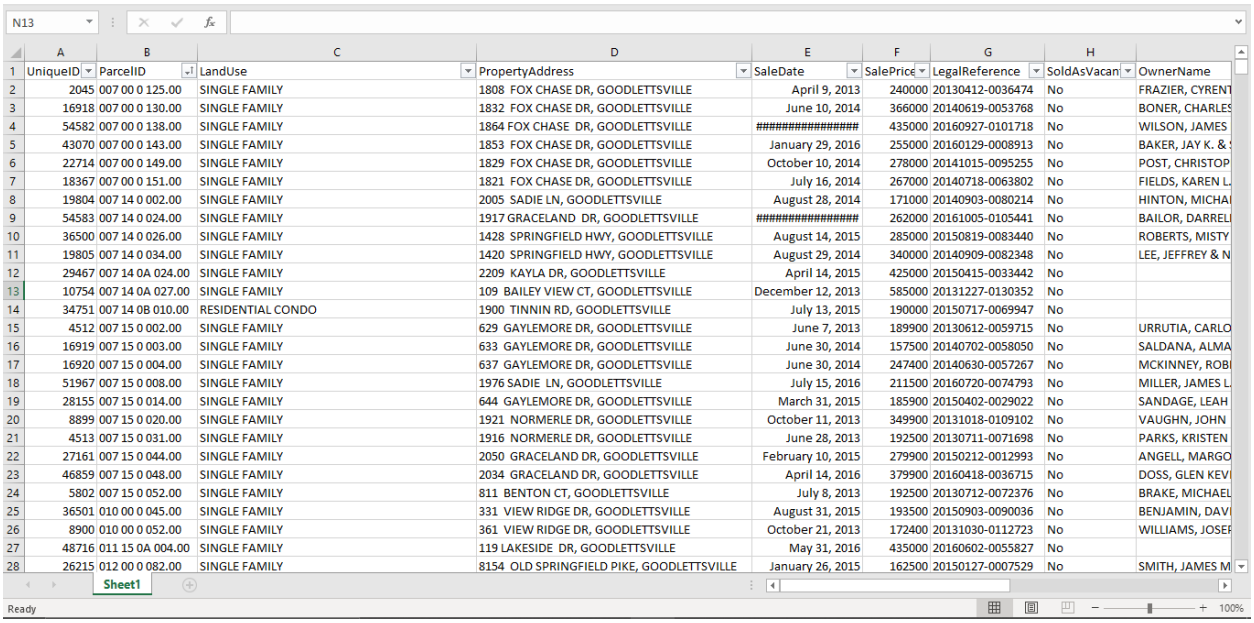
In this SQL project, I focused on data cleaning techniques. I downloaded a dataset from GitHub and imported it into an MS SQL server. I followed a data cleaning checklist that included identifying errors, ensuring consistent units and meanings, handling missing values, removing duplicates, addressing outliers, and more. Using SQL queries, I standardized date formats, populated property address data, broke addresses into individual columns, and made necessary changes to fields like "Sold as Vacant." I also removed duplicates and unused columns. This project deepened my understanding of data cleaning and its importance in accurate analysis.

Skills: Data Cleaning, SQL.

This project, it's all about cleaning data only using SQL.

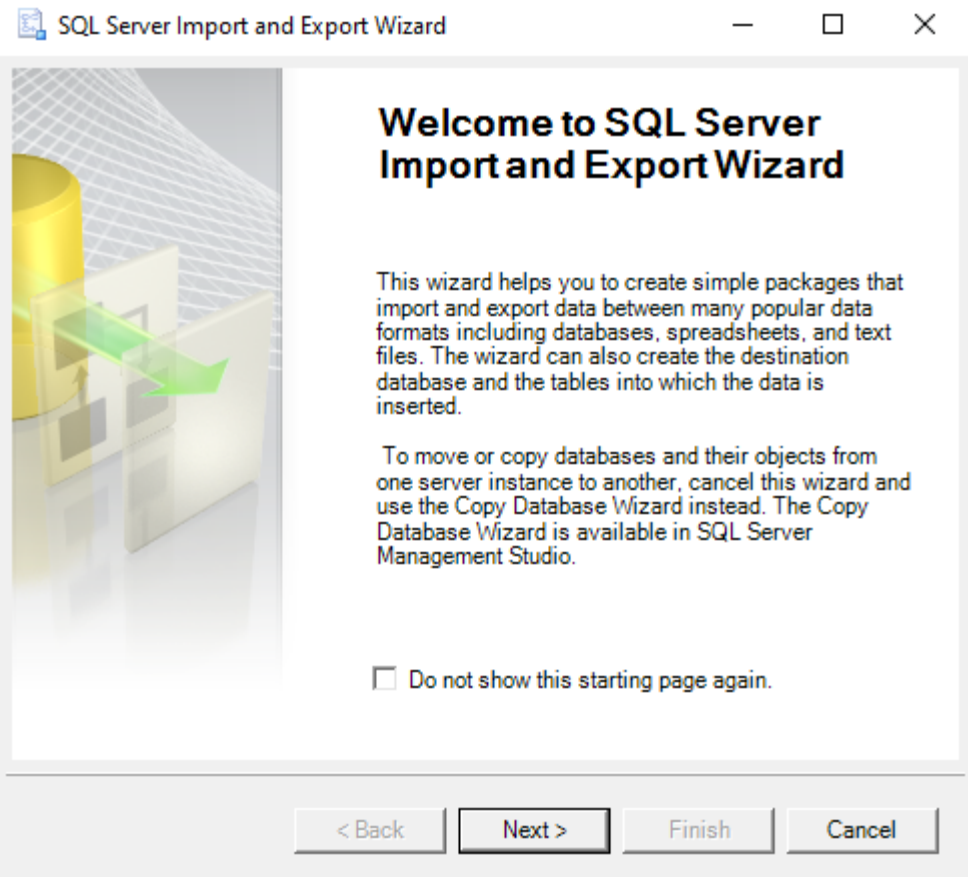
To do that I download a dataset from GitHub.

I download that file in xlsx format. To check that file, I opened it in MS Excel.



UniqueID	ParcelID	LandUse	PropertyAddress	SaleDate	SalePrice	LegalReference	SoldAsVacan	OwnerName
2045	007 00 0 125.00	SINGLE FAMILY	1808 FOX CHASE DR, GOODLETTSVILLE	April 9, 2013	240000	20130412-0036474	No	FRAZIER, CYRENT
16918	007 00 0 130.00	SINGLE FAMILY	1832 FOX CHASE DR, GOODLETTSVILLE	June 10, 2014	366000	20140619-0053768	No	BONER, CHARLES
54582	007 00 0 138.00	SINGLE FAMILY	1864 FOX CHASE DR, GOODLETTSVILLE	################	435000	20160927-0101718	No	WILSON, JAMES
43070	007 00 0 143.00	SINGLE FAMILY	1853 FOX CHASE DR, GOODLETTSVILLE	January 29, 2016	255000	20160129-0008913	No	BAKER, JAY K. &
22714	007 00 0 149.00	SINGLE FAMILY	1829 FOX CHASE DR, GOODLETTSVILLE	October 10, 2014	278000	20141015-0095255	No	POST, CHRISTOP
18367	007 00 0 151.00	SINGLE FAMILY	1821 FOX CHASE DR, GOODLETTSVILLE	July 16, 2014	267000	20140718-0063802	No	FIELDS, KAREN L.
19804	007 14 0 002.00	SINGLE FAMILY	2005 SADIE LN, GOODLETTSVILLE	August 28, 2014	171000	20140903-0080214	No	HINTON, MICHAI
54583	007 14 0 024.00	SINGLE FAMILY	1917 GRACELAND DR, GOODLETTSVILLE	################	262000	20161005-0105441	No	BAILOR, DARRELI
36500	007 14 0 026.00	SINGLE FAMILY	1428 SPRINGFIELD HWY, GOODLETTSVILLE	August 14, 2015	285000	20150819-0083440	No	ROBERTS, MISTY
19805	007 14 0 034.00	SINGLE FAMILY	1420 SPRINGFIELD HWY, GOODLETTSVILLE	August 29, 2014	340000	20140909-0082348	No	LEE, JEFFREY & N
29467	007 14 0A 024.00	SINGLE FAMILY	2209 KAYLA DR, GOODLETTSVILLE	April 14, 2015	425000	20150415-0033442	No	
10754	007 14 0A 027.00	SINGLE FAMILY	109 BAILEY VIEW CT, GOODLETTSVILLE	December 12, 2013	585000	20131227-0130352	No	
34751	007 14 0B 010.00	RESIDENTIAL CONDO	1900 TINNIN RD, GOODLETTSVILLE	July 13, 2015	190000	20150717-0069947	No	
4512	007 15 0 002.00	SINGLE FAMILY	629 GAYLEMORE DR, GOODLETTSVILLE	June 7, 2013	189900	20130612-0059715	No	URRUTIA, CARLO
16919	007 15 0 003.00	SINGLE FAMILY	633 GAYLEMORE DR, GOODLETTSVILLE	June 30, 2014	157500	20140702-0058050	No	SALDANA, ALMA
16920	007 15 0 004.00	SINGLE FAMILY	637 GAYLEMORE DR, GOODLETTSVILLE	June 30, 2014	247400	20140630-0057267	No	MCKINNEY, ROBI
51967	007 15 0 008.00	SINGLE FAMILY	1976 SADIE LN, GOODLETTSVILLE	July 15, 2016	211500	20160720-0074793	No	MILLER, JAMES L
28155	007 15 0 014.00	SINGLE FAMILY	644 GAYLEMORE DR, GOODLETTSVILLE	March 31, 2015	185900	20150402-0029022	No	SANDAGE, LEAH
8899	007 15 0 020.00	SINGLE FAMILY	1921 NORMERLE DR, GOODLETTSVILLE	October 11, 2013	349900	20131018-0109102	No	VAUGHN, JOHN
4513	007 15 0 031.00	SINGLE FAMILY	1916 NORMERLE DR, GOODLETTSVILLE	June 28, 2013	192500	20130711-0071698	No	PARKS, KRISTEN
27161	007 15 0 044.00	SINGLE FAMILY	2050 GRACELAND DR, GOODLETTSVILLE	February 10, 2015	279900	20150212-0012993	No	ANGELL, MARGO
46859	007 15 0 048.00	SINGLE FAMILY	2034 GRACELAND DR, GOODLETTSVILLE	April 14, 2016	379900	20160418-0036715	No	DOSS, GLEN KEV
5802	007 15 0 052.00	SINGLE FAMILY	811 BENTON CT, GOODLETTSVILLE	July 8, 2013	192500	20130712-0072376	No	BRAKE, MICHAEL
36501	010 00 0 045.00	SINGLE FAMILY	331 VIEW RIDGE DR, GOODLETTSVILLE	August 31, 2015	193500	20150903-0090036	No	BENJAMIN, DAV
8900	010 00 0 052.00	SINGLE FAMILY	361 VIEW RIDGE DR, GOODLETTSVILLE	October 21, 2013	172400	20131030-0112723	No	WILLIAMS, JOSEI
48716	011 15 0A 004.00	SINGLE FAMILY	119 LAKESIDE DR, GOODLETTSVILLE	May 31, 2016	435000	20160602-0055827	No	
26215	012 00 0 082.00	SINGLE FAMILY	8154 OLD SPRINGFIELD PIKE, GOODLETTSVILLE	January 26, 2015	162500	20150127-0007529	No	SMITH, JAMES M

After I checked, I started to import that file into the MS SQL server.



Then I started to clean the data. To clean data, everyone has some kind of checklist.

This is my data-cleaning checklist.

1. Are values prone to error?
2. Do we have the same unit for the data?
3. Is there consistency in the meaning of data?
4. Are there missing values in your data and what is the reason and what you can do about it?
5. Is the same value recorded in the same way everywhere?
6. Are there any duplicates?
7. Is your data unbiased?
8. Removed all sources of noise from your data?
9. Identified and remove sources of data leakage?
10. Are there any obvious outliers?

To clean data this wise, I used SQL language (Structured Query Language). This language is used to communicate databases.

In this project, I added command lines to my code to make everyone understand.

These are commands that I added to my queries.

- CLEANING DATA IN SQL QUERIES
- STANDARDIZE DATE FORMAT
- POPULATE PROPERTY ADDRESS DATA
- BREAKING OUT ADDRESSES INTO INDIVIDUAL COLUMNS (ADDRESS, CITY, STATE)
- CHANGE Y AND N TO YES AND NO IN THE "SOLD AS VACANT" FIELD
- REMOVE DUPLICATES
- ORDER BY PropertyAddress
- DELETE UNUSED COLUMNS

My Quires:

```
-----------------------------------------------------------------------------------------

--CLEANING DATA IN SQL QUERIES



SELECT *

FROM PortfolioProject..NashvilleHousing




--STANDARDIZE DATE FORMAT



SELECT saleDateConverted, CONVERT(Date,SaleDate)

FROM PortfolioProject..NashvilleHousing



UPDATE NashvilleHousing
```

```
SET SaleDate = CONVERT(Date,SaleDate)


ALTER TABLE NashvilleHousing

ADD SaleDateConverted Date;


UPDATE NashvilleHousing

SET SaleDateConverted = CONVERT(Date,SaleDate)



--POPULATE PROPERTY ADDRESS DATA


SELECT *

FROM PortfolioProject..NashvilleHousing

ORDER BY ParcelID


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

FROM PortfolioProject..NashvilleHousing AS a

JOIN PortfolioProject..NashvilleHousing AS 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..NashvilleHousing AS a

JOIN PortfolioProject..NashvilleHousing AS b

  ON a.ParcelID = b.ParcelID

AND a.UniqueID <> b.UniqueID

WHERE a.PropertyAddress IS NULL



--BREAKING OUT ADDRESSES INTO INDIVIDUAL COLUMNS (ADDRESS, CITY, STATE)


SELECT PropertyAddress

FROM PortfolioProject..NashvilleHousing


SELECT

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

SUBSTRING(PropertyAddress, CHARINDEX(',', PropertyAddress) +1 ,
LEN(PropertyAddress)) AS Address

FROM PortfolioProject..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..NashvilleHousing


SELECT

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

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

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

FROM PortfolioProject..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..NashvilleHousing



--CHANGE Y AND N TO YES AND NO IN THE "SOLD AS VACANT" FIELD
```

```
SELECT DISTINCT(SoldAsVacant), COUNT(SoldAsVacant)

FROM PortfolioProject..NashvilleHousing

GROUP BY SoldAsVacant

ORDER BY 2


SELECT SoldAsVacant,

   CASE WHEN SoldAsVacant = 'Y' THEN 'YES'

      WHEN SoldAsVacant = 'N' THEN 'NO'

  END

FROM PortfolioProject..NashvilleHousing


UPDATE NashvilleHousing

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

  WHEN SoldAsVacant = 'N' THEN 'No'

ELSE SoldAsVacant

END

FROM PortfolioProject..NashvilleHousing



--REMOVE DUPLICATES
```

```
WITH RowNumCTE AS(

SELECT *,

   ROW_NUMBER() OVER (

 PARTITION BY ParcelID,

             PropertyAddress,

SalePrice,

SaleDate,

LegalReference

ORDER BY

    UniqueID

  ) row_num

FROM PortfolioProject..NashvilleHousing

)

DELETE

FROM RowNumCTE

WHERE row_num > 1

--ORDER BY PropertyAddress


SELECT *

FROM PortfolioProject.dbo.NashvilleHousing
```

--DELETE UNUSED COLUMNS

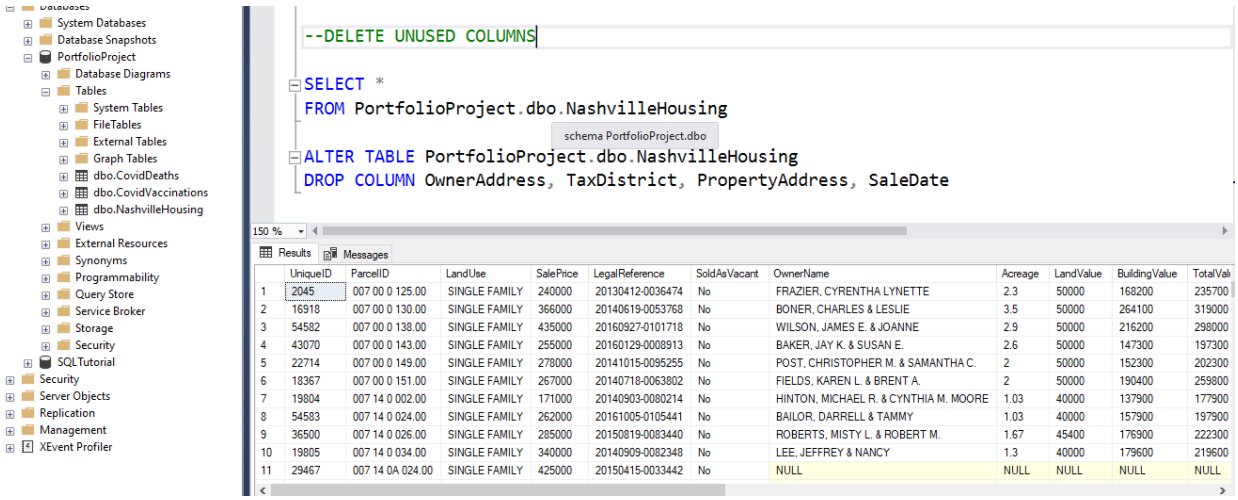
SELECT *

FROM PortfolioProject.dbo.NashvilleHousing

ALTER TABLE PortfolioProject.dbo.NashvilleHousing

DROP COLUMN OwnerAddress, TaxDistrict, PropertyAddress, SaleDate

---



	UniqueID	ParcelID	LandUse	SalePrice	LegalReference	SoldAsVacant	OwnerName	Acreage	LandValue	BuildingValue	TotalVal
1	2045	007 00 0 125.00	SINGLE FAMILY	240000	20130412-0036474	No	FRAZIER, CYRENTHA LYNETTE	2.3	50000	168200	235700
2	16918	007 00 0 130.00	SINGLE FAMILY	366000	20140619-0053768	No	BONER, CHARLES & LESLIE	3.5	50000	264100	319000
3	54582	007 00 0 138.00	SINGLE FAMILY	435000	20160927-0101718	No	WILSON, JAMES E. & JOANNE	2.9	50000	216200	298000
4	43070	007 00 0 143.00	SINGLE FAMILY	255000	20160129-0008913	No	BAKER, JAY K. & SUSAN E.	2.6	50000	147300	197300
5	22714	007 00 0 149.00	SINGLE FAMILY	278000	20141015-0095255	No	POST, CHRISTOPHER M. & SAMANTHA C.	2	50000	152300	202300
6	18367	007 00 0 151.00	SINGLE FAMILY	267000	20140718-0063802	No	FIELDS, KAREN L. & BRENT A.	2	50000	190400	259800
7	19804	007 14 0 002.00	SINGLE FAMILY	171000	20140903-0080214	No	HINTON, MICHAEL R. & CYNTHIA M. MOORE	1.03	40000	137900	177900
8	54583	007 14 0 024.00	SINGLE FAMILY	262000	20161005-0105441	No	BAILOR, DARRELL & TAMMY	1.03	40000	157900	197900
9	36500	007 14 0 026.00	SINGLE FAMILY	285000	20150819-0083440	No	ROBERTS, MISTY L. & ROBERT M.	1.67	45400	176900	222300
10	19805	007 14 0 034.00	SINGLE FAMILY	340000	20140909-0082348	No	LEE, JEFFREY & NANCY	1.3	40000	179600	219600
11	29467	007 14 0A 024.00	SINGLE FAMILY	425000	20150415-0033442	No	NULL	NULL	NULL	NULL	NULL

In the end, I gained a lot of experience in SQL. As a Data Analyst, I have to know about data cleaning. In this project, I learned more about data cleaning. Data cleaning is one of the important parts of analysis. By cleaning data at the start, we will save lots of time in the analyses and helps ensure that data is consistent so it can be analyzed accurately. While It's come to data cleaning everyone has different kinds of checklists. For me, I have my favorite checklist.