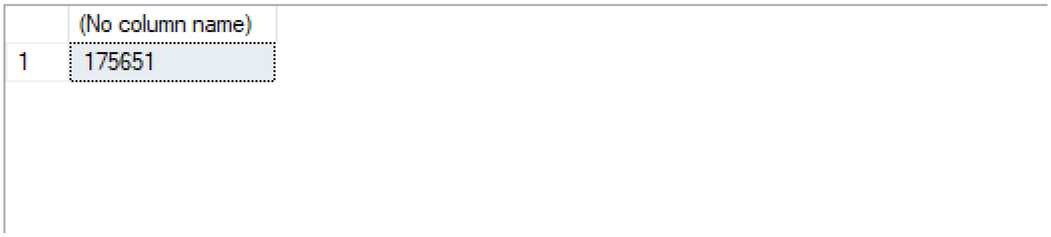
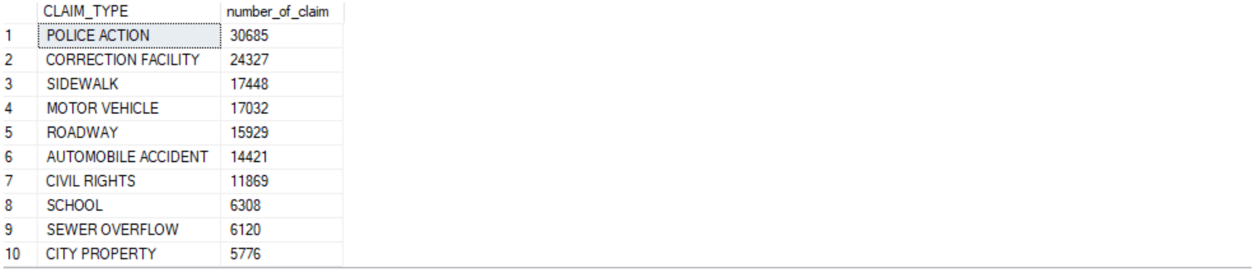
CLAIM INVESTIGATION

```
--- Let's create  a copy of the dataset

INSERT INTO  claims_study
SELECT * FROM [dbo].[claims_reports]

-- checking for duplicated. We will use the Claim # to check for duplactes
because it is likely to be unique indentifier for each claim
-- Some claims have beeng updated over time and we will be working only with
the most recent disposition date

WITH duplicates as
(SELECT
COUNT(*) OVER(partition by CLAIM ORDER BY DISPOSITION_DATE DESC) as
uniq_check,
*
FROM [dbo].[claims_study]
)
DELETE FROM duplicates WHERE uniq_check >1


--- For CONSISTENCY  we need to upadate the table to standardize the
different claim types

UPDATE  [dbo].[claims_study]
SET  [CLAIM_TYPE] = UPPER(TRIM(REPLACE([CLAIM_TYPE], '(PI)' , ' ')))
FROM [dbo].[claims_study]


 ---  UPDATE to get the unique description for the claim type

 UPDATE [dbo].[claims_study]
SET [CLAIM_TYPE] =
  CASE
  WHEN [CLAIM_TYPE] = 'NON COVERED AGENCY/CITY ' THEN 'NON COVERED AGENCY '
  WHEN [CLAIM_TYPE] = 'WATER MAIN ' THEN 'WATERMAIN BREAK '
  WHEN [CLAIM_TYPE] = 'DEFECT TRAF/LIGHT/STOP SIGN ' THEN 'DEFECTIVE
TRAFF/LIGHT/SIGN '
  WHEN [CLAIM_TYPE] = 'PARKS & RECREATION ' THEN 'RECREATION '
  WHEN [CLAIM_TYPE] = 'CIVIL RIGHTS CLAIMS ' THEN 'CIVIL RIGHTS '
  WHEN [CLAIM_TYPE] = 'DEFECTIVE SIDEWALK' THEN 'SIDEWALK'
  WHEN [CLAIM_TYPE] = 'DEFECTIVE ROADWAY' THEN 'ROADWAY'
  WHEN [CLAIM_TYPE] = 'CITY ONLY(NON COV AGY/BODY)' THEN 'CITY PROPERTY'
  WHEN [CLAIM_TYPE] = 'NON COVERED AGENCY AND CITY' THEN 'NON COVERED AGENCY'
  WHEN [CLAIM_TYPE] = 'EMPLOYEE UNIFORMED SERVICE' THEN 'UNIFORMED SERVICES
EMPLOYEE'
  WHEN [CLAIM_TYPE]  IN ('PEACE OFFICER (POLICE ACT) ','PEACE OFFICER/POLICE
ACTION ') THEN 'POLICE ACTION'
  WHEN [CLAIM_TYPE]  IN ('BUILDING AND PROPERTY ','BUILDINGS AND PROPERTY ')
THEN 'CITY PROPERTY'
  ELSE [CLAIM_TYPE]  END
 FROM [dbo].[claims_study]


-- Total number of claims
   SELECT
   COUNT(*)
   FROM [dbo].[claims_study]
```



	(No column name)
1	175651

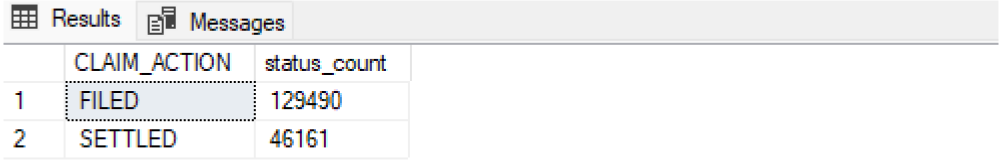
```
--- total claim in regard to claim type
  SELECT top 10
        CLAIM_TYPE,
        COUNT(*) AS number_of_claim
  FROM [dbo].[claims_study]
  GROUP BY [CLAIM_TYPE]
 order by number_of_claim desc
```



	CLAIM_TYPE	number_of_claim
1	POLICE ACTION	30685
2	CORRECTION FACILITY	24327
3	SIDEWALK	17448
4	MOTOR VEHICLE	17032
5	ROADWAY	15929
6	AUTOMOBILE ACCIDENT	14421
7	CIVIL RIGHTS	11869
8	SCHOOL	6308
9	SEWER OVERFLOW	6120
10	CITY PROPERTY	5776

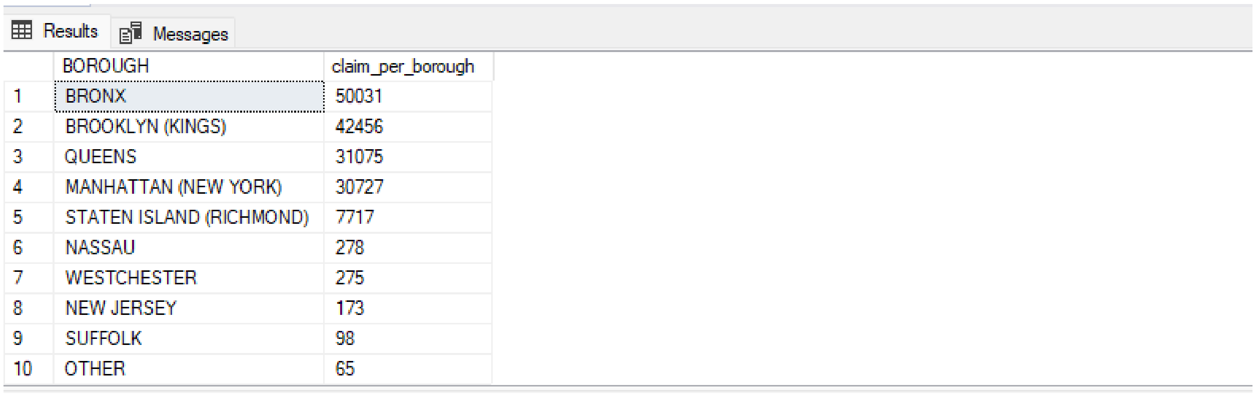
```
--Claim status count
select
       [CLAIM_ACTION],
       count([CLAIM]) status_count
 from [dbo].[claims_study]
 group by [CLAIM_ACTION]
```

```sql
    order by status_count desc
```



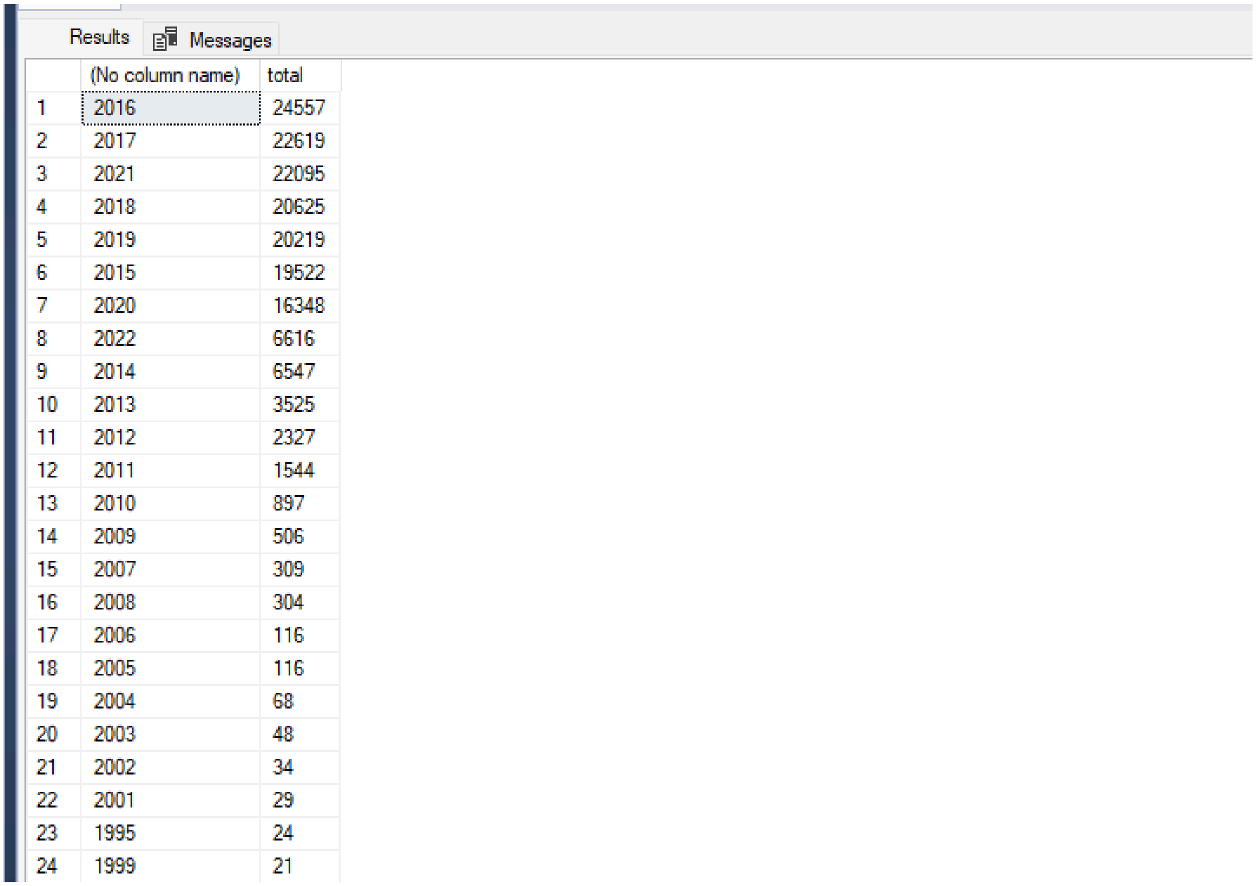
	CLAIM_ACTION	status_count
1	FILED	129490
2	SETTLED	46161

```sql
-- Claim by borough
SELECT TOP 10
       [BOROUGH],
       count([CLAIM]) as claim_per_borough
 FROM [dbo].[claims_study]
 where BOROUGH is not null
GROUP BY  [BOROUGH]
ORDER BY claim_per_borough desc
```



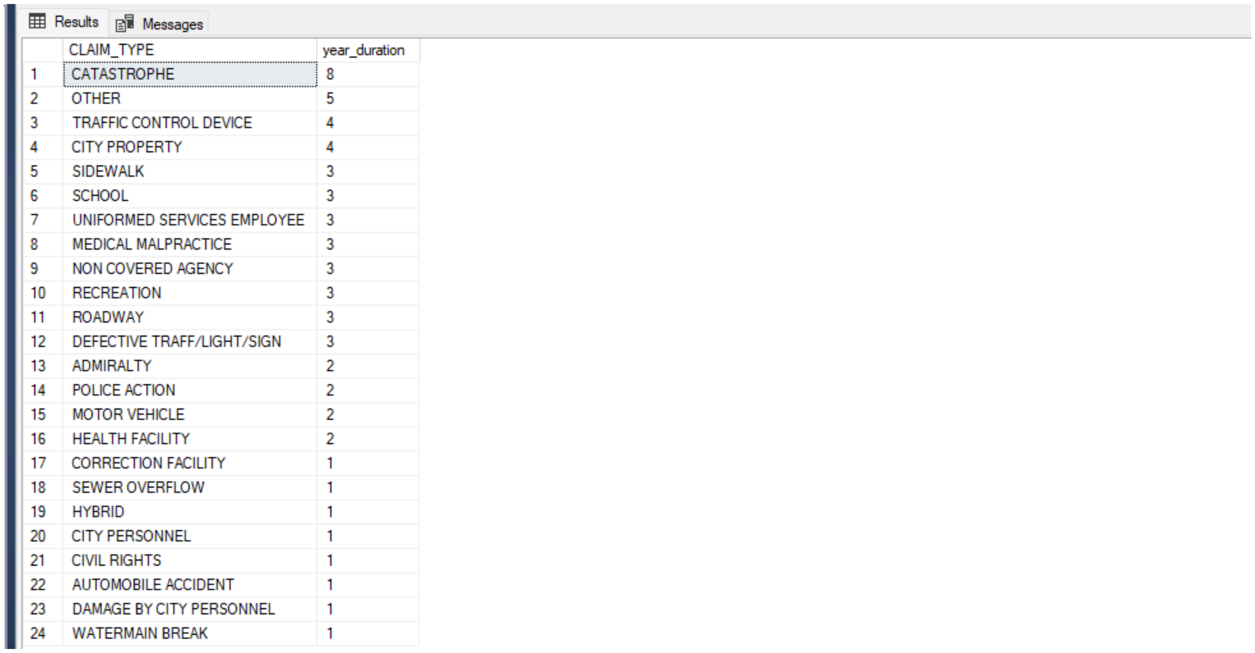
	BOROUGH	claim_per_borough
1	BRONX	50031
2	BROOKLYN (KINGS)	42456
3	QUEENS	31075
4	MANHATTAN (NEW YORK)	30727
5	STATEN ISLAND (RICHMOND)	7717
6	NASSAU	278
7	WESTCHESTER	275
8	NEW JERSEY	173
9	SUFFOLK	98
10	OTHER	65

```sql
  ---yearly distribution of claims
 SELECT
       datepart(year,[OCCURRENCE_DATE]),
       count(*) as total
FROM[dbo].[claims_study]
 WHERE [OCCURRENCE_DATE] is not null
 GROUP BY datepart(year,[OCCURRENCE_DATE])
ORDER BY total desc;
```



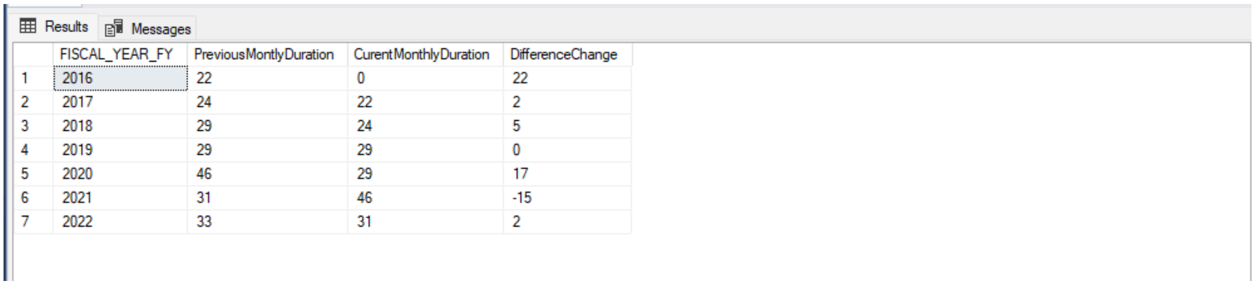
	(No column name)	total
1	2016	24557
2	2017	22619
3	2021	22095
4	2018	20625
5	2019	20219
6	2015	19522
7	2020	16348
8	2022	6616
9	2014	6547
10	2013	3525
11	2012	2327
12	2011	1544
13	2010	897
14	2009	506
15	2007	309
16	2008	304
17	2006	116
18	2005	116
19	2004	68
20	2003	48
21	2002	34
22	2001	29
23	1995	24
24	1999	21

```
  --what the avereage claim duration by claim type
  SELECT
      [CLAIM_TYPE],
       AVG(datediff(YEAR,[FILED_DATE], [DISPOSITION_DATE])) as year_duration
 from [dbo].[claims_study]
 where [DISPOSITION_DATE] is not null and [FILED_DATE] is not null
GROUP BY [CLAIM_TYPE]
ORDER BY year_duration DESC
```



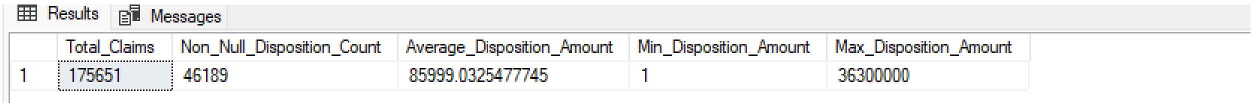
	CLAIM_TYPE	year_duration
1	CATASTROPHE	8
2	OTHER	5
3	TRAFFIC CONTROL DEVICE	4
4	CITY PROPERTY	4
5	SIDEWALK	3
6	SCHOOL	3
7	UNIFORMED SERVICES EMPLOYEE	3
8	MEDICAL MALPRACTICE	3
9	NON COVERED AGENCY	3
10	RECREATION	3
11	ROADWAY	3
12	DEFECTIVE TRAFF/LIGHT/SIGN	3
13	ADMIRALTY	2
14	POLICE ACTION	2
15	MOTOR VEHICLE	2
16	HEALTH FACILITY	2
17	CORRECTION FACILITY	1
18	SEWER OVERFLOW	1
19	HYBRID	1
20	CITY PERSONNEL	1
21	CIVIL RIGHTS	1
22	AUTOMOBILE ACCIDENT	1
23	DAMAGE BY CITY PERSONNEL	1
24	WATERMAIN BREAK	1

```
---Year-over-Year Analysis of Monthly Claim Durations
with x as
 (SELECT
 FISCAL_YEAR_FY,
 AVG(datediff(month,FILED_DATE, DISPOSITION_DATE)) as PreviousMontlyDuration
FROM[dbo].[claims_study]
WHERE DISPOSITION_DATE is not null and FILED_DATE is not null
GROUP BY FISCAL_YEAR_FY
), y as
(SELECT
      *,
       LAG( PreviousMontlyDuration,1,0) OVER(ORDER BY FISCAL_YEAR_FY  ) as
CurentMonthlyDuration
       from x
)
SELECT *,
PreviousMontlyDuration-CurentMonthlyDuration  as DifferenceChange
FROM Y
```



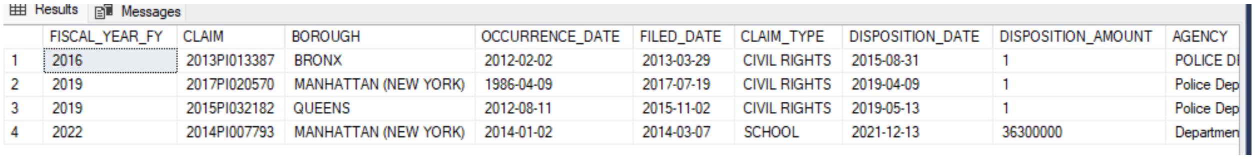
	FISCAL_YEAR_FY	PreviousMontlyDuration	CurentMonthlyDuration	DifferenceChange
1	2016	22	0	22
2	2017	24	22	2
3	2018	29	24	5
4	2019	29	29	0
5	2020	46	29	17
6	2021	31	46	-15
7	2022	33	31	2

```sql
-- Claim disposition summary
SELECT
    COUNT(*) AS Total_Claims,
    COUNT(Disposition_Amount) AS Non_Null_Disposition_Count,
    AVG(Disposition_Amount) AS Average_Disposition_Amount,
    MIN(Disposition_Amount) AS Min_Disposition_Amount,
    MAX(Disposition_Amount) AS Max_Disposition_Amount
FROM [dbo].[claims_study];
```



	Total_Claims	Non_Null_Disposition_Count	Average_Disposition_Amount	Min_Disposition_Amount	Max_Disposition_Amount
1	175651	46189	85999.0325477745	1	36300000

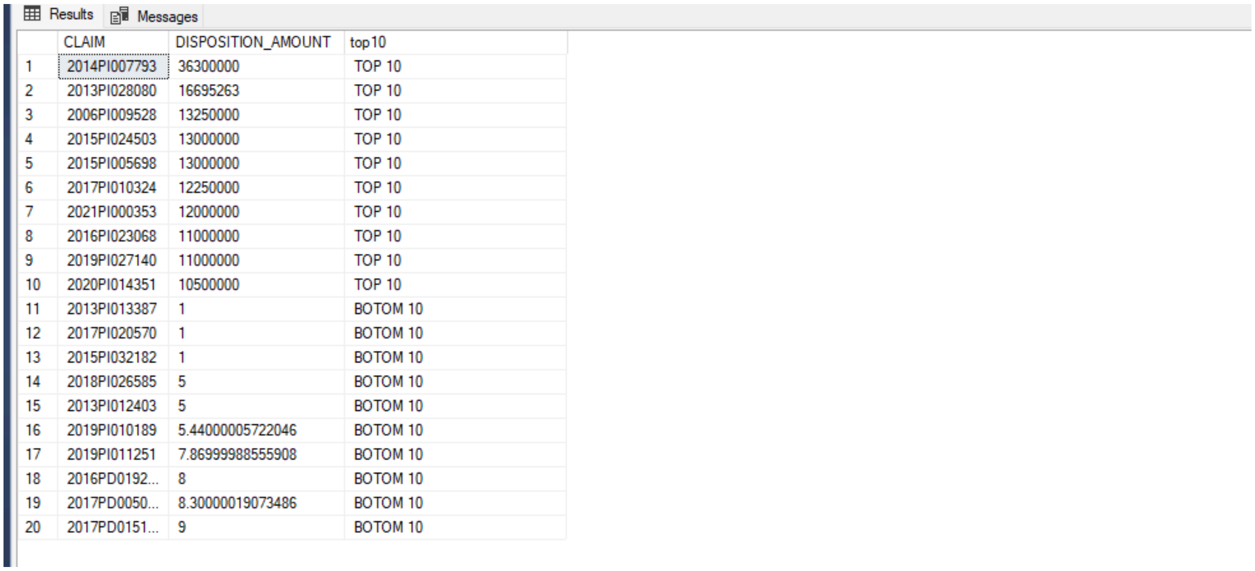
```sql
---let's investigate those claims with the min and max disposition amount.
SELECT *
FROM  [dbo].[claims_study]
WHERE
DISPOSITION_AMOUNT = (select min([DISPOSITION_AMOUNT]) from
[dbo].[claims_study])
or
DISPOSITION_AMOUNT  = (select max([DISPOSITION_AMOUNT]) from
[dbo].[claims_study]);
```



	FISCAL_YEAR_FY	CLAIM	BOROUGH	OCCURRENCE_DATE	FILED_DATE	CLAIM_TYPE	DISPOSITION_DATE	DISPOSITION_AMOUNT	AGENCY
1	2016	2013PI013387	BRONX	2012-02-02	2013-03-29	CIVIL RIGHTS	2015-08-31	1	POLICE DI
2	2019	2017PI020570	MANHATTAN (NEW YORK)	1986-04-09	2017-07-19	CIVIL RIGHTS	2019-04-09	1	Police Dep
3	2019	2015PI032182	QUEENS	2012-08-11	2015-11-02	CIVIL RIGHTS	2019-05-13	1	Police Dep
4	2022	2014PI007793	MANHATTAN (NEW YORK)	2014-01-02	2014-03-07	SCHOOL	2021-12-13	36300000	Departmen

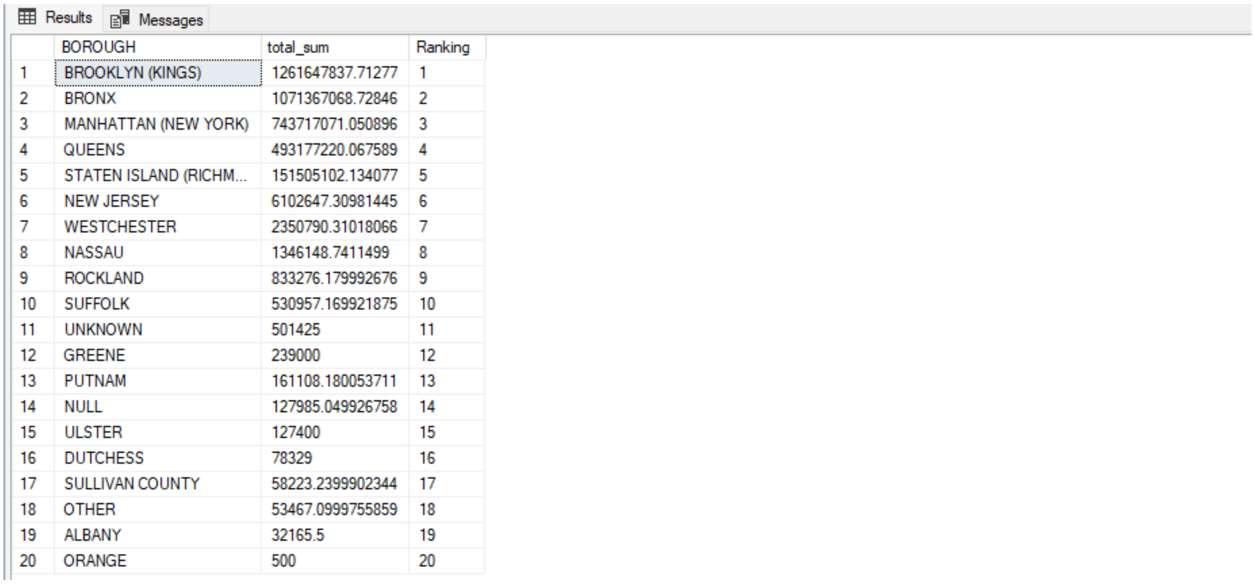
```sql
---Top and bottom Claims by disposition amount
 WITH claimCost as
 (SELECT
[CLAIM],
[DISPOSITION_AMOUNT],
 ROW_NUMBER() OVER (ORDER BY[DISPOSITION_AMOUNT] DESC) AS TopAnalysis,
 ROW_NUMBER() OVER(ORDER BY [DISPOSITION_AMOUNT] ASC) BottomAnalysis
FROM [dbo].[claims_study]
WHERE [DISPOSITION_AMOUNT]
IS NOT NULL
)
SELECT
CLAIM,
[DISPOSITION_AMOUNT],

'TOP 10' AS top10
FROM claimCost
WHERE TopAnalysis <=10

UNION all
```

```sql
SELECT
[CLAIM],
[DISPOSITION_AMOUNT],
'BOTOM 10'
FROM claimCost
WHERE BottomAnalysis <=10;
```



	CLAIM	DISPOSITION_AMOUNT	top10
1	2014PI007793	36300000	TOP 10
2	2013PI028080	16695263	TOP 10
3	2006PI009528	13250000	TOP 10
4	2015PI024503	13000000	TOP 10
5	2015PI005698	13000000	TOP 10
6	2017PI010324	12250000	TOP 10
7	2021PI000353	12000000	TOP 10
8	2016PI023068	11000000	TOP 10
9	2019PI027140	11000000	TOP 10
10	2020PI014351	10500000	TOP 10
11	2013PI013387	1	BOTOM 10
12	2017PI020570	1	BOTOM 10
13	2015PI032182	1	BOTOM 10
14	2018PI026585	5	BOTOM 10
15	2013PI012403	5	BOTOM 10
16	2019PI010189	5.44000005722046	BOTOM 10
17	2019PI011251	7.86999988555908	BOTOM 10
18	2016PD0192...	8	BOTOM 10
19	2017PD0050...	8.30000019073486	BOTOM 10
20	2017PD0151...	9	BOTOM 10

```sql
-- let rank the BOROUGH based on their disposition amount
SELECT
      [BOROUGH],
      sum(DISPOSITION_AMOUNT) as total_sum,
      ROW_NUMBER() over( ORDER BY sum(DISPOSITION_AMOUNT) DESC) Ranking
FROM[dbo].[claims_study]
WHERE BOROUGH is not null AND DISPOSITION_AMOUNT IS NOT NULL
GROUP BY BOROUGH;
```



	BOROUGH	total_sum	Ranking
1	BROOKLYN (KINGS)	1261647837.71277	1
2	BRONX	1071367068.72846	2
3	MANHATTAN (NEW YORK)	743717071.050896	3
4	QUEENS	493177220.067589	4
5	STATEN ISLAND (RICHM...	151505102.134077	5
6	NEW JERSEY	6102647.30981445	6
7	WESTCHESTER	2350790.31018066	7
8	NASSAU	1346148.7411499	8
9	ROCKLAND	833276.179992676	9
10	SUFFOLK	530957.169921875	10
11	UNKNOWN	501425	11
12	GREENE	239000	12
13	PUTNAM	161108.180053711	13
14	NULL	127985.049926758	14
15	ULSTER	127400	15
16	DUTCHESS	78329	16
17	SULLIVAN COUNTY	58223.2399902344	17
18	OTHER	53467.0999755859	18
19	ALBANY	32165.5	19
20	ORANGE	500	20