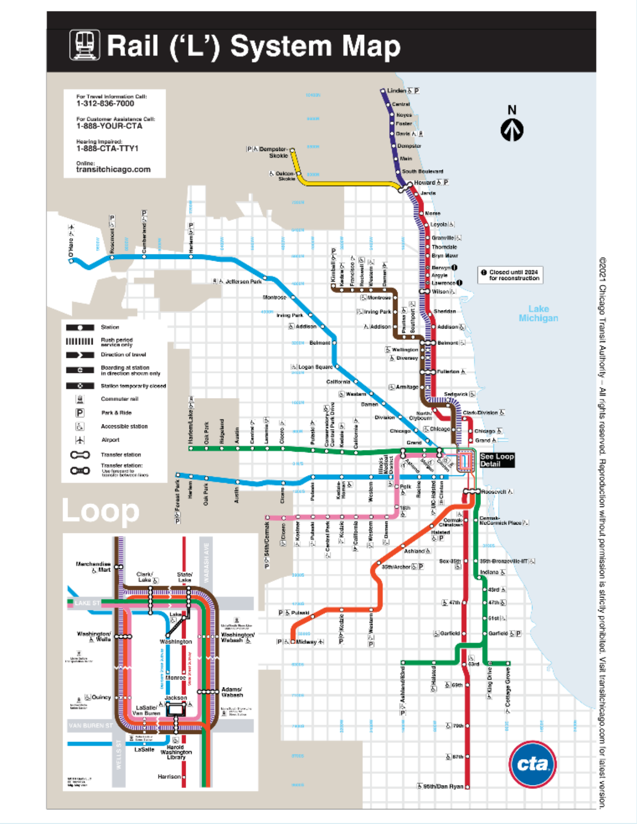
More with SQL Select queries

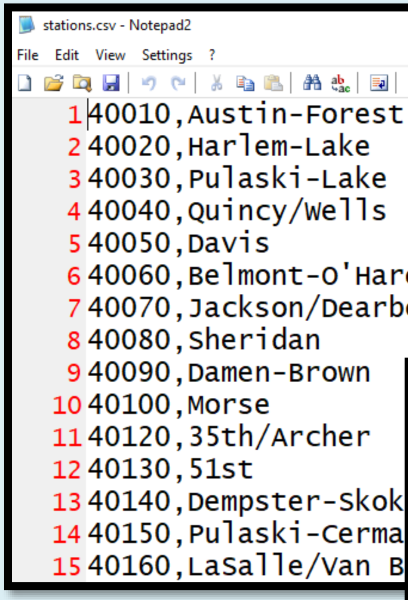
- SQL Select queries, part 02
- Grouping data
- Joining tables

Northwestern University

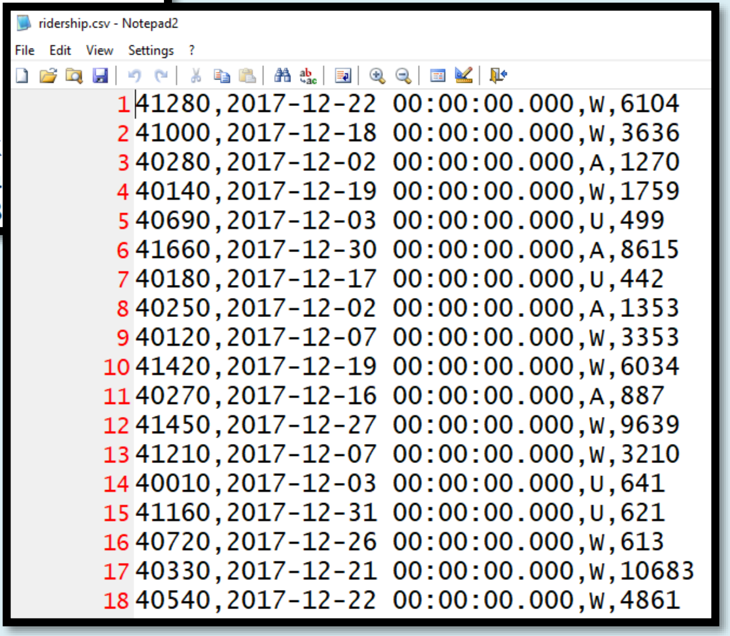
Database example: CTA

- We have some CTA ridership data (L stations) that we need to analyze…





```
40010,Austin-Forest Park
40020,Harlem-Lake
40030,Pulaski-Lake
40040,Quincy/Wells
40050,Davis
40060,Belmont-O'Hare
40070,Jackson/Dearborn
40080,Sheridan
40090,Damen-Brown
40100,Morse
40120,35th/Archer
40130,51st
40140,Dempster-Skok
40150,Pulaski-Cerma
40160,LaSalle/Van B
```



```
41280,2017-12-22 00:00:00.000,W,6104
41000,2017-12-18 00:00:00.000,W,3636
40280,2017-12-02 00:00:00.000,A,1270
40140,2017-12-19 00:00:00.000,W,1759
40690,2017-12-03 00:00:00.000,U,499
41660,2017-12-30 00:00:00.000,A,8615
40180,2017-12-17 00:00:00.000,U,442
40250,2017-12-02 00:00:00.000,A,1353
40120,2017-12-07 00:00:00.000,W,3353
41420,2017-12-19 00:00:00.000,W,6034
40270,2017-12-16 00:00:00.000,A,887
41450,2017-12-27 00:00:00.000,W,9639
41210,2017-12-07 00:00:00.000,W,3210
40010,2017-12-03 00:00:00.000,U,641
41160,2017-12-31 00:00:00.000,U,621
40720,2017-12-26 00:00:00.000,W,613
40330,2017-12-21 00:00:00.000,W,10683
40540,2017-12-22 00:00:00.000,W,4861
```

CTA database (subset)

table: Stations

Station_ID	Station_Name
40010	*Austin-Forest Park*
40020	*Harlem-Lake*
40030	*Pulaski-Lake*
…	…

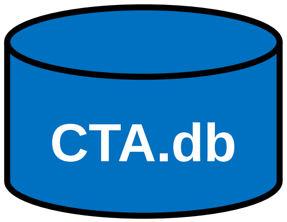


table: Ridership

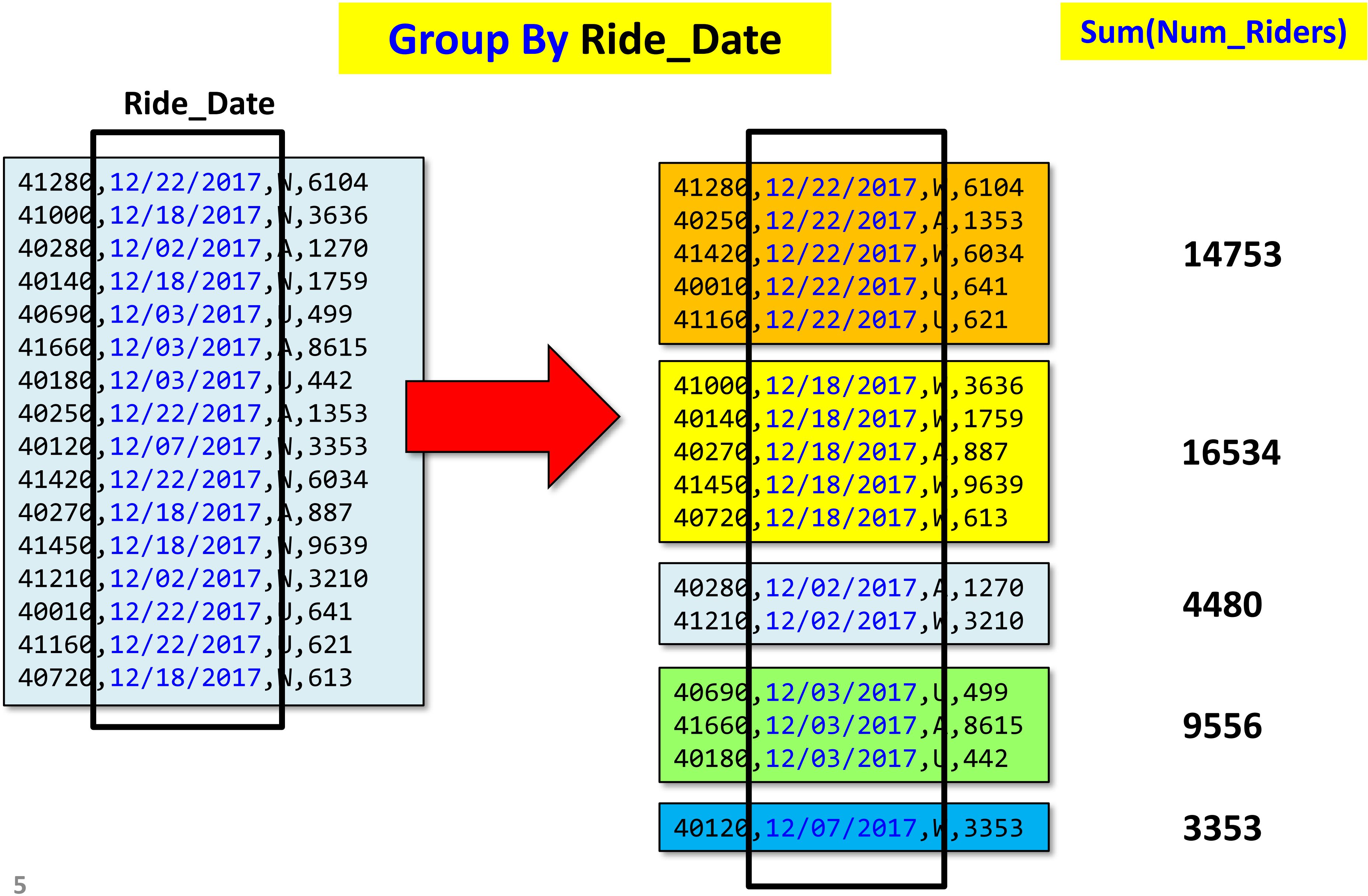
Station_ID	Ride_Date	Type_of_Day	Num_Riders
41280	2017-12-22 00:00:00.000	W	6104
40010	2017-12-28 00:00:00.000	W	1155
40280	2017-12-02 00:00:00.000	A	1270
40030	2017-12-24 00.00.00.000	U	595
…	…	…	…

Group By and Having

- **Group by partitions the data into subsets**
  - *Functions then apply to the subsets*
  - ***Where*** *clause applies before grouping,* ***Having*** *applies after*

```
SELECT <<the data you want>>
FROM   <<table(s)>>
[ WHERE     <<condition(s)>> ]
[ GROUP BY  <<one or more fields>> ]
[ HAVING    <<condition(s)>> ]
[ ORDER BY  <<one or more fields>> ]
;
```

Example



Group By Ride_Date

Sum(Num_Riders)

Ride_Date

```
41280,12/22/2017,W,6104
41000,12/18/2017,W,3636
40280,12/02/2017,A,1270
40140,12/18/2017,W,1759
40690,12/03/2017,U,499
41660,12/03/2017,A,8615
40180,12/03/2017,U,442
40250,12/22/2017,A,1353
40120,12/07/2017,W,3353
41420,12/22/2017,W,6034
40270,12/18/2017,A,887
41450,12/18/2017,W,9639
41210,12/02/2017,W,3210
40010,12/22/2017,U,641
41160,12/22/2017,U,621
40720,12/18/2017,W,613
```

```
41280,12/22/2017,W,6104
40250,12/22/2017,A,1353
41420,12/22/2017,W,6034
40010,12/22/2017,U,641
41160,12/22/2017,U,621
```

14753

```
41000,12/18/2017,W,3636
40140,12/18/2017,W,1759
40270,12/18/2017,A,887
41450,12/18/2017,W,9639
40720,12/18/2017,W,613
```

16534

```
40280,12/02/2017,A,1270
41210,12/02/2017,W,3210
```

4480

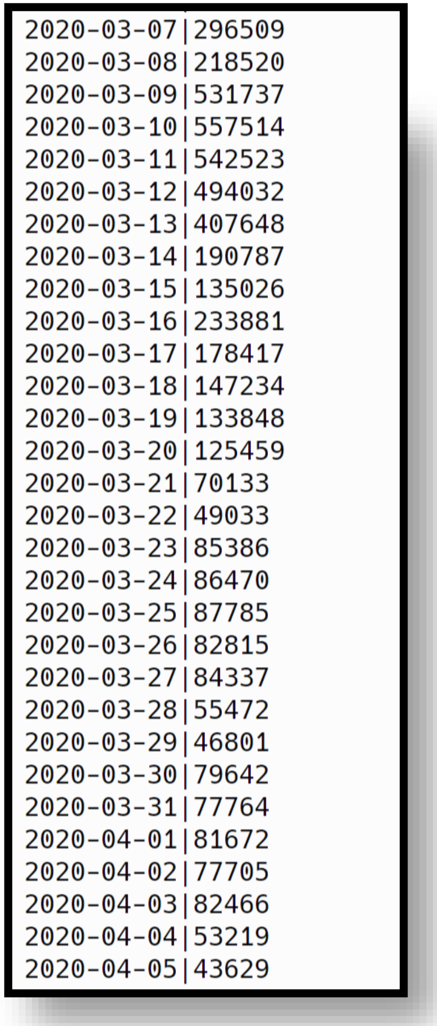
```
40690,12/03/2017,U,499
41660,12/03/2017,A,8615
40180,12/03/2017,U,442
```

9556

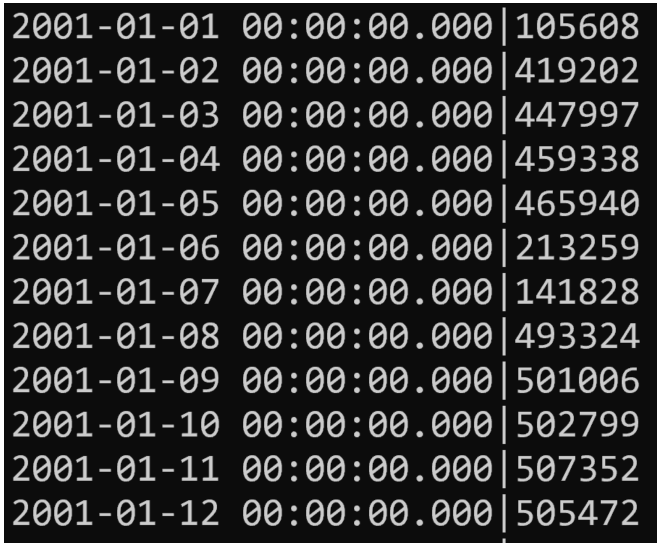
```
40120,12/07/2017,W,3353
```

3353

Example: riders per day



```
2020-03-07|296509
2020-03-08|218520
2020-03-09|531737
2020-03-10|557514
2020-03-11|542523
2020-03-12|494032
2020-03-13|407648
2020-03-14|190787
2020-03-15|135026
2020-03-16|233881
2020-03-17|178417
2020-03-18|147234
2020-03-19|133848
2020-03-20|125459
2020-03-21|70133
2020-03-22|49033
2020-03-23|85386
2020-03-24|86470
2020-03-25|87785
2020-03-26|82815
2020-03-27|84337
2020-03-28|55472
2020-03-29|46801
2020-03-30|79642
2020-03-31|77764
2020-04-01|81672
2020-04-02|77705
2020-04-03|82466
2020-04-04|53219
2020-04-05|43629
```

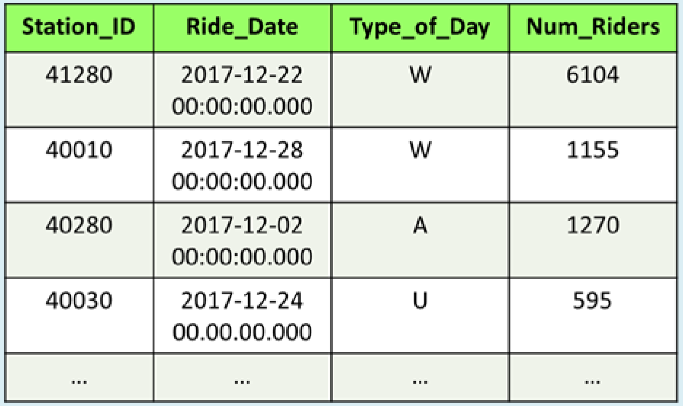


```
2001-01-01 00:00:00.000|105608
2001-01-02 00:00:00.000|419202
2001-01-03 00:00:00.000|447997
2001-01-04 00:00:00.000|459338
2001-01-05 00:00:00.000|465940
2001-01-06 00:00:00.000|213259
2001-01-07 00:00:00.000|141828
2001-01-08 00:00:00.000|493324
2001-01-09 00:00:00.000|501006
2001-01-10 00:00:00.000|502799
2001-01-11 00:00:00.000|507352
2001-01-12 00:00:00.000|505472
```

```
select   Ride_Date, Sum(Num_Riders)
from     Ridership
group by Ride_Date
order by Ride_Date ASC;
```

Question

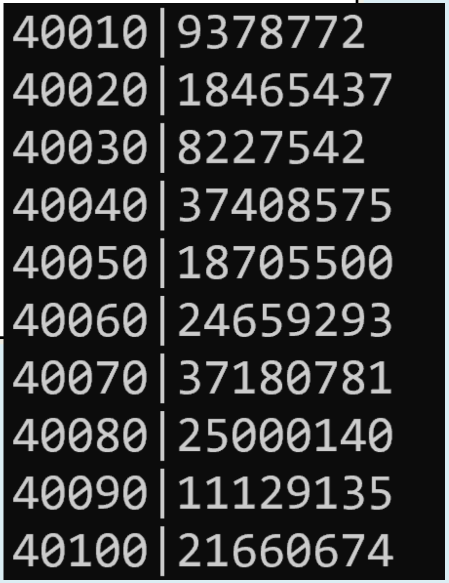
- **What is the sum of ridership per station, on weekdays?**
  - *Hint: group by what?*



Station_ID	Ride_Date	Type_of_Day	Num_Riders
41280	2017-12-22 00:00:00.000	W	6104
40010	2017-12-28 00:00:00.000	W	1155
40280	2017-12-02 00:00:00.000	A	1270
40030	2017-12-24 00.00.00.000	U	595
…	…	…	…

Ridership

```
select    ?
from      Ridership
where     ?
group by  ?
order by  ?;
```



```
40010|9378772
40020|18465437
40030|8227542
40040|37408575
40050|18705500
40060|24659293
40070|37180781
40080|25000140
40090|11129135
40100|21660674
```

Query

```sql
select   Station_ID, sum(Num_Riders)
from     Ridership
where    Type_of_Day = 'W'
group by Station_ID
order by Station_ID ASC;
```

Consider the table… Let's look at the execution pipeline…

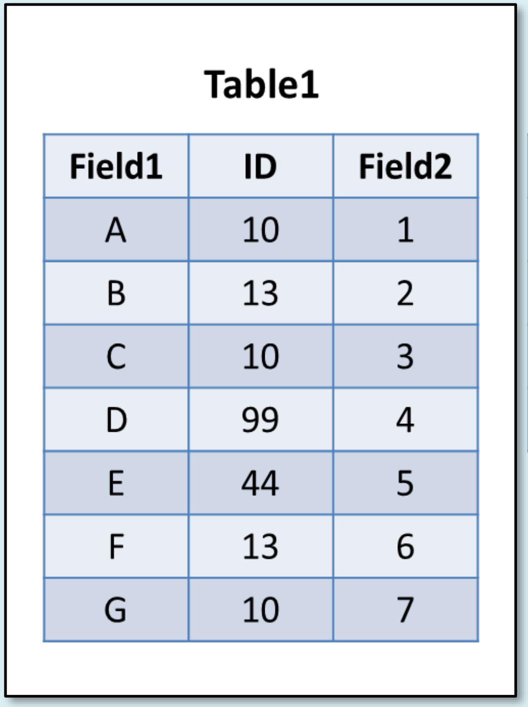


Table1

Field1	ID	Field2
A	10	1
B	13	2
C	10	3
D	99	4
E	44	5
F	13	6
G	10	7

```
SELECT    ID, Count(ID) AS Num
FROM      Table1
WHERE     1 < Field2 AND Field2 < 7
GROUP BY ID
HAVING    Num > 1
ORDER BY ID ASC;
```

Execution

```
SELECT   ID, Count(ID) AS Num
FROM     Table1
WHERE    1 < Field2 AND Field2 < 7
GROUP BY ID
HAVING   Num > 1
ORDER BY ID ASC;
```

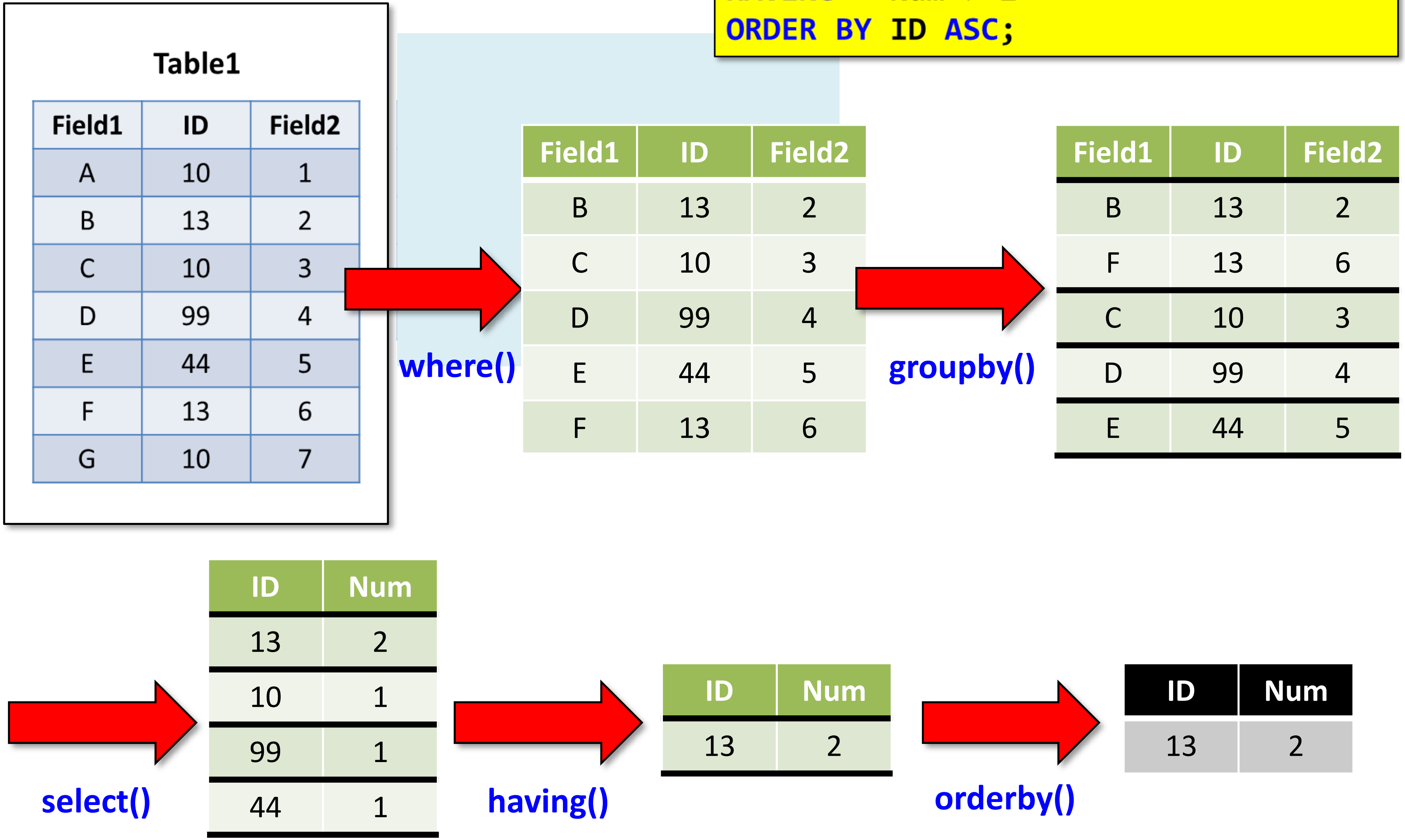


Table1

Field1	ID	Field2
A	10	1
B	13	2
C	10	3
D	99	4
E	44	5
F	13	6
G	10	7

where()

Field1	ID	Field2
B	13	2
C	10	3
D	99	4
E	44	5
F	13	6

groupby()

Field1	ID	Field2
B	13	2
F	13	6
C	10	3
D	99	4
E	44	5

select()

ID	Num
13	2
10	1
99	1
44	1

having()

ID	Num
13	2

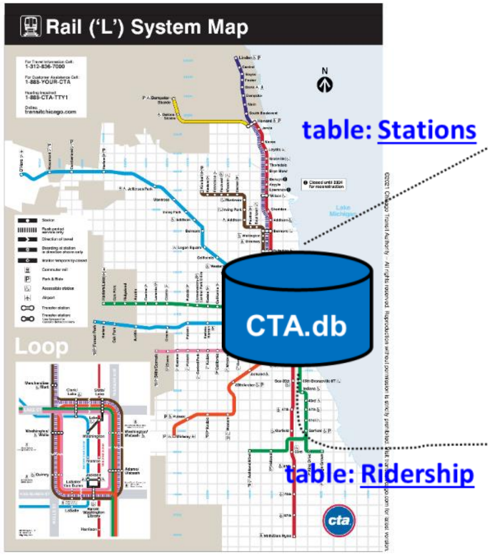
orderby()

ID	Num
13	2

Joins

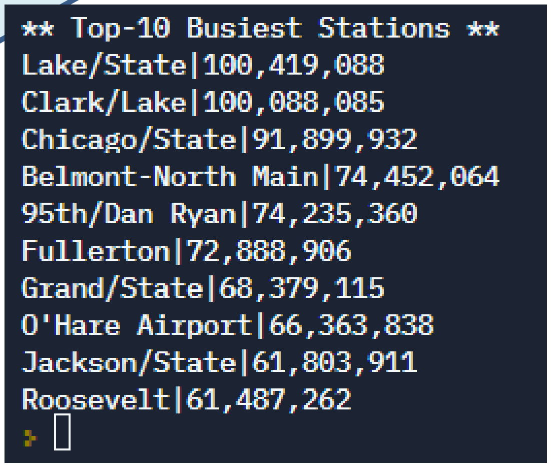
- **Joins are used to efficiently merge tables together**
  - *When we need data from both…*

Example: we want top-10 stations in terms of ridership, with Name not Station ID…



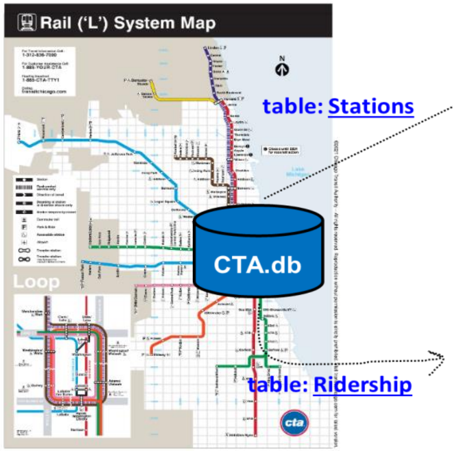
Station_ID	Station_Name
40010	*Austin-Forest Park*
40020	*Harlem-Lake*
40030	*Pulaski-Lake*
…	…

Station_ID	Ride_Date	Type_of_Day	Num_Riders
41280	2017-12-22 00:00:00.000	W	6104
40010	2017-12-28 00:00:00.000	W	1155
40280	2017-12-02 00:00:00.000	A	1270
40030	2017-12-24 00.00.00.000	U	595
…	…	…	…



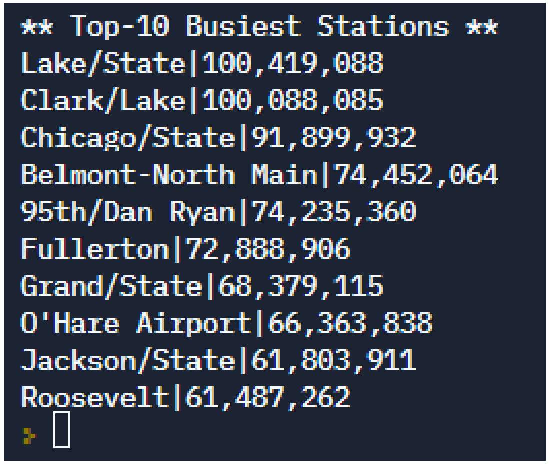
```
** Top-10 Busiest Stations **
Lake/State|100,419,088
Clark/Lake|100,088,085
Chicago/State|91,899,932
Belmont-North Main|74,452,064
95th/Dan Ryan|74,235,360
Fullerton|72,888,906
Grand/State|68,379,115
O'Hare Airport|66,363,838
Jackson/State|61,803,911
Roosevelt|61,487,262
```

Top-10 Query



Station_ID	Station_Name
40010	*Austin-Forest Park*
40020	*Harlem-Lake*
40030	*Pulaski-Lake*
…	…

Station_ID	Ride_Date	Type_of_Day	Num_Riders
41280	2017-12-22 00:00:00.000	W	6104
40010	2017-12-28 00:00:00.000	W	1155
40280	2017-12-02 00:00:00.000	A	1270
40030	2017-12-24 00.00.00.000	U	595
…	…	…	…



```
** Top-10 Busiest Stations **
Lake/State|100,419,088
Clark/Lake|100,088,085
Chicago/State|91,899,932
Belmont-North Main|74,452,064
95th/Dan Ryan|74,235,360
Fullerton|72,888,906
Grand/State|68,379,115
O'Hare Airport|66,363,838
Jackson/State|61,803,911
Roosevelt|61,487,262
```

```
select "** Top-10 Busiest Stations **";

Select Station_Name, Sum(Num_Riders)
From   Stations
Join Ridership On Stations.Station_ID = Ridership.Station_ID
Group By Stations.Station_ID
Order By Sum(Num_Riders) DESC
Limit 10;
```

Join

```
SELECT <<the data you want>>
FROM   <<table(s)>>
[ JOIN      <<other table(s)>> ]
[ WHERE     <<condition(s)>> ]
[ GROUP BY  <<one or more fields>> ]
[ Having    <<conditions(s)>> ]
[ ORDER BY  <<one or more fields>> ]
;
```

- **By default, join performs cartesian product**
  - *All possible combinations*
  - *i.e. combines each row in left table with each row in right table*

```
SELECT *
FROM Table1
JOIN Table2;
```

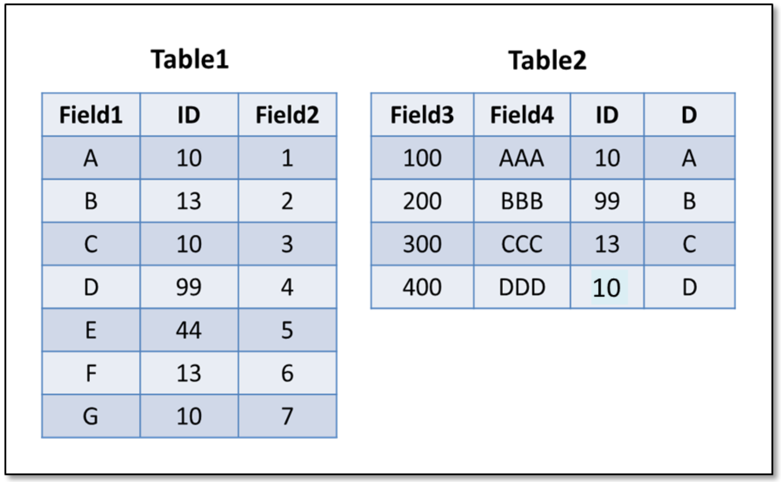
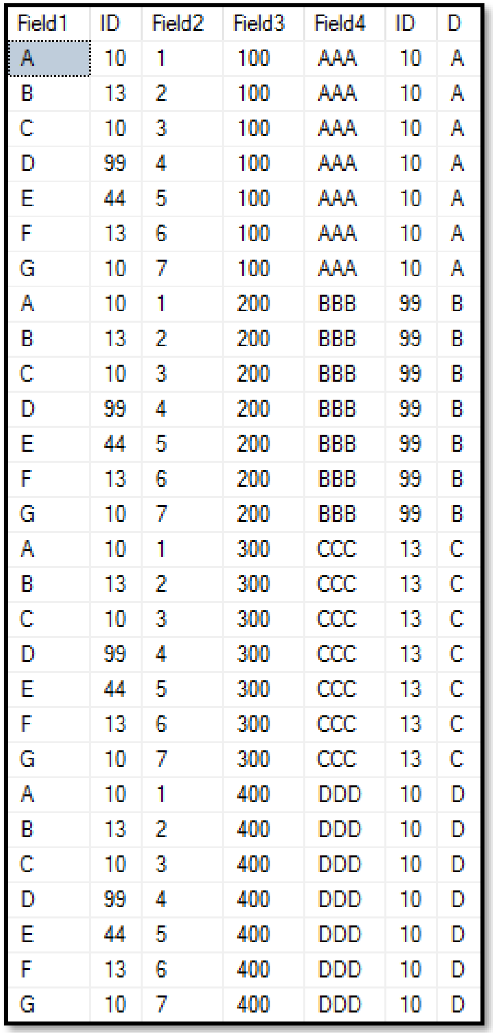


Table1

Field1	ID	Field2
A	10	1
B	13	2
C	10	3
D	99	4
E	44	5
F	13	6
G	10	7

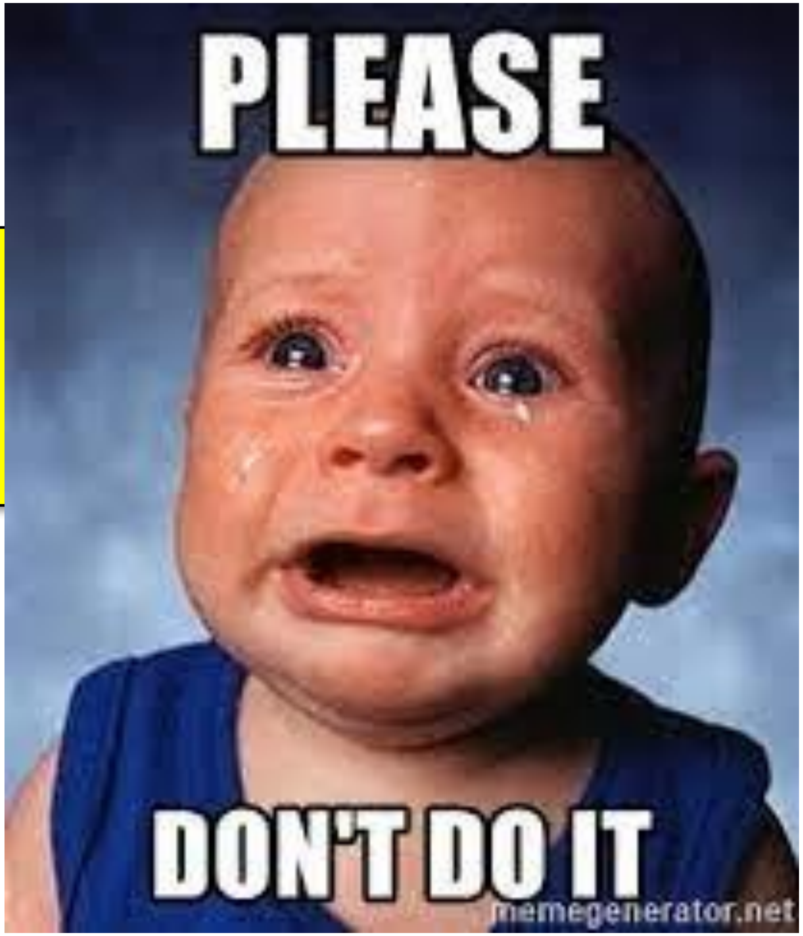
Table2

Field3	Field4	ID	D
100	AAA	10	A
200	BBB	99	B
300	CCC	13	C
400	DDD	10	D



Field1	ID	Field2	Field3	Field4	ID	D
A	10	1	100	AAA	10	A
B	13	2	100	AAA	10	A
C	10	3	100	AAA	10	A
D	99	4	100	AAA	10	A
E	44	5	100	AAA	10	A
F	13	6	100	AAA	10	A
G	10	7	100	AAA	10	A
A	10	1	200	BBB	99	B
B	13	2	200	BBB	99	B
C	10	3	200	BBB	99	B
D	99	4	200	BBB	99	B
E	44	5	200	BBB	99	B
F	13	6	200	BBB	99	B
G	10	7	200	BBB	99	B
A	10	1	300	CCC	13	C
B	13	2	300	CCC	13	C
C	10	3	300	CCC	13	C
D	99	4	300	CCC	13	C
E	44	5	300	CCC	13	C
F	13	6	300	CCC	13	C
G	10	7	300	CCC	13	C
A	10	1	400	DDD	10	D
B	13	2	400	DDD	10	D
C	10	3	400	DDD	10	D
D	99	4	400	DDD	10	D
E	44	5	400	DDD	10	D
F	13	6	400	DDD	10	D
G	10	7	400	DDD	10	D

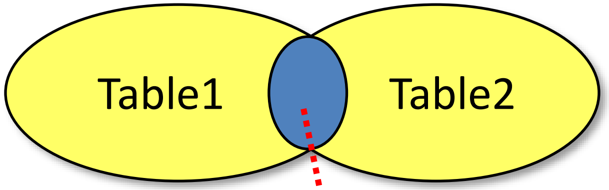
```
SELECT *
FROM Table1
JOIN Table2;
```



Join on condition

```
SELECT * FROM Table1
INNER JOIN    Table2
ON Table1.ID = Table2.ID;
```

- **Join == Inner Join == intersection**
- **Inner join => each row in left with matching row in right**



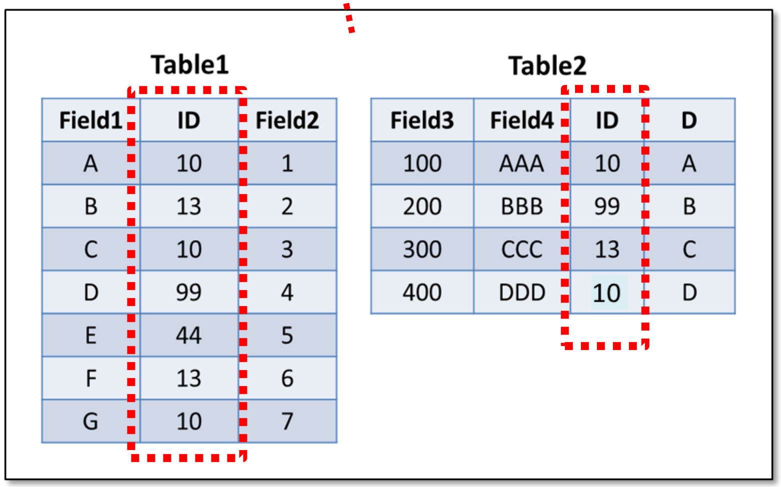


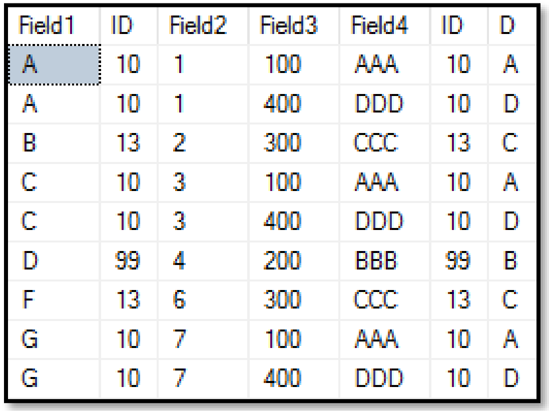
Table1

Field1	ID	Field2
A	10	1
B	13	2
C	10	3
D	99	4
E	44	5
F	13	6
G	10	7

Table2

Field3	Field4	ID	D
100	AAA	10	A
200	BBB	99	B
300	CCC	13	C
400	DDD	10	D

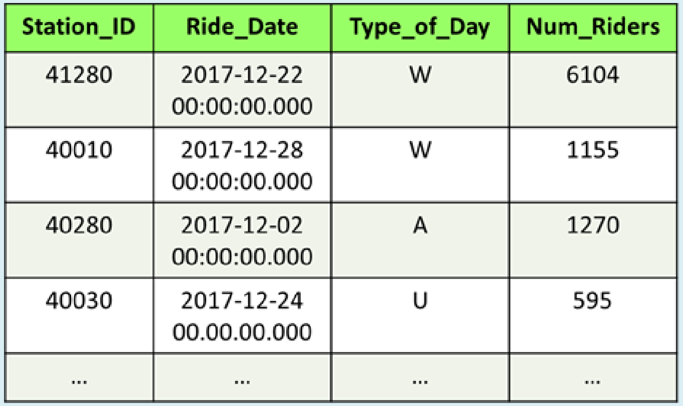
Beware: sometimes you get multiple matches, or none



Field1	ID	Field2	Field3	Field4	ID	D
A	10	1	100	AAA	10	A
A	10	1	400	DDD	10	D
B	13	2	300	CCC	13	C
C	10	3	100	AAA	10	A
C	10	3	400	DDD	10	D
D	99	4	200	BBB	99	B
F	13	6	300	CCC	13	C
G	10	7	100	AAA	10	A
G	10	7	400	DDD	10	D

Example

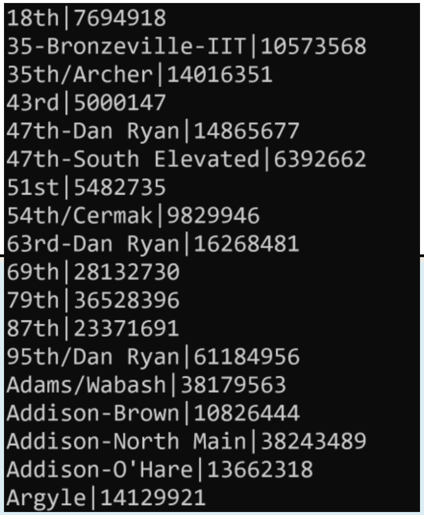
- What is the total # of riders through each station on weekdays, with station names not ids?



Station_ID	Ride_Date	Type_of_Day	Num_Riders
41280	2017-12-22 00:00:00.000	W	6104
40010	2017-12-28 00:00:00.000	W	1155
40280	2017-12-02 00:00:00.000	A	1270
40030	2017-12-24 00.00.00.000	U	595
...	...	...	...

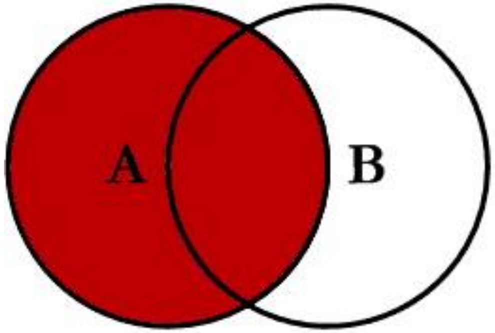
Ridership

```
select    Station_Name, Sum(Num_Riders)
from      Ridership
inner join Stations on Stations.Station_ID = Ridership.Station_ID
where     Type_of_Day = 'W'
group by Stations.Station_ID
order by Station_Name;
```



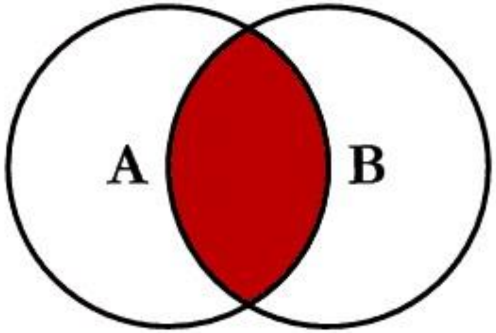
```
18th|7694918
35-Bronzeville-IIT|10573568
35th/Archer|14016351
43rd|5000147
47th-Dan Ryan|14865677
47th-South Elevated|6392662
51st|5482735
54th/Cermak|9829946
63rd-Dan Ryan|16268481
69th|28132730
79th|36528396
87th|23371691
95th/Dan Ryan|61184956
Adams/Wabash|38179563
Addison-Brown|10826444
Addison-North Main|38243489
Addison-O'Hare|13662318
Argyle|14129921
```

SQL JOINS

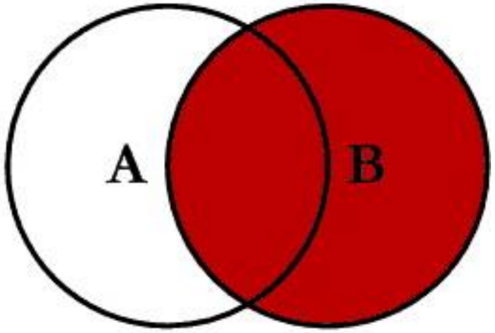


```
SELECT <select_list>
FROM TableA A
LEFT JOIN TableB B
ON A.Key = B.Key
```

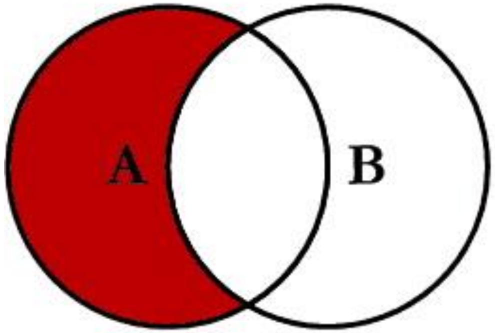
inner join => intersection



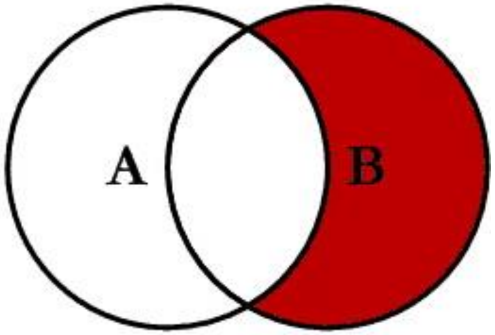
```
SELECT <select_list>
FROM TableA A
INNER JOIN TableB B
ON A.Key = B.Key
```



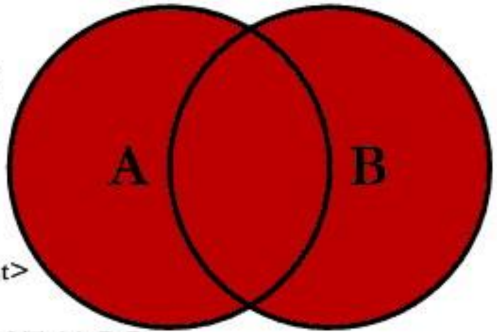
```
SELECT <select_list>
FROM TableA A
RIGHT JOIN TableB B
ON A.Key = B.Key
```



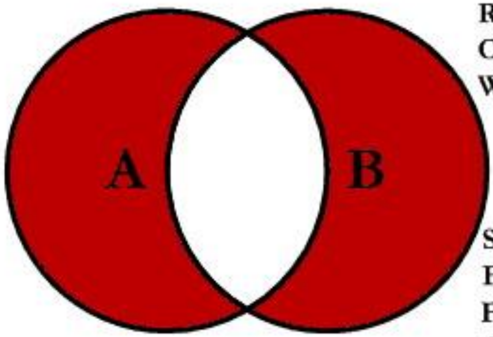
```
SELECT <select_list>
FROM TableA A
LEFT JOIN TableB B
ON A.Key = B.Key
WHERE B.Key IS NULL
```



```
SELECT <select_list>
FROM TableA A
RIGHT JOIN TableB B
ON A.Key = B.Key
WHERE A.Key IS NULL
```

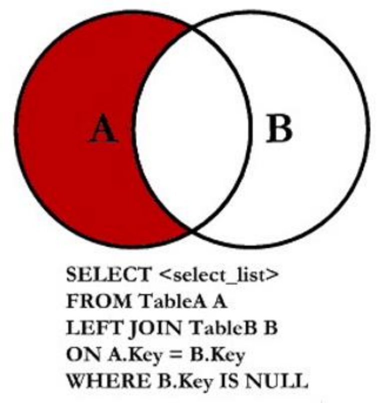


```
SELECT <select_list>
FROM TableA A
FULL OUTER JOIN TableB B
ON A.Key = B.Key
```

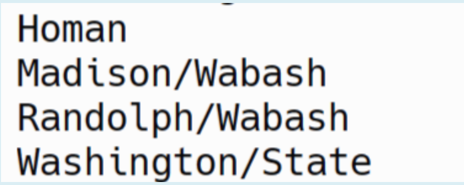


```
SELECT <select_list>
FROM TableA A
FULL OUTER JOIN TableB B
ON A.Key = B.Key
WHERE A.Key IS NULL
OR B.Key IS NULL
```

Question: which stations have no stops?



```
SELECT <select_list>
FROM TableA A
LEFT JOIN TableB B
ON A.Key = B.Key
WHERE B.Key IS NULL
```



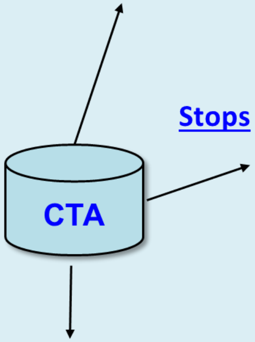
```
Homan
Madison/Wabash
Randolph/Wabash
Washington/State
```

Stations

Station_ID	Station_Name
40710	*Chicago/Franklin*
…	…

Stops

Stop_ID	Station_ID	Stop_Name	Direction	ADA	Latitude	Longitude
30137	*40710*	*Chicago (Kimball-Linden-bound)*	*N*	*1*	*41.89681*	*-87.635924*
30138	*40710*	*Chicago (Loc bound)*				
…	…	…				



Ridership

Station_ID	Ride_Date	Type_of_Day	Num_Riders
…	…	…	…
40710	2001-02-28 00:00:00.000	W	4206
…	…	…	…

```
Select  Station_Name
From    Stations
Left Join Stops
        On Stations.Station_ID = Stops.Station_ID
Where      Stops.Station_ID IS NULL
Order by   Station_Name;
```

That's it, thank you!