

10 SQL Queries waste collection-Outputs.

1. Listar todas las estaciones recolectoras en orden alfabético. El resultado debe presentar el stationId, stationname.

```
select stationid, stationname FROM "Collection".dimstation  
order by stationname
```

Output:

	stationid integer	stationname character varying
1	77	Brasilia
2	40	Brasilia
3	43	Brasilia
4	46	Brasilia
5	97	Brasilia
6	71	Brasilia
7	44	Rio de Janeiro
8	31	Rio de Janeiro
9	32	Rio de Janeiro
10	84	Rio de Janeiro
11	86	Rio de Janeiro
12	57	Rio de Janeiro
13	48	Salvador
14	47	Salvador
15	81	Salvador
16	82	Salvador
17	83	Sao Paulo
18	19	Sao Paulo
19	21	Sao Paulo

2. Listar todos los viajes que recolectaron una cantidad de residuos > 40. El resultado debe presentar el tripid, Waste.

```
select tripid, waste FROM "Collection".facttrips  
where waste > 40
```

Output:

	tripid integer	waste real
1	23480	40.33
2	23481	41.92
3	23483	43.56
4	23486	41.89
5	23488	43.25
6	23498	43.11
7	23499	43
8	23504	41.67
9	23508	44.95
10	23509	44.53
11	23510	43.42
12	23512	43.2
13	23514	43.57
14	23516	43.23
15	23528	41.02
16	23529	43.95
17	23532	44.6
18	23542	44.83
19	23545	40.59
20	23546	40.06
21	23551	44.11
22	23552	41.06
23	23557	42.04
24	23563	43.04
Total rows: 1000 of 17181		

3. Cuento la cantidad de resultados de la anterior consulta.

```
SELECT count(*) as CantTripsOver40 FROM "Collection".facttrips  
WHERE waste > 40
```

Output:

		canttripsover40	
		bigint	
1		17181	

4. Listar el promedio de residuos recolectados cada fecha. El resultado debe contener dateid, average waste.

```
select dateid, avg(waste) as Recolec_Prom FROM "Collection".facttrips  
group by dateid  
order by dateid
```

Output:

	dateid integer	recolec_prom double precision
1	1	37.4498025492618
2	2	37.13526307281695
3	3	37.715756579449305
4	4	36.995098741430986
5	5	37.344046097052725
6	6	37.44200657543383
7	7	37.61592105815285
8	8	37.293585419654846
9	9	37.620953911229186
10	10	37.640329009608216
11	11	37.57217117359764
12	12	37.510953865553205
13	13	37.116381551090036
14	14	37.13440792184127
15	15	37.58819080026526
16	16	37.21138152950689
17	17	37.05243424365395
18	18	37.640394825684396
19	19	37.51713824899573
20	20	37.27898027394947
21	21	37.860921131937125
22	22	37.293881604545994
23	23	37.34440790352068
24	24	37.50049344489449
Total rows: 172 of 172		

5. Listar los nombres de las marcas de los camiones recolectores y la cantidad que hay de cada uno. La salida debe contener truckName, count.

```
select truckname, count(truckname) from "Collection".dimtruck  
group by truckname
```

Output:

	truckname character varying	count bigint
1	Volvo	32
2	Scania	39

6. Listar cada ciudad con el total de residuos recolectados en ella. El resultado debe contener cityname, total_Waste

Acá tenemos que traer información de dos tablas: dimstation y facttrips. Luego el uso de join es aplicable acá. El campo común entre ambas tablas es el stationid

```
select s.stationname as CityName, sum(f.waste) as total_waste  
from "Collection".dimstation s  
left join "Collection".facttrips f on s.stationid=f.stationid  
group by CityName
```

Output:

	cityname character varying	total_waste real
1	Salvador	487086.4
2	Sao Paulo	488028.06
3	Brasilia	487831.6
4	Rio de Janeiro	487199.84

7. ¿Cantidad mínima de residuos recolectados por trimestre en 2019? El resultado debe contener QuarterName, minimum waste.

Para responder la pregunta, se deben consultar las tablas: dimdate y facttrips. Se usará JOIN. El campo común entre ambas tablas es dateid

```
select d.quartername, min(f.waste) as min_waste_per_quarter_in_2019
from "Collection".facttrips f
left join "Collection".dimdate d on d.dateid=f.dateid
where year=2019
group by d.quartername
order by d.quartername
```

Output:

	quartername character varying	min_waste_per_quarter_in_2019 real
1	Q1	30
2	Q2	30
3	Q3	30

8. ¿Cantidad máxima de residuos recolectados en el trimestre 1 en Sao Paulo? El resultado debe contener quarterName, City, maximum Waste.

Para responder la pregunta, se deben consultar tres tablas: dimdate, dimstation y facttrips. Se usará JOIN. El campo común entre la tabla facttrips y dimdate es dateid, y entre facttrips y dimstation es stationid.

```
select s.stationname as City, d.quartername, max(f.waste) as max_waste
from "Collection".facttrips f
left join "Collection".dimdate d on d.dateid=f.dateid
left join "Collection".dimstation s on s.stationid=f.stationid
where d.quartername='Q1' and s.stationname ='Sao Paulo'
group by s.stationname, d.quartername
```

Output:

	city character varying	quartername character varying	max_waste real
1	Sao Paulo	Q1	45

9. Listar los días de la semana en los que los camiones Volvo recogen una media de residuos más alta. El resultado debe contener WeekDayName, TruckName, avg_Waste.

Para responder la pregunta, se deben consultar tres tablas: facttrips, dimdate y dimtruck. Se usará JOIN. El campo común entre la tabla facttrips y dimdate es dateid, y entre facttrips y dimtruck es truckid.

```
select d.weekdayname, t.truckname, avg(f.waste) as avg_waste
from "Collection".facttrips f
left join "Collection".dimtruck t on f.truckid = t.truckid
left join "Collection".dimdate d on f.dateid=d.dateid
where t.truckname='Volvo'
group by d.weekdayname, t.truckname
order by avg_waste desc
```

Output:

	weekdayname character varying	truckname character varying	avg_waste double precision
1	Saturday	Volvo	37.56371518588771
2	Thursday	Volvo	37.5634078635087
3	Monday	Volvo	37.54559559513811
4	Friday	Volvo	37.527673218046544
5	Wednesday	Volvo	37.50538820208925
6	Tuesday	Volvo	37.498569872702056
7	Sunday	Volvo	37.47507110466321

10. Encontrar el tripid para el cual se dio el máximo de residuos recolectados en cada dateid.

```
select tripid, dateid, waste from "Collection".facttrips
where (dateid, waste) in (
select dateid, max(waste) from "Collection".facttrips
group by dateid)
```

Output:

	tripid integer	dateid integer	waste real
1	23508	1	44.95
2	24043	2	44.9
3	24146	3	44.97
4	24583	4	45
5	24834	5	45
6	25154	6	44.97
7	25381	7	44.99
8	25899	8	44.96
9	26131	9	44.94
10	26508	10	44.96
11	26774	11	44.95
12	27093	12	44.87
13	27125	13	44.97
14	27607	14	44.96
15	27886	15	44.95
16	28084	16	44.97
17	28459	17	44.91
18	28578	17	44.91
19	28731	18	45
20	29023	19	44.97
21	29317	20	44.97
22	29567	21	45
23	29989	22	44.98
24	30444	23	44.97
Total rows: 190 of 190			

Bonus: Listar las fechas en las que cada ciudad recolectó el máximo de residuos. El resultado debe contener city, date, maximum Waste.

```

Select s.stationName as city, d.datec as date, a.waste
From (
Select stationid, dateid, waste, rank() over (partition by stationId order by waste desc) as rnk
From "Collection".facttrips) a
Left outer join "Collection".dimdate d on a.DateId = d.DateId
Left outer join "Collection".dimstation s on a.stationId = s.stationId
Where a.rnk = 1
order by stationname

```

Output:

	city character varying	datec character varying	waste real
1	Brasilia	13-03-2019	44.99
2	Brasilia	21-07-2019	45
3	Brasilia	01-05-2019	45
4	Brasilia	04-07-2019	45
5	Brasilia	08-05-2019	45
6	Brasilia	29-03-2019	45
7	Brasilia	22-08-2019	45
8	Brasilia	21-06-2019	44.99
9	Brasilia	23-08-2019	45
10	Brasilia	13-05-2019	45
11	Brasilia	27-05-2019	44.99
12	Rio de Janeiro	26-08-2019	45
13	Rio de Janeiro	18-06-2019	45
14	Rio de Janeiro	23-07-2019	45
15	Rio de Janeiro	02-05-2019	45
16	Rio de Janeiro	02-08-2019	45
17	Rio de Janeiro	13-05-2019	45
18	Rio de Janeiro	12-07-2019	45
19	Rio de Janeiro	02-07-2019	45
20	Rio de Janeiro	02-06-2019	45
21	Rio de Janeiro	08-05-2019	45
22	Rio de Janeiro	25-07-2019	45
23	Rio de Janeiro	27-04-2019	45
24	Salvador	21-06-2019	45
25	Salvador	05-05-2019	45
26	Salvador	21-08-2019	45
Total rows: 43 of 43			