## Grupo #3

## Informe de optimización de consultas

## Consultas:

1) ¿Qué tipo de residuo se genera con mayor frecuencia en cada ubicación? Antes de la optimización:

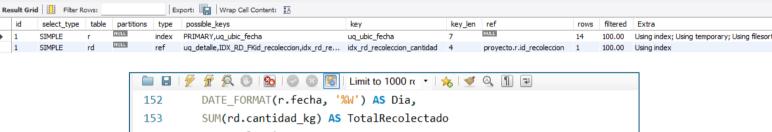
```
-- Consultas
  -- 1) ¿Qué tipo de residuo se genera con mayor frecuencia en cada ubicación?
     ubicación, residuo, Total -- Seleccionar Ubicación, residuo, total
FROM (
      -- Primero hacemos un query para mostrar por cada Ubicación un ranking con el tipo de residuo
     SELECT
         CONCAT(u.nombre, ' ', u.ciudad) AS Ubicacion,
         tr.nombre AS residuo,
         SUM(rd.cantidad kg) AS Total,
         ROW_NUMBER() OVER(PARTITION BY u.id_ubicacion ORDER BY SUM(rd.cantidad_kg) DESC) AS ranking
     FROM
         recoleccion AS r
      INNER JOIN recoleccion_detalle AS rd ON r.id_recoleccion = rd.id_recoleccion
      INNER JOIN ubicacion AS u ON u.id_ubicacion = r.id_ubicacion
      INNER JOIN tipo_residuo AS tr ON tr.id_tipo_residuo = rd.id_tipo_residuo
     GROUP BY
         u.id_ubicacion, tr.nombre
` ) AS T1
  WHERE
     T1.ranking = 1 -- Filtramos solo el primer ranking de cada Ubicación
 ORDER BY
     T1.Total DESC;
```

Result Grid   Export: Here   Wrap Cell Content: 1/4												
	id	select_type	table	partitions	type	possible_keys	key	key_len	ref	rows	filtered	Extra
-	1	PRIMARY	<derived2></derived2>	NULL	ref	<auto_key0></auto_key0>	<auto_key0></auto_key0>	8	const	2	100.00	Using filesort
	2	DERIVED	rd	NULL	ALL	uq_detalle,id_tipo_residuo	NULL	NULL	HULL	23	100.00	Using temporary; Using fileson
	2	DERIVED	u	NULL	eq_ref	PRIMARY	PRIMARY	4	proyecto.rd.id_tipo_residuo	1	100.00	HULL
	2	DERIVED	r	NULL	eq_ref	PRIMARY,uq_ubic_fecha	PRIMARY	4	proyecto.rd.id_recoleccion	1	100.00	HULL
	2	DERIVED	u	NULL	eq_ref	PRIMARY	PRIMARY	4	proyecto.r.id_ubicacion	1	100.00	NULL

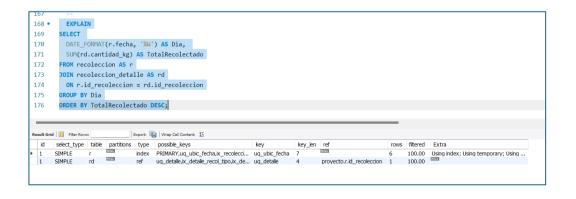
## 2) ¿Qué días de la semana tienen mayor volumen de recolección?

Antes de la optimización:

```
150
       -- 2) ¿Qué días de la semana tienen mayor volumen de recolección?
151 •
152
         DATE_FORMAT(r.fecha, '%W') AS Dia,
         SUM(rd.cantidad kg) AS TotalRecolectado
153
       FROM recoleccion AS r
154
       INNER JOIN recoleccion_detalle AS rd
155
         ON r.id recolection = rd.id recolection
156
157
       GROUP BY Dia
158
       ORDER BY TotalRecolectado DESC;
159
```



```
154
       FROM recoleccion AS r
155
       INNER JOIN recoleccion detalle AS rd
        ON r.id_recoleccion = rd.id_recoleccion
156
       GROUP BY Dia
157
158
       ORDER BY TotalRecolectado DESC;
159
       -- Optimizacion de datos con index
160
161 •
       CREATE INDEX ix_recoleccion_fecha_recol
        ON recoleccion (fecha, id_recoleccion);
162
163
164 •
      CREATE INDEX ix_detalle_recol_cant
         ON recolection_detalle (id_recolection, cantidad_kg);
165
```



3) ¿Qué ubicaciones tienen mayor eficiencia en la separación de residuos reciclables? Antes de la optimización:

```
-- 3) ¿Qué ubicaciones tienen mayor eficiencia en la separación de residuos reciclables?
  -- Ubicaciones con más kilos reciclables
   SELECT
    Ubicacion,

⇒ ROUND (
    SUM(CASE WHEN Clasificacion = 'Reciclable' THEN Cantidad ELSE 0 END)
    / SUM(Cantidad) * 100, 2) AS Tasa Eficiencia -- round
    (
      SELECT CONCAT(u.nombre, ' ', u.ciudad) AS Ubicacion,
        tr.clasificacion AS Clasificacion,
        SUM(rd.cantidad kg) AS Cantidad
        recoleccion AS r
        INNER JOIN recolection_detalle AS rd ON r.id_recolection = rd.id_recolection
        INNER JOIN ubicacion AS u ON u.id ubicacion = r.id ubicacion
        INNER JOIN tipo_residuo AS tr ON tr.id_tipo_residuo = rd.id_tipo_residuo
      GROUP BY
        Ubicacion,
        Clasificacion
    ) AS Totales
  GROUP BY
    Ubicacion
  ORDER BY
    Tasa_Eficiencia DESC;
```

Después de la optimización agregando INDEX:

