

TALLER 2-2 – CATÁLOGO DE UN SMBDR Y SQL

Sebastian Ramirez, Nicolas Sandoval

Universidad de los Andes, Bogotá, Colombia

{s.ramirez112, n.sandovalf}@uniandes.edu.co

Fecha de presentación: Marzo 19 de 2023

DESARROLLO

1. Información general estadística de tablas: Obtener un listado con información de las tablas existentes en la base de datos de Parranderos. El listado debe incluir: El nombre de la tabla, el número de columnas que posee, el número de columnas que pueden tener NULL como valor y la fecha del último análisis hecho por Oracle sobre la tabla, ordenado por el nombre de la tabla

Texto plano:

```
SELECT TABLE_NAME AS NOMBRETABLA,  
       COUNT(*) AS NUMCOLUMNAS,  
       SUM(CASE WHEN NULLABLE='Y' THEN 1 ELSE 0 END) AS NUMCOLSNULL,  
       LAST_ANALYZED AS ÚLTIMO_ANÁLISIS  
FROM USER_TAB_COLUMNS  
WHERE TABLE_NAME IN ('BARES', 'BEBEDORES', 'BEBIDAS', 'FRECUENTAN',  
                       'GUSTAN', 'SIRVEN', 'TIPO_BEBIDA')  
GROUP BY TABLE_NAME, LAST_ANALYZED  
ORDER BY TABLE_NAME;
```

Foto resultado:

	⚡ NOMBRETABLA	⚡ NUMCOLUMNAS	⚡ NUMCOLSNULL	⚡ ÚLTIMO_ANÁLISIS
1	BARES	5	4	11-MAR-23
2	BEBEDORES	4	3	11-MAR-23
3	BEBIDAS	4	3	10-MAR-23
4	FRECUENTAN	4	1	10-MAR-23
5	GUSTAN	2	2	10-MAR-23
6	SIRVEN	3	0	10-MAR-23
7	TIPO_BEBIDA	2	1	10-MAR-23

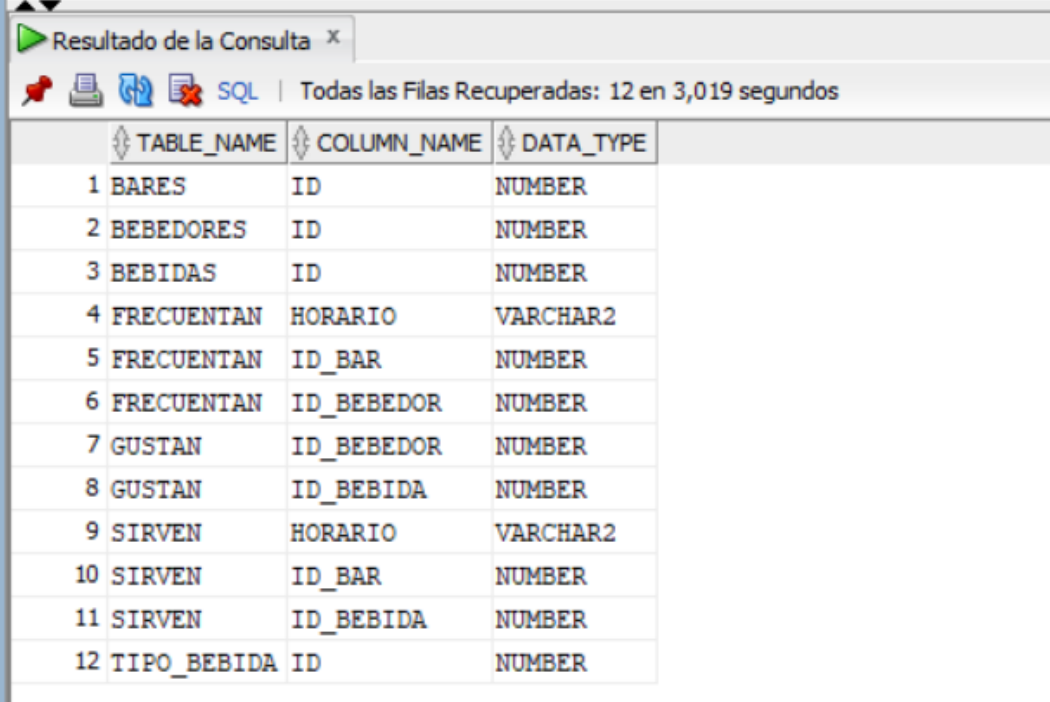
2. Información de las llaves primarias: Para todas las tablas de Parranderos se quiere mostrar el nombre de la tabla y la información de su llave primaria: cuáles son las columnas que la conforman y el tipo de datos de cada una. El resultado debe estar ordenado ascendentemente

por nombre de la tabla y el nombre de la columna. Note que el nombre de la tabla se repite para cada una de las columnas de su llave primaria. El formato esperado se presenta a continuación

Texto plano:

```
SELECT A.table_name, A.column_name, b.data_type
FROM
  (SELECT cols.table_name, cols.column_name
   FROM all_constraints cons, all_cons_columns cols
   WHERE cons.constraint_type = 'P'
   AND cons.constraint_name = cols.constraint_name
   AND cons.owner = cols.owner
   AND cons.owner = 'PARRANDEROS'
   AND cols.CONSTRAINT_NAME LIKE 'PK%'
   ORDER BY cols.table_name)A
JOIN
  (select table_name, column_name, data_type
   from all_tab_columns
   where owner = 'PARRANDEROS') B
ON
  B.table_name = A.table_name
AND A.column_name = B.column_name
order by A.table_name asc,A.column_name asc;
```

Foto resultado:



	TABLE_NAME	COLUMN_NAME	DATA_TYPE
1	BARES	ID	NUMBER
2	BEBEDORES	ID	NUMBER
3	BEBIDAS	ID	NUMBER
4	FRECUENTAN	HORARIO	VARCHAR2
5	FRECUENTAN	ID_BAR	NUMBER
6	FRECUENTAN	ID_BEBEDOR	NUMBER
7	GUSTAN	ID_BEBEDOR	NUMBER
8	GUSTAN	ID_BEBIDA	NUMBER
9	SIRVEN	HORARIO	VARCHAR2
10	SIRVEN	ID_BAR	NUMBER
11	SIRVEN	ID_BEBIDA	NUMBER
12	TIPO_BEBIDA	ID	NUMBER

3. Caracterización de los tipos de datos de las tablas: Para todas las tablas de la aplicación (Parranderos) y para cada tipo de dato que utilice, se quiere mostrar cuántas columnas hay en

la tabla de ese tipo de datos y el promedio de las longitudes de dichas columnas. El resultado debe estar ordenado de forma ascendente por el nombre de la tabla, el nombre del tipo de dato y el número de columnas con ese tipo de dato. Note que el nombre de la tabla se repite para cada tipo de dato que utilice.

Texto plano:

```
SELECT TABLE_NAME AS NOMBRETABLA, DATA_TYPE AS TIPODEDATOS,
COUNT(DATA_TYPE) AS NUMCOLSTIPODATO, ROUND(AVG(avg_col_len),2) AS
PROMEDIOLONGITUDCOL
FROM ALL_TAB_COLUMNS cols
WHERE owner = 'PARRANDEROS'
GROUP BY TABLE_NAME, DATA_TYPE
ORDER BY TABLE_NAME ASC;
```

Foto resultado:

	NOMBRETABLA	TIPODEDATOS	NUMCOLSTIPODATO	PROMEDIOLONGITUDCOL
1	BARES	NUMBER	2	3
2	BARES	VARCHAR2	3	10,33
3	BEBEDORES	VARCHAR2	3	10,67
4	BEBEDORES	NUMBER	1	4
5	BEBIDAS	NUMBER	3	3,33
6	BEBIDAS	VARCHAR2	1	11
7	FRECUELTAN	NUMBER	2	3,5
8	FRECUELTAN	VARCHAR2	1	8
9	FRECUELTAN	DATE	1	8
10	GUSTAN	NUMBER	2	4
11	SIRVEN	VARCHAR2	1	8
12	SIRVEN	NUMBER	2	3,5
13	TIPO_BEBIDA	NUMBER	1	3
14	TIPO_BEBIDA	VARCHAR2	1	9

- Listado general de las columnas y sus restricciones: Para todas las columnas de las tablas de Parranderos se debe mostrar el nombre de la tabla, el nombre de la columna, el tipo de dato de la columna, el nombre de la restricción y si permite nulos o no. El resultado debe estar ordenado de forma ascendente por el nombre de la tabla, el nombre de la columna y el nombre de la restricción. Para las columnas que no tienen restricciones, en el nombre de la restricción debe aparecer “NO TIENE”. Note que para las columnas que tienen más de una restricción, la información de la columna se repite.

Texto plano:

```
SELECT COLS.TABLE_NAME AS NOMBRETABLA,
       COLS.COLUMN_NAME AS NOMBRECOLUMNA,
       COLS.DATA_TYPE AS TIPODEDATO,
       DECODE(COLS.NULLABLE, 'Y', 'SI', 'N', 'NO') AS PERMITENULO,
       DECODE(CON.CONSTRAINT_NAME, NULL, 'NO TIENE',
CON.CONSTRAINT_NAME) AS NOMBREERESTRICCION
```

```

FROM USER_TAB_COLUMNS COLS
LEFT JOIN (
    SELECT UC.TABLE_NAME, UC.COLUMN_NAME, UC.CONSTRAINT_NAME
    FROM USER_CONS_COLUMNS UC
    JOIN USER_CONSTRAINTS C ON UC.CONSTRAINT_NAME =
C.CONSTRAINT_NAME AND C.CONSTRAINT_TYPE = 'C') CON
ON COLS.TABLE_NAME = CON.TABLE_NAME AND COLS.COLUMN_NAME =
CON.COLUMN_NAME
WHERE COLS.TABLE_NAME IN ('BARES', 'BEBEDORES', 'BEBIDAS',
'FRECUENTAN', 'GUSTAN', 'SIRVEN', 'TIPO_BEBIDA')
ORDER BY COLS.TABLE_NAME, COLS.COLUMN_NAME,
CON.CONSTRAINT_NAME;

```

Foto resultado:

	NOMBRETABLA	NOMBRECOLUMNA	TIPODEDATO	PERMITENULO	NOMBRERESTRICCION
1	BARES	CANT_SEDES	NUMBER	SI	CK_CANTSEDES
2	BARES	CIUDAD	VARCHAR2	SI	NO TIENE
3	BARES	ID	NUMBER	NO	NO TIENE
4	BARES	NOMBRE	VARCHAR2	SI	NO TIENE
5	BARES	PRESUPUESTO	VARCHAR2	SI	CK_PRESUPUESTO
6	BEBEDORES	CIUDAD	VARCHAR2	SI	NO TIENE
7	BEBEDORES	ID	NUMBER	NO	NO TIENE
8	BEBEDORES	NOMBRE	VARCHAR2	SI	NO TIENE
9	BEBEDORES	PRESUPUESTO	VARCHAR2	SI	NO TIENE
10	BEBIDAS	GRADO_ALCOHOL	NUMBER	SI	CK_GRADOALCHL
11	BEBIDAS	ID	NUMBER	NO	NO TIENE
12	BEBIDAS	NOMBRE	VARCHAR2	SI	NO TIENE
13	BEBIDAS	TIPO	NUMBER	SI	NO TIENE
14	FRECUENTAN	FECHA_ULTIMA_VISITA	DATE	SI	NO TIENE
15	FRECUENTAN	HORARIO	VARCHAR2	NO	CK_HORARIO
16	FRECUENTAN	ID_BAR	NUMBER	NO	NO TIENE
17	FRECUENTAN	ID_BEBEDOR	NUMBER	NO	NO TIENE
18	GUSTAN	ID_BEBEDOR	NUMBER	SI	NO TIENE
19	GUSTAN	ID_BEBIDA	NUMBER	SI	NO TIENE
20	SIRVEN	HORARIO	VARCHAR2	NO	CK_S_HORARIO
21	SIRVEN	ID_BAR	NUMBER	NO	NO TIENE
22	SIRVEN	ID_BEBIDA	NUMBER	NO	NO TIENE
23	TIPO_BEBIDA	ID	NUMBER	NO	NO TIENE
24	TIPO_BEBIDA	NOMBRE	VARCHAR2	SI	NO TIENE

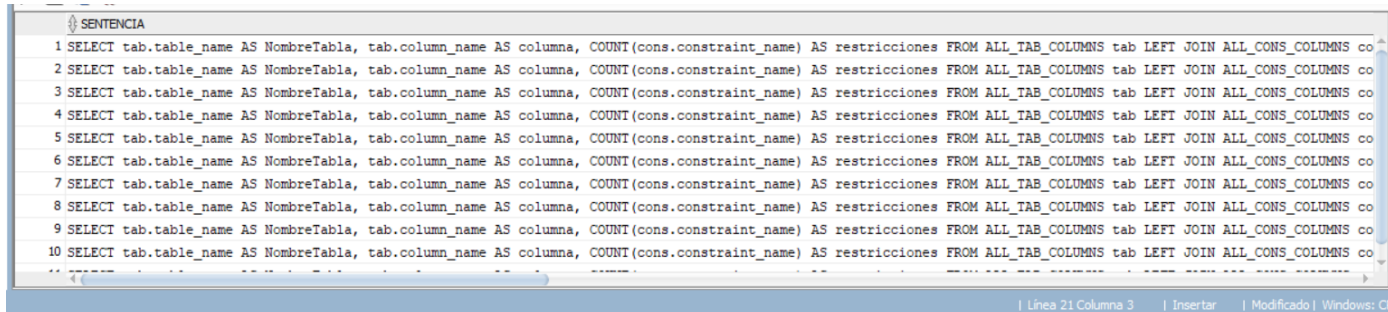
5. Se quiere tener un listado del subconjunto de columnas de la base de datos de Parranderos, cuyo nombre no contiene la letra “R” y que tiene máximo 2 vocales, indicando para cada columna su nombre, tabla a la que pertenece y el número de restricciones que la afectan (por ejemplo, si una columna es parte de la llave primaria de la tabla y tiene una restricción de llave foránea, el resultado en esta columna debe ser mayor o igual a dos (2)) Este listado puede ser solicitado en cualquier momento y por lo tanto debe generarse de forma semiautomática, mediante el siguiente proceso:

a. Mediante UNA consulta al catálogo, obtener un listado donde, para cada columna de Parranderos que debe ser incluida en el resultado, hay una fila de respuesta. Cada fila de respuesta de la consulta anterior contiene una sentencia SQL que permite encontrar la información solicitada para la columna correspondiente, terminado con punto y coma (;). Este listado contiene las sentencias SQL que se requieren para la segunda etapa.

Texto plano:

```
SELECT
'SELECT tab.table_name AS NombreTabla, tab.column_name AS columna,
COUNT(cons.constraint_name) AS restricciones
FROM ALL_TAB_COLUMNS tab
LEFT JOIN ALL_CONS_COLUMNS cons
  ON tab.COLUMN_NAME = cons.COLUMN_NAME AND tab.table_name =
cons.table_name
WHERE tab.owner = "PARRANDEROS"
AND tab.table_name = "||NOMBRETABLA||"
AND tab.column_name = "||COLUMNA||"
AND LENGTH(tab.column_name) >6
GROUP BY tab.table_name, tab.column_name
ORDER BY NOMBRETABLA' AS SENTENCIA
FROM(
  SELECT tab.table_name AS NombreTabla, tab.column_name AS columna,
COUNT(cons.constraint_name) AS restricciones
  FROM ALL_TAB_COLUMNS tab
  LEFT JOIN ALL_CONS_COLUMNS cons
    ON tab.COLUMN_NAME = cons.COLUMN_NAME AND tab.table_name =
cons.table_name
  WHERE tab.owner = 'PARRANDEROS'
  AND LENGTH(tab.column_name) >6
  GROUP BY tab.table_name, tab.column_name
  ORDER BY NOMBRETABLA
);
```

Foto resultado:



b. Utilizar el listado de la etapa a) como un script, cuya ejecución devuelve un nuevo listado donde cada fila contiene la información solicitada de las columnas solicitadas de Parranderos.

```
SELECT tab.table_name AS NombreTabla, tab.column_name AS columna,
COUNT(cons.constraint_name) AS restricciones
FROM ALL_TAB_COLUMNS tab
LEFT JOIN ALL_CONS_COLUMNS cons
  ON tab.COLUMN_NAME = cons.COLUMN_NAME AND tab.table_name =
cons.table_name
WHERE tab.owner = 'PARRANDEROS'
AND tab.table_name = 'BARES'
AND tab.column_name = 'CANT_SEDES'
AND LENGTH(tab.column_name) > 6
GROUP BY tab.table_name, tab.column_name
ORDER BY NOMBRETABLA;
```

```
SELECT tab.table_name AS NombreTabla, tab.column_name AS columna,
COUNT(cons.constraint_name) AS restricciones
FROM ALL_TAB_COLUMNS tab
LEFT JOIN ALL_CONS_COLUMNS cons
  ON tab.COLUMN_NAME = cons.COLUMN_NAME AND tab.table_name =
cons.table_name
WHERE tab.owner = 'PARRANDEROS'
AND tab.table_name = 'BARES'
AND tab.column_name = 'PRESUPUESTO'
AND LENGTH(tab.column_name) > 6
GROUP BY tab.table_name, tab.column_name
ORDER BY NOMBRETABLA;
```

```
SELECT tab.table_name AS NombreTabla, tab.column_name AS columna,
COUNT(cons.constraint_name) AS restricciones
FROM ALL_TAB_COLUMNS tab
LEFT JOIN ALL_CONS_COLUMNS cons
```

```

        ON tab.COLUMN_NAME = cons.COLUMN_NAME AND tab.table_name =
cons.table_name
    WHERE tab.owner = 'PARRANDEROS'
    AND tab.table_name = 'BEBEDORES'
    AND tab.column_name = 'PRESUPUESTO'
    AND LENGTH(tab.column_name) > 6
    GROUP BY tab.table_name, tab.column_name
    ORDER BY NOMBRETABLA;

```

```

SELECT tab.table_name AS NombreTabla, tab.column_name AS columna,
COUNT(cons.constraint_name) AS restricciones
FROM ALL_TAB_COLUMNS tab
LEFT JOIN ALL_CONS_COLUMNS cons
    ON tab.COLUMN_NAME = cons.COLUMN_NAME AND tab.table_name =
cons.table_name
    WHERE tab.owner = 'PARRANDEROS'
    AND tab.table_name = 'BEBIDAS'
    AND tab.column_name = 'GRADO_ALCOHOL'
    AND LENGTH(tab.column_name) > 6
    GROUP BY tab.table_name, tab.column_name
    ORDER BY NOMBRETABLA;

```

```

SELECT tab.table_name AS NombreTabla, tab.column_name AS columna,
COUNT(cons.constraint_name) AS restricciones
FROM ALL_TAB_COLUMNS tab
LEFT JOIN ALL_CONS_COLUMNS cons
    ON tab.COLUMN_NAME = cons.COLUMN_NAME AND tab.table_name =
cons.table_name
    WHERE tab.owner = 'PARRANDEROS'
    AND tab.table_name = 'FRECUENTAN'
    AND tab.column_name = 'FECHA_ULTIMA_VISITA'
    AND LENGTH(tab.column_name) > 6
    GROUP BY tab.table_name, tab.column_name
    ORDER BY NOMBRETABLA;

```

```

SELECT tab.table_name AS NombreTabla, tab.column_name AS columna,
COUNT(cons.constraint_name) AS restricciones
FROM ALL_TAB_COLUMNS tab
LEFT JOIN ALL_CONS_COLUMNS cons
    ON tab.COLUMN_NAME = cons.COLUMN_NAME AND tab.table_name =
cons.table_name
    WHERE tab.owner = 'PARRANDEROS'
    AND tab.table_name = 'FRECUENTAN'
    AND tab.column_name = 'HORARIO'
    AND LENGTH(tab.column_name) > 6
    GROUP BY tab.table_name, tab.column_name
    ORDER BY NOMBRETABLA;

```

```

SELECT tab.table_name AS NombreTabla, tab.column_name AS columna,
COUNT(cons.constraint_name) AS restricciones
FROM ALL_TAB_COLUMNS tab
LEFT JOIN ALL_CONS_COLUMNS cons
  ON tab.COLUMN_NAME = cons.COLUMN_NAME AND tab.table_name =
cons.table_name
WHERE tab.owner = 'PARRANDEROS'
AND tab.table_name = 'FRECUENTAN'
AND tab.column_name = 'ID_BEBEDOR'
AND LENGTH(tab.column_name) >6
GROUP BY tab.table_name, tab.column_name
ORDER BY NOMBRETABLA;

```

```

SELECT tab.table_name AS NombreTabla, tab.column_name AS columna,
COUNT(cons.constraint_name) AS restricciones
FROM ALL_TAB_COLUMNS tab
LEFT JOIN ALL_CONS_COLUMNS cons
  ON tab.COLUMN_NAME = cons.COLUMN_NAME AND tab.table_name =
cons.table_name
WHERE tab.owner = 'PARRANDEROS'
AND tab.table_name = 'GUSTAN'
AND tab.column_name = 'ID_BEBEDOR'
AND LENGTH(tab.column_name) >6
GROUP BY tab.table_name, tab.column_name
ORDER BY NOMBRETABLA;

```

```

SELECT tab.table_name AS NombreTabla, tab.column_name AS columna,
COUNT(cons.constraint_name) AS restricciones
FROM ALL_TAB_COLUMNS tab
LEFT JOIN ALL_CONS_COLUMNS cons
  ON tab.COLUMN_NAME = cons.COLUMN_NAME AND tab.table_name =
cons.table_name
WHERE tab.owner = 'PARRANDEROS'
AND tab.table_name = 'GUSTAN'
AND tab.column_name = 'ID_BEBIDA'
AND LENGTH(tab.column_name) >6
GROUP BY tab.table_name, tab.column_name
ORDER BY NOMBRETABLA;

```

```

SELECT tab.table_name AS NombreTabla, tab.column_name AS columna,
COUNT(cons.constraint_name) AS restricciones
FROM ALL_TAB_COLUMNS tab
LEFT JOIN ALL_CONS_COLUMNS cons
  ON tab.COLUMN_NAME = cons.COLUMN_NAME AND tab.table_name =
cons.table_name
WHERE tab.owner = 'PARRANDEROS'
AND tab.table_name = 'SIRVEN'
AND tab.column_name = 'HORARIO'

```



```

AND LENGTH(tab.column_name) > 6
GROUP BY tab.table_name, tab.column_name
ORDER BY NOMBRETABLA;
SELECT tab.table_name AS NombreTabla, tab.column_name AS columna,
COUNT(cons.constraint_name) AS restricciones
FROM ALL_TAB_COLUMNS tab
LEFT JOIN ALL_CONS_COLUMNS cons
ON tab.COLUMN_NAME = cons.COLUMN_NAME AND tab.table_name =
cons.table_name
WHERE tab.owner = 'PARRANDEROS'
AND tab.table_name = 'SIRVEN'
AND tab.column_name = 'ID_BEBIDA'
AND LENGTH(tab.column_name) > 6
GROUP BY tab.table_name, tab.column_name
ORDER BY NOMBRETABLA;

```

	NOMBRETABLA	COLUMNA	RESTRICCIONES
1	BARES	CANT_SEDES	2

	NOMBRETABLA	COLUMNA	RESTRICCIONES
1	BARES	PRESUPUESTO	2

	NOMBRETABLA	COLUMNA	RESTRICCIONES
1	BEBEDORES	PRESUPUESTO	1

	NOMBRETABLA	COLUMNA	RESTRICCIONES
1	BEBIDAS	GRADO_ALCOHOL	2

	NOMBRETABLA	COLUMNA	RESTRICCIONES
1	FRECUELTAN	FECHA_ULTIMA_VISITA	0

	NOMBRETABLA	COLUMNA	RESTRICCIONES
1	FRECUELTAN	HORARIO	4

	NOMBRETABLA	COLUMNA	RESTRICCIONES
1	FRECUELTAN	ID_BEBEDOR	4

	NOMBRETABLA	COLUMNA	RESTRICCIONES
1	GUSTAN	ID_BEBEDOR	4

	NOMBRETABLA	COLUMNA	RESTRICCIONES
1	GUSTAN	ID_BEBIDA	4

	NOMBRETABLA	COLUMNA	RESTRICCIONES
1	SIRVEN	HORARIO	5

	NOMBRETABLA	COLUMNA	RESTRICCIONES
1	SIRVEN	ID_BEBIDA	4