

Taller 2-2: SQL

Juan Felipe Cucalon, Gabriela Caballero
Taller 2-2 sobre consultas SQL
Universidad de los Andes, Bogotá, Colombia
{jf.cucalon , g.caballero}@uniandes.edu.co
Fecha de presentación: Marzo 23 de 2021

Tabla de contenido

1 Consultas al catálogo.....	1
1.1 Información general estadística de tablas.....	1
1.2 Información de las llaves primarias.....	2
1.3 Caracterización de los tipos de datos de las tablas	3
1.4 Listado general de las columnas y sus restricciones	4
2 Consultas sobre Catálogo y Datos.....	5

1. Consultas al catálogo

1.1 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 el número de columnas con restricción de llave foránea, ordenado por el nombre de la tabla. En caso de que la tabla no tenga llaves foráneas y/o columnas con restricciones de nulidad se debe marcar 0 en el campo respectivo.

Texto plano de SQL:

```
With Fks as (SELECT TABLE_NAME, COLUMN_NAME AS NumColsFK
FROM ALL_CONS_COLUMNS
WHERE OWNER = 'PARRANDEROS' AND constraint_name like '%FK%'),
nn as (select c.table_name, c.column_name as NumColsNotNull
from all_tab_columns c
where owner = 'PARRANDEROS' and c.column_name not in (select column_name from
all_cons_columns))

SELECT c.TABLE_NAME as NombreTabla, count(fks.numcolsfk) as NumColsFk,
count(column_name) as NumColumnas, count(nn.numcolsnnull) as NumCollsNull
from all_tab_columns c
full outer join Fks on fks.table_name = c.table_name
full outer join nn on nn.table_name = c.table_name
where c.owner = 'PARRANDEROS'
group by c.table_name;
```

Resultado SQL Developer:

	⚡ NOMBRETABLA	⚡ NUMCOLSFK	⚡ NUMCOLUMNAS	⚡ NUMCOLLSNULL
1	BAPES	0	10	10
2	SIRVEN	6	6	0
3	FRECUELTAN	8	8	8
4	TIPO_BEBIDA	0	2	2
5	BEEDAS	4	4	4
6	GUSTAN	4	4	0
7	BEEDOPES	0	8	8

1.2 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 de SQL:

```
SELECT UNIQUE TC.TABLE_NAME AS NOMBRETABLA, TC.COLUMN_NAME AS NOMBRECOLSPK,
C.DATA_TYPE AS TIPODEDATO
FROM ALL_CONS_COLUMNS TC
INNER JOIN ALL_TAB_COLUMNS C ON C.TABLE_NAME = TC.TABLE_NAME
WHERE C.OWNER = 'PARRANDEROS' AND TC.CONSTRAINT_NAME LIKE '%PK%'
ORDER BY NOMBRETABLA ASC, NOMBRECOLSPK
```

Resultados SQL Developer:

	❖ NOMBRETABLA	❖ NOMBRECOLSPK	❖ TIPODEDATO
1	BARES	ID	NUMBER
2	BARES	ID	VARCHAR2
3	BEBEDORES	ID	NUMBER
4	BEBEDORES	ID	VARCHAR2
5	BEBIDAS	ID	NUMBER
6	BEBIDAS	ID	VARCHAR2
7	FRECUELTAN	HORARIO	DATE
8	FRECUELTAN	HORARIO	NUMBER
9	FRECUELTAN	HORARIO	VARCHAR2
10	FRECUELTAN	ID_BAR	DATE

1.3 Caracterización de los tipos de datos de las tablas

Para todas las tablas de la aplicación 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, 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 de SQL:

```
alter session set current_schema=parranderos;

SELECT TABLE_NAME as NombreTabla ,DATA_TYPE as TipoDeDato, COUNT(*)as
NumColsTipoDato, AVG(AVG_COL_LEN) as PromedioLongitudCol
FROM ALL_TAB_COLUMNS
WHERE OWNER = 'PARRANDEROS'
group by TABLE_NAME, DATA_TYPE
ORDER BY TABLE_NAME;
```

Resultado SQL Developer:

[illegible]

Resultados SQL Developer:

<pre> alter session set current_schema = Parranderos; SELECT C.TABLE_NAME NOMBRETABLA, TC.COLUMN_NAME NOMBRE_COL, TC.DATA_TYPE TIPODATO, C.CONSTRAINT_NAME NOMBRE_RESTR, tc.nullable PERMITE_NULOS FROM ALL_CONSTRAINTS C INNER JOIN ALL_TAB_COLUMNS TC ON C.table_name=tc.table_name WHERE C.OWNER='PARRANDEROS' ORDER BY C.TABLE_NAME, TC.COLUMN_NAME, C.CONSTRAINT_NAME; </pre>				
Salida de Script x Resultado de la Consulta x Se han recuperado 50 filas en 0.273 segundos				
NOMBRETABLA	NOMBRE_COL	TIPODATO	NOMBRE_RESTR	PERMITE_NULOS
1 BARES	CANT_SEDES	NUMBER	CK_CANTSEDES	Y
2 BARES	CANT_SEDES	NUMBER	CK_PRESUPUESTO	Y
3 BARES	CANT_SEDES	NUMBER	PK_BARES	Y
4 BARES	CIUDAD	VARCHAR2	CK_CANTSEDES	Y
5 BARES	CIUDAD	VARCHAR2	CK_PRESUPUESTO	Y
6 BARES	CIUDAD	VARCHAR2	PK_BARES	Y
7 BARES	ID	NUMBER	CK_CANTSEDES	N
8 BARES	ID	NUMBER	CK_PRESUPUESTO	N
9 BARES	ID	NUMBER	PK_BARES	N
10 BARES	NOMBRE	VARCHAR2	CK_CANTSEDES	Y
11 BARES	NOMBRE	VARCHAR2	CK_PRESUPUESTO	Y
12 BARES	NOMBRE	VARCHAR2	PK_BARES	Y
13 BARES	PRESUPUESTO	VARCHAR2	CK_CANTSEDES	Y
14 BARES	PRESUPUESTO	VARCHAR2	CK_PRESUPUESTO	Y
15 BARES	PRESUPUESTO	VARCHAR2	PK_BARES	Y
16 BEBEDORES	CIUDAD	VARCHAR2	PK_BEBEDORES	Y
17 BEBEDORES	ID	NUMBER	PK_BEBEDORES	N
18 BEBEDORES	NOMBRE	VARCHAR2	PK_BEBEDORES	Y
19 BEBEDORES	PRESUPUESTO	VARCHAR2	PK_BEBEDORES	Y

2. Consultas sobre Catálogo y Datos

Se quiere tener un listado del subconjunto de columnas (TABLAS) de la base de datos de Parranderos, cuyo nombre tiene más de seis (6) (3) caracteres (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 semi-automática, mediante el siguiente proceso:

- 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.

- 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.

Texto plano de SQL (a):

```

SELECT CONSULTA1.NOMBRE_COLUMNNA, CONSULTA1.TABLA, CONSULTA1.numero
FROM ((SELECT COLUMN_NAME NOMBRE_COLUMNNA, t.table_name TABLA, count
(c.constraint_name) numero
FROM ALL_TAB_COLUMNS T

```

```

INNER JOIN ALL_CONSTRAINTS C ON c.table_name = t.table_name
WHERE t.owner = 'PARRANDEROS' AND REGEXP_LIKE (CONSTRAINT_NAME,
COLUMN_NAME)
GROUP BY COLUMN_NAME, t.table_name) CONSULTA1)
WHERE REGEXP_LIKE (CONSULTA1.NOMBRE_COLUMNA, '[A-Z] . (([A-Z] [A-Z])) | ([A-
Z] . [A-Z] | [A-Z] . [A-Z] | [A-Z] . [A-Z] | [A-Z] . [A-Z])) ');

```

Resultado SQL Developer (a):

```

SELECT CONSULTA1.NOMBRE_COLUMNA, CONSULTA1.TABLA, CONSULTA1.numero
FROM ((SELECT COLUMN_NAME NOMBRE_COLUMNA, t.table_name TABLA, count (c.constraint_name) numero
FROM ALL_TAB_COLUMNS T
INNER JOIN ALL_CONSTRAINTS C ON c.table_name = t.table_name
WHERE t.owner = 'PARRANDEROS' AND REGEXP_LIKE (CONSTRAINT_NAME, COLUMN_NAME)
GROUP BY COLUMN_NAME, t.table_name) CONSULTA1)
WHERE REGEXP_LIKE (CONSULTA1.NOMBRE_COLUMNA, '[A-Z] . (([A-Z] [A-Z])) | ([A-Z] . [A-Z] | [A-Z] . [A-Z] | [A-Z] . [A-Z] | [A-Z] . [A-Z])) ');

```

Salida de Script x Resultado de la Consulta x		
SQL Todas las Filas Recuperadas: 4 en 0.224 segundos		
NOMBRE_COLUMNA	TABLA	NUMERO
1 TIPO	BEBIDAS	1
2 HORARIO	SIRVEN	1
3 PRESUPUESTO	BARES	1
4 HORARIO	FRECIENTAN	1

Texto plano de SQL (b):

@\Users\gabycaballero\Documents\Universidad\Sistrans\Clausula5a

Resultado SQL Developer (b):

@\Users\gabycaballero\Documents\Universidad\Sistrans\Clausula5a		
Salida de Script x		
Tarea terminada en 0.141 segundos		
NOMBRE_COLUMNA	TABLA	NUMERO
TIPO	BEBIDAS	1
HORARIO	SIRVEN	1
PRESUPUESTO	BARES	1
HORARIO	FRECIENTAN	1