

1. Construya una consulta SQL que dé respuesta a la necesidad de datos expresada con la pregunta «¿Cuáles son las canciones de mayor y menor duración por cada género?».

R/

```
select idgénero, min(duración) as Duracion_Minima, max(duración) as duracion_maxima
from canciones
group by idgénero
order by idgénero asc
```

2. Explique cuál es la necesidad de datos que se busca satisfacer con la consulta presentada en el siguiente script:

```
SELECT idCanción,
       AVG(calificación),
       COUNT(idUsuario)
FROM Calificaciones
GROUP BY idCanción
HAVING AVG(calificación) > 3
```

R/ La necesidad de datos que se busca satisfacer con la consulta es la de: obtener el ID de la canción, el promedio de las calificaciones y la cantidad de usuarios que las escuchan de las canciones cuyo promedio sea mayor a 3.

3. Construya una consulta SQL que dé respuesta a la necesidad de datos expresada con la pregunta «¿Cuántos usuarios residen en el mismo país en el que nacieron?».

R/

```
select count(*)
from usuarios
where paísnacimiento = "paísresidencia"
```

4. Explique cuál es la necesidad de datos que se busca satisfacer con la consulta presentada en el siguiente script:

```
SELECT idIntérpretePrincipal,
       COUNT(*)
FROM Canciones
GROUP BY idIntérpretePrincipal
HAVING COUNT(DISTINCT idioma) > 1
```

R/ Cual es el ID del interprete principal y la cantidad de canciones que tiene dicho interprete siempre que tengan canciones en más de 1 idioma.

5. Construya una consulta SQL que dé respuesta a la necesidad de datos expresada con la pregunta «¿Cuál es promedio del tempo en pulsos de las canciones de cada intérprete por cada uno de sus álbumes?».

```
select avg(tempobpm), idintérpreteprincipal, idÁlbumoriginal
from canciones
group by idÁlbumoriginal, idintérpreteprincipal
order by "idintérpreteprincipal" asc
```

6. Modifique la consulta presentada en el siguiente script de modo que se mejore la forma en que se presenta la tabla resultante:

```
SELECT idGénero,
       idioma,
       COUNT(*),
       MIN(tempoBPM),
       MAX(tempoBPM)
FROM Canciones
GROUP BY idGénero,
         idioma
HAVING COUNT(*) > 1
ORDER BY idGénero,
         idioma
```

```
SELECT idGénero, idioma,
       COUNT(*) as "Cantidad canciones", MIN(tempoBPM) as "Tempo BPM MINIMO", MAX(tempoBPM) as "Tempo BPM MAX"
FROM Canciones
GROUP BY idGénero, idioma
HAVING COUNT(*) > 1
ORDER BY idGénero, idioma
```

7. Responda las preguntas planteadas en los siguientes literales tomando como base las dos consultas presentadas a continuación:

```
SELECT idIntérpretePrincipal,
       COUNT(*)
FROM Canciones
WHERE idIntérpretePrincipal <> 1
      AND esInstrumental <> '1'
      AND idGénero NOT IN (1, 2)
GROUP BY idIntérpretePrincipal
HAVING COUNT(*) > 1
ORDER BY COUNT(*) DESC
```

```
SELECT idIntérpretePrincipal,
       COUNT(*)
FROM Canciones
WHERE idGénero > 2
      AND esInstrumental IN ('0')
GROUP BY idIntérpretePrincipal
HAVING COUNT(*) > 1
      AND idIntérpretePrincipal <> 1
ORDER BY COUNT(*) DESC
```

- ¿Se obtiene el mismo resultado?
  - No, Se obtienen resultados distintos
- ¿Son equivalentes?
  - Si, Son equivalentes con los datos que entrega
- ¿Cuál de las dos le parece mejor? ¿Por qué?
  - Me parece mejor la segunda respuesta planteada, porque nos entrega los resultados de manera ordenada.
- ¿Qué mejora le haría a la que le parece mejor?
  - Le cambiaría pondría alias a ambos para que se mire mejor

```
SELECT idIntérpretePrincipal as "ID del interprete",
       COUNT(*) as "Cantidad"
FROM Canciones
WHERE idGénero > 2
      AND esInstrumental IN ('0')
GROUP BY idIntérpretePrincipal
HAVING COUNT(*) > 1
      AND idIntérpretePrincipal <> 1
ORDER BY COUNT(*) DESC
```

8. Construya una consulta SQL que dé respuesta a la necesidad de datos expresada con la pregunta «¿Cuántas veces por cada día de la semana se reproduce cada una de las canciones?». Mostrar las canciones que tienen al menos tres reproducciones cada día y listar los resultados ordenados de la canción con más reproducciones a la canción con menos reproducciones.

```
select idcanción, count(*) as "cantidad reproducciones"
from reproducciones group by idcanción having count(*) > 3 order by count(*) desc
```

9. Explique cuál es la necesidad de datos que se busca satisfacer con la consulta presentada en el siguiente script:

```
SELECT EXTRACT("year" FROM fechaReproducción) año,
       EXTRACT("month" FROM fechaReproducción) mes,
       idCanción, COUNT(*), COUNT(DISTINCT idUsuario),
       AVG(segundosReproducidos)
FROM Reproducciones
WHERE segundosReproducidos > 30
GROUP BY año, mes, idCanción
ORDER BY año DESC, mes DESC, COUNT(*) DESC,
        COUNT(DISTINCT idUsuario) DESC
```

R/ La necesidad que busca satisfacer esta consulta es: cual es el año, el mes, el id, el número de canciones, el número de canciones distintas por usuario y el promedio de los segundos reproducidos de las canciones cuyos segundos reproducidos son mayores a 30 ordenadas descendentemente

10. Utilizando la nueva versión de la base de datos de la colección de canciones presentada en este capítulo, defina cinco preguntas que le permitan construir cinco consultas SQL en las que se utilicen los datos de una sola tabla. En todas debe aplicarse agrupamiento con filtrado basado en los datos obtenidos con funciones de agregación, filtrado de filas y ordenamiento de resultados
- ¿Cual es el promedio de los las calificaciones de los usuarios para las canciones con id mayor a 5?
  - ¿Cuantos usuarios hay regritados con fecha de registro posterior al inicio de los 2000's?
  - ¿Cuales son las canciones que tienen la mayor y menor duracion en cada idioma?
  - ¿ Cual es la suma de los usuarios de cada pais?
  - ¿Cual es el año de mas reproducciones para cada cancion?