SQL exercises

Please use Google Chrome on a PC

Connecting to

Ex.1 - Consider the [schema](#) of the Chinook database. Is it possible for albums not to be associated with tracks?

- ◉ Yes, because the Album table has no attribute referencing the primary key of the Track table.
- ○ No, because integrity constraints are bidirectional.
- ○ No, because entity integrity constraints do not allow tables to have null values for foreign key attributes.
- ○ Yes, the TrackId attribute of the Album table can be NULL.

Ex.2 - Consider the following query (The TIMESTAMPDIFF returns the difference in years between two dates. See http://dev.mysql.com/doc/refman/5.7/en/date-and-time-functions.html#function_timestampdiff).
Which statement better describes the meaning of the query?

click [here](#) for the schema

```sql
SELECT c.FirstName, c.LastName, c.Country
FROM Chinook.customer AS c, Chinook.employee AS e
WHERE c.supportRepId = e.employeeId
AND TIMESTAMPDIFF(year, birthdate, hiredate) < 35
```

- ○ List the names and the countries of those employees that took less than 35 years to answer a customers.
- ○ List the names and the countries of those customers who are supported by an employee younger than 35.
- ◉ List the names and the countries of those customers who are supported by an employee who was younger than 35 when hired.
- ○ List the names and the countries of those customers who are supported by an employee hired less than 35 years ago.

Ex.3 - Retrieve the name of Artists that published an album having a 3-letter title that contains a 2-letters song

click here for the schema

```
1  SELECT DISTINCT Artist.Name
2  FROM Artist JOIN Album ON Artist.ArtistId = Album.ArtistId
3  JOIN Track ON Album.AlbumId = Track.AlbumId
4  WHERE Album.Title LIKE "___" AND Track.Name LIKE "__"
```

Ex.4 - Retrieve the name of all the pair of artists that published an album containing the number 5 in the title

click here for the schema

```
1  SELECT ar1.name, ar2.name
2  FROM Album AS al1, Album AS al2, Artist AS ar1, Artist AS ar2
3  WHERE al1.title LIKE "%5%" and  al2.title LIKE "%5%"
4  AND al1.AlbumId < al2.AlbumId
5  AND al1.ArtistId = ar1.ArtistId AND al2.ArtistId = ar2.ArtistId
```

Ex.5 - Retrieve the name of composers of classical songs longer than 5 minutes

click here for the schema

```
1  SELECT DISTINCT Composer
2  FROM track JOIN Genre ON track.GenreId = Genre.GenreId
3  WHERE Genre.Name = "Classical" AND track.Milliseconds &gt; 300000
4  ORDER BY track.Milliseconds DESC
```