BMAT Development Test

1- SQL (I):

- Tenemos las tablas "department" (id, name) y "employer" (id, name, dep_id).

  Completa la siguiente query que lista los departamentos con más de 10 trabajadores:

  SELECT d.name FROM department d JOIN employer e ON e.dep_id = d.id ...

  Respuestas:

  a) ... GROUP BY d.id HAVING COUNT(e.id) > 10
  b) ... WHERE COUNT(e.id) > 10 AND COUNT(d.id) > 1
  c) ... WHERE SUM(e.id) > 10
  d) Ninguna es correcta
  e) Las tres primeras son correctas

2- XML:

- Dado el siguiente fragmento de XML:

```
<collection>
        <album>
                <track id="123">
                        <title>One</title>
                        <artist>U2</artist>
                </track>
        </album>
</collection>
```

  Qué expresión XPath permite obtener el elemento 'title'?

  Respuestas:

  a) /collection/album/track[@id="123"]/title
  b) //track[@id="123"]/title
  c) /collection/album/track[artist="U2"]/title
  d) Ninguna es correcta
  e) Las tres primeras son correctas

- Razona brevemente la respuesta de la pregunta 2. ¿En qué se diferencian a), b) y c)?

3- SQL (II):

- Tables:

Tracks(id, title, duration, mbid)
{id} is the Primary KEY

Tags(id, name, weight)
{id} is the Primary KEY

Tags_tracks(track_id, tag_id)
{track_id, tag_id} is the Primary KEY
{track_id} is a Foreign Key to Tracks table
{tag_id} is a Foreign Key to Tags table

a) Show those tracks of duration greater than 300 (seconds), where the track title starts with an 'A'.

b) Show all the tags (and its weight) assigned to the track with id 1234, ordered by tag weight in descending order.

c) Show the name of the tracks that had no tag assigned

4- Linux Shell Question:

Count all references to the artist 'Bob Marley' in all text files with tsv extension contained in a directory. You may assume there is only one reference per line.

find_bob.sh /link/to/directory
53

5- Parse xls -> xml

Instructions:
Write a python script parser which reads the attached *metadata.xls* file and produces a file *ingestion.xml* with the following structure:

```
<?xml version="1.0" encoding="UTF-8"?>
<ingestion name="customer name" dateCreated="2011-03-16T11:13:15-04:00">
    <tracks>
        <track id = 'track_id' >
            <title>track title</title>
            <fingerprint_uri>uri_to_fingerprint_file</fingerprint_uri>
            <audio_uri>uri_to_audio_file</audio_uri>
            <video_uri>uri_to_video_file</video_uri>
            <year>2009</year>
            <label>My Label</label>
            <duration>189</duration>
            <number>4</number>
            <ISRC>GBCRW0664902</ISRC>
            <artists>
                <artist>
                    <name>artist name</name>
                </artist>
            </artists>
            <releases>
                <release>
                    <title>release title</title>
                </release>
            </releases>
            <genres>
                <genre>rock</genre>
            </genres>
            <extras>
                <extra name = 'label'>extra value</extra>
            </extras>
        </track>
    </tracks>
</ingestion>
```

Each row in the *metadata.xls* file is one track. The tags which can not be found in the *metadata.xls* should not appear in the *ingestion.xml* file. Columns which are found in the *metadata.xls* but which can not be associated with an xml tag, have to be inserted with an 'extra' tag using the column name for 'value'.

Usage of the python script:

```
$python parse_metadata_xls.py metadata.xls ingestion.xml
```

Help:
You should use the following python modules:
- xlrd (Excel reading)
- lxml (etree class for XML processing)

Any other additional python module is allowed to use.

6- And test xls:

Instructions:
The *ingestion.xml* should only contain the metadata of tracks which have a track title, a main artist, an ISRC and which URL exists in the local file system. The URL is the relative path from the *metadata.xls* files location to the audio file entered in the row of the excel file.

Integrate functionalities into your script to detect the missing values and which indicates which rows are incorrect and what is missing. Incorrect rows should not enter the *ingestion.xml* file. To check if the files exists you can find the list of filenames in the attached *filenames.txt.*

Validate the ISRC – International Standard Recording Code - by implementing a regular expression check using the described format of ISRC in:
http://en.wikipedia.org/wiki/International_Standard_Recording_Code

Help:
You should use the following python modules:
- re (regular expressions - ISRC)