

## ENUNCIADO - EJERCICIO 3 – *bailes*

Dado el documento XML Ejercicio 3 - Query (bailes).xml, realiza las siguientes consultas con XQuery:

1. Mostrar cada uno de los nombres de los bailes con la etiqueta "losbailes".

```
for $i in //baile  
  
return <losbailes>{$i/nombre/text()}</losbailes>
```

2. Mostrar los nombres de los bailes seguidos con el número de plazas entre paréntesis, ambos dentro de la misma etiqueta "losbailes".

```
for $i in //baile  
  
return <losbailes>{$i/nombre/text()} ({ $i/plazas/text()})</losbailes>
```

3. Mostrar los nombres de los bailes cuyo precio sea mayor de 30.

```
for $i in //baile  
  
where $i/precio>30  
  
return <losbailes>{$i/nombre/text()}</losbailes>
```

4. Mostrar los nombres de los bailes cuyo precio sea mayor de 30 y la moneda "euro".

```
for $i in //baile  
  
where $i/precio>30 and $i/precio[@moneda="euro"]  
  
return <losbailes>{$i/nombre/text()}</losbailes>
```

5. Mostrar los nombres y la fecha de comienzo de los bailes que comiencen el mes de enero (utiliza para buscarlo la cadena de texto "/1/").

```
for $i in //baile  
  
where contains($i/comienzo,'/1/')  
  
return <losbailes>{$i/nombre/text()} - { $i/comienzo/text() }</losbailes>
```

6. Mostrar los nombres de los profesores y la sala en la que dan clase, ordénalos por sala.

```
for $i in //baile  
  
order by $i/sala  
  
return <resultado>{$i/profesor/text() } - { $i/sala/text() }</resultado>
```

7. Mostrar los nombres de los profesores eliminando los repetidos y acompañar cada nombre con todas las salas en la que da clase, ordénalos por nombre.

```
for $i in distinct-values(//baile/profesor)  
  
let $sala := //baile[profesor=$i]/sala  
  
order by $i
```

```
return <profesores>

{$i} - {$sala}</profesores>
```

8. Mostrar la media de los precios de todos los bailes.

```
let $i := //baile

return <media>{ avg($i/precio)} €</media>
```

9. Mostrar la suma de los precios de los bailes de la sala 1.

```
let $i := //baile[sala="1"]

return <media>{ sum($i/precio)} €</media>
```

10. Mostrar cuántas plazas en total oferta el profesor "Jesus Lozano".

```
let $i := //baile[profesor="Jesus Lozano"]

return <resultado>{sum($i/plazas)}</resultado>
```

11. Mostrar el dinero que ganaría la profesora "Laura Mendiola" si se completaran todas las plazas de su baile, sabiendo que sólo tiene un baile.

```
let $i := //baile[profesor="Laura Mendiola"]

return <resultado>{$i/precio*$i/plazas} €</resultado>
```

12. Mostrar el dinero que ganaría el profesor "Jesus Lozano" si se completaran todas las plazas de su baile, pero mostrando el beneficio de cada baile por separado.

```
for $i in //baile

where $i/profesor = "Jesus Lozano"

return <resultado>{$i/nombre/text()} - {$i/precio*$i/plazas} €</resultado>
```

13. Mostrar el dinero que ganaría la profesora "Laura" (no conocemos su apellido) si se completaran todas las plazas de su baile.

```
for $i in //baile

where starts-with($i/profesor, "Laura")

return <resultado>{$i/nombre/text()} - {$i/precio*$i/plazas} €</resultado>
```

14. Mostrar el nombre del baile, su precio y el precio con un descuento del 15% para familias numerosas. Ordenar por el nombre del baile.

```
for $i in //baile

order by $i/nombre

return <resultado>{$i/nombre/text()} - {$i/precio*$i/plazas} € -
<descuento>{$i/precio*085}</descuento></resultado>
```

**15.** Mostrar todos los datos de cada baile excepto la fecha de comienzo y de fin.

```
for $i in //baile
return <resultado>{$i/* except ($i/comienzo|$i/fin)}</resultado>
```