

López Calderón Francisco Javier

PRÁCTICA Xquery

Entrega v2

1. Nombres de oficio que empiezan por P

```
file2*
1 for $emp in /EMPLEADOS/EMP_ROW
2 where starts-with($emp/OFICIO, "P")
3 return $emp/OFICIO
```

<OFICIO>PRESIDENTE</OFICIO>

2. Nombres de oficio y los nº empleados de cada oficio

```
file2*
1 for $ofi in distinct-values(/EMPLEADOS/EMP_ROW/OFICIO)
2 let $emp := count( /EMPLEADOS/EMP_ROW[OFICIO=$ofi] )
3 return concat($ofi, " Posee ", $emp, " empleados")
```

EMPLEADO Posee 4 empleados
VENDEDOR Posee 4 empleados
DIRECTOR Posee 3 empleados
ANALISTA Posee 2 empleados
PRESIDENTE Posee 1 empleados

3. El número de empleados que tiene cada departamento y la media de salario redondeada.

La salida se mostrará:

Departamento: 10. Num emles=3. Media salario=2892

```
file2*
1 for $dept in distinct-values(/EMPLEADOS/EMP_ROW/DEPT_NO)
2 for $emp in count( /EMPLEADOS/EMP_ROW[DEPT_NO=$dept] )
3 for $medi in round(avg( /EMPLEADOS/EMP_ROW[DEPT_NO=$dept]/SALARIO ))
4 return concat("Departamento: ", $dept, ". Num emles= ", $emp, ". Media salario=", $medi)
```

Departamento: 20. Num emles= 5. Media salario=2274
Departamento: 30. Num emles= 6. Media salario=1736
Departamento: 10. Num emles= 3. Media salario=2892

4. Modifica la salida anterior para que quede entre etiquetas

```
<depart>
  <cod>20</cod>
  <emples>5</emples>
  <medsal>2274</medsal>
</depart>
```

```
file2*
1 for $dept in distinct-values(/EMPLEADOS/EMP_ROW/DEPT_NO)
2 for $emp in count( /EMPLEADOS/EMP_ROW[DEPT_NO=$dept] )
3 for $medi in round(avg( /EMPLEADOS/EMP_ROW[DEPT_NO=$dept]/SALARIO ))
4 return <depart>{
5   <cod>{$dept}</cod>,
6   <emples>{$emp}</emples>,
7   <medsal>{$medi}</medsal>
8 }</depart>
```

<depart>
 <cod>20</cod>
 <emples>5</emples>
 <medsal>2274</medsal>
</depart>
<depart>
 <cod>30</cod>
 <emples>6</emples>
 <medsal>1736</medsal>
</depart>
<depart>
 <cod>10</cod>
 <emples>3</emples>
 <medsal>2892</medsal>
</depart>

OK 8 : 12

5. Los nombres de los empleados, los que son directores entre etiquetas `<DIRECTOR></DIRECTOR>` y los que no lo son entre etiquetas `<EMPLE></EMPLE>`

```

file2*
for $ofi in distinct-values(/EMPLEADOS/EMP_ROW/OFICIO)
for $emp in /EMPLEADOS/EMP_ROW[OFICIO=$ofi]/APELLIDO
return if ($ofi="DIRECTOR") then
  <DIRECTOR>{data($emp)}</DIRECTOR>
else
  <EMPLE>{data($emp)}</EMPLE>

```

```

<EMPLE>SANCHEZ</EMPLE>
<EMPLE>ALONSO</EMPLE>
<EMPLE>JIMENO</EMPLE>
<EMPLE>MUÑOZ</EMPLE>
<EMPLE>ARROYO</EMPLE>
<EMPLE>SALA</EMPLE>
<EMPLE>MARTIN</EMPLE>
<EMPLE>TOVAR</EMPLE>
<DIRECTOR>JIMENEZ</DIRECTOR>
<DIRECTOR>NEGRO</DIRECTOR>
<DIRECTOR>CEREZO</DIRECTOR>
<EMPLE>GIL</EMPLE>
<EMPLE>FERNANDEZ</EMPLE>
<EMPLE>REY</EMPLE>

```

6. Nombre y el oficio concatenado entre las etiquetas `<APE_OFI> </APE_OFI>`

```

file2*
1 for $ofi in distinct-values(/EMPLEADOS/EMP_ROW/OFICIO)
2 for $emp in /EMPLEADOS/EMP_ROW[OFICIO=$ofi]/APELLIDO
3 return <APE_OFI>{concat( $emp, " ", $ofi )}</APE_OFI>

```

```

<APE_OFI>SANCHEZ EMPLEADO</APE_OFI>
<APE_OFI>ALONSO EMPLEADO</APE_OFI>
<APE_OFI>JIMENO EMPLEADO</APE_OFI>
<APE_OFI>MUÑOZ EMPLEADO</APE_OFI>
<APE_OFI>ARROYO VENDEDOR</APE_OFI>
<APE_OFI>SALA VENDEDOR</APE_OFI>

```

7. Los nombres de empleados de los departamentos de tipo A, cuyo puesto es profesor.

```

file2*
1 for $dep in /universidad/departamento/empleado[puesto="
  Profesor"]
2 for $tipo in /universidad/departamento[empleado=$dep]
3 where $tipo/@tipo="A"
4 return $dep/nombre

```

```

<nombre>Alicia Martín</nombre>
<nombre>Ma Jesús Ramos</nombre>
<nombre>Pedro Paniagua</nombre>

```

8. Nombre de departamento encerrado entre las etiquetas `<tipo=A></tipo=A>` si es de tipo A, y entre las etiquetas `<tipo=B></tipo=B>` , si no lo es.

```

file2*
1 for $dep in /universidad/departamento
2 return if ($dep/@tipo="A") then
3   <tipoA>{data($dep/nombre)}</tipoA>
4   else
5   <tipoB>{data($dep/nombre)}</tipoB>

```

```

<tipoA>Informática</tipoA>
<tipoA>Matemáticas</tipoA>
<tipoB>Análisis</tipoB>

```

9. Nombres de los departamentos y los empleados que tiene cada departamento.

<pre> 1 for \$dep in /universidad/departamento/nombre 2 for \$emp in count(/universidad/departamento[nombre=\$dep]/ empleado) 3 return concat(\$dep," posee ",\$emp," empleados") </pre>	<p>Informática posee 2 empleados Matemáticas posee 4 empleados Análisis posee 2 empleados</p>
--	---

10. Nombres de los departamentos, los empleados que tiene cada departamento y la media del salario entre etiquetas.

<pre> 1 for \$dep in /universidad/departamento/nombre 2 let \$emp := count(/universidad/departamento[nombre=\$dep]/empleado) 3 for \$mon in avg(/universidad/departamento[nombre=\$dep]/empleado/@ salario) 4 return <departamentos>{ 5 \$dep, 6 <Empleados>{\$emp}</Empleados>, 7 <Media>{\$mon}</Media> 8 }</departamentos> </pre>	<pre> <departamentos> <nombre>Informática</nombre> <Empleados>2</Empleados> <Media>2150</Media> </departamentos> <departamentos> <nombre>Matemáticas</nombre> <Empleados>4</Empleados> <Media>2200</Media> </departamentos> <departamentos> <nombre>Análisis</nombre> <Empleados>2</Empleados> <Media>2050</Media> </departamentos> </pre>
--	--

11. Obtener por cada zona el número de productos que tiene.

<pre> 1 for \$zona in distinct-values(/productos/produc/cod_zona) 2 for \$pro in count(/productos/produc[cod_zona=\$zona]) 3 return concat("La zona ",\$zona," posee ",\$pro," productos") </pre>	<p>La zona 10 posee 4 productos La zona 20 posee 3 productos La zona 30 posee 4 productos La zona 40 posee 2 productos</p>
---	---

12. Obtener la denominación de los productos entre las etiquetas

<zona10></zona10> si son del código de zona 10, <zona20></zona20> si son de la zona 20, <zona30></zona30> si son de la 30 y <zona40></zona40> si son de la 40.

<pre> 1 for \$zona in distinct-values(/productos/produc/cod_zona) 2 for \$pro in /productos/produc[cod_zona=\$zona]/denominacion 3 order by \$zona 4 return if (\$zona=10) then 5 <Zona10>{data(\$pro)}</Zona10> 6 else if (\$zona=20) then 7 <Zona20>{data(\$pro)}</Zona20> 8 else if (\$zona=30) then 9 <Zona30>{data(\$pro)}</Zona30> 10 else 11 <Zona40>{data(\$pro)}</Zona40> </pre>	<pre> <Zona10>Placa Base MSI G41M-P26</Zona10> <Zona10>Micro Intel Core i5-2320</Zona10> <Zona10>Memoria DDR3 G.Skill 2GB</Zona10> <Zona10>Memoria DDR3 G.Skill 4GB</Zona10> <Zona20>Micro Intel Core i5 2500</Zona20> <Zona20>HD Seagate Barracuda 250GB SATA</Zona20> <Zona20>Memoria DDR3 Kingston HyperX 4GB</Zona20> <Zona30>HD Caviar Blue 500GB SATA3</Zona30> <Zona30>Tarjeta gráfica Asus GeForce EN210 Silent 1GB</Zona30> </pre>
---	---

13. Obtener por cada zona la denominación del o de los productos más caros.

```

1 for $zona in distinct-values(/productos/produccion/cod_zona)
2 let $pre := max(/productos/produccion[cod_zona=$zona]/precio)
3 let $pro := /productos/produccion[precio=$pre]/denominacion
4 return concat("La zona ", $zona, " tiene el producto más caro: ", $pre, " ",
  data($pro) )

```

La zona 10 tiene el producto más caro: 120 Micro Intel Core i5-2320
 La zona 20 tiene el producto más caro: 170 Micro Intel Core i5 2500
 La zona 30 tiene el producto más caro: 150 HD Caviar Blue 500GB SATA3
 La zona 40 tiene el producto más caro: 60 Micro Intel Dual Core G620

14. Obtener la denominación de los productos contenida entre las etiquetas <placa></placa> para los productos en cuya denominación aparece la palabra Placa Base, <memoria></memoria>, para los que contienen a la palabra Memoria <micro></micro>, para los que contienen la palabra Micro y <otros></otros> para el resto de productos.

```

1 for $pro in /productos/produccion
2 return if ($pro[contains(denominacion,"Placa")]) then
3   <placa>{data($pro)}</placa>
4   else if ($pro[contains(denominacion,"Memoria")]) then
5     <memoria>{data($pro)}</memoria>
6   else if ($pro[contains(denominacion,"Micro")]) then
7     <micro>{data($pro)}</micro>
8   else
9     <otros>{data($pro)}</otros>

```

<placa>1010Placa Base MSI G41M-P265010310</placa>
 <micro>1011Micro Intel Core i5-23201203510</micro>
 <micro>1012Micro Intel Core i5 25001705620</micro>
 <otros>1013HD Seagate Barracuda 250GB SATA8010520</otros>
 <otros>1014HD Caviar Blue 500GB SATA31505630</otros>
 <otros>1015Tarjeta gráfica Asus GeForce EN210 Silent 1GB4010530</otros>
 <otros>1016Tarjeta gráfica Gigabyte GeForce 1GB505640</otros>
 <otros>1017Tarjeta gráfica Nvidia Express 1GB4510530</otros>

15. A partir del fichero xml, genera mediante una consulta XQuery un HTML con los datos de los productos en forma de tabla y dale formato con un css externo.

```

1 <html>
2 <head>
3   <title>Tabla Ejercicio 15</title>
4   <link rel="stylesheet" href="ejercicio15.css"/>
5 </head>
6 <body>
7   <table>
8   {
9     for $prueba in doc("xml/productos.xml")//produccion
10    return <tr>
11      <td>{$prueba/cod_zona}</td>
12      <td>{$prueba/stock_minimo}</td>
13      <td>{$prueba/stock_actual}</td>
14      <td>{$prueba/precio}</td>
15      <td>{$prueba/denominacion}</td>
16      <td>{$prueba/cod_prod}</td>
17    }
18  }
19 </table>
20 </body>
21 </html>

```

<cod_prod>1021</cod_prod>
 </td>
 </tr>
 <tr>
 <td>
 <cod_zona>30</cod_zona>
 </td>
 <td>
 <stock_minimo>2</stock_minimo>
 </td>
 <td>
 <stock_actual>2</stock_actual>
 </td>
 <td>
 <precio>52</precio>
 </td>
 <td>
 <denominacion>Placa Base ASRock G41M-S3</denominacion>
 </td>
 <td>
 <cod_prod>1022</cod_prod>
 ...

Página Subida

103	10	50	Placa Base MSI G41M-P26	1010
105	3	120	Micro Intel Core i5-2320	1011
206	5	170	Micro Intel Core i5 2500	1012
205	10	80	HD Seagate Barracuda 250GB SATA	1013
306	5	150	HD Caviar Blue 500GB SATA3	1014
305	10	40	Tarjeta gráfica Asus GeForce EN210 Silent 1GB	1015
406	5	50	Tarjeta gráfica Gigabyte GeForce 1GB	1016
305	10	45	Tarjeta gráfica Nvidia Express 1GB	1017
405	15	60	Micro Intel Dual Core G620	1018
103	5	10	Memoria DDR3 G.Skill 2GB	1019
1010	30	30	Memoria DDR3 G.Skill 4GB	1020
204	15	16	Memoria DDR3 Kingston HyperX 4GB	1021
302	2	52	Placa Base ASRock G41M-S3	1022