

# Practica 6: Teórico-práctica

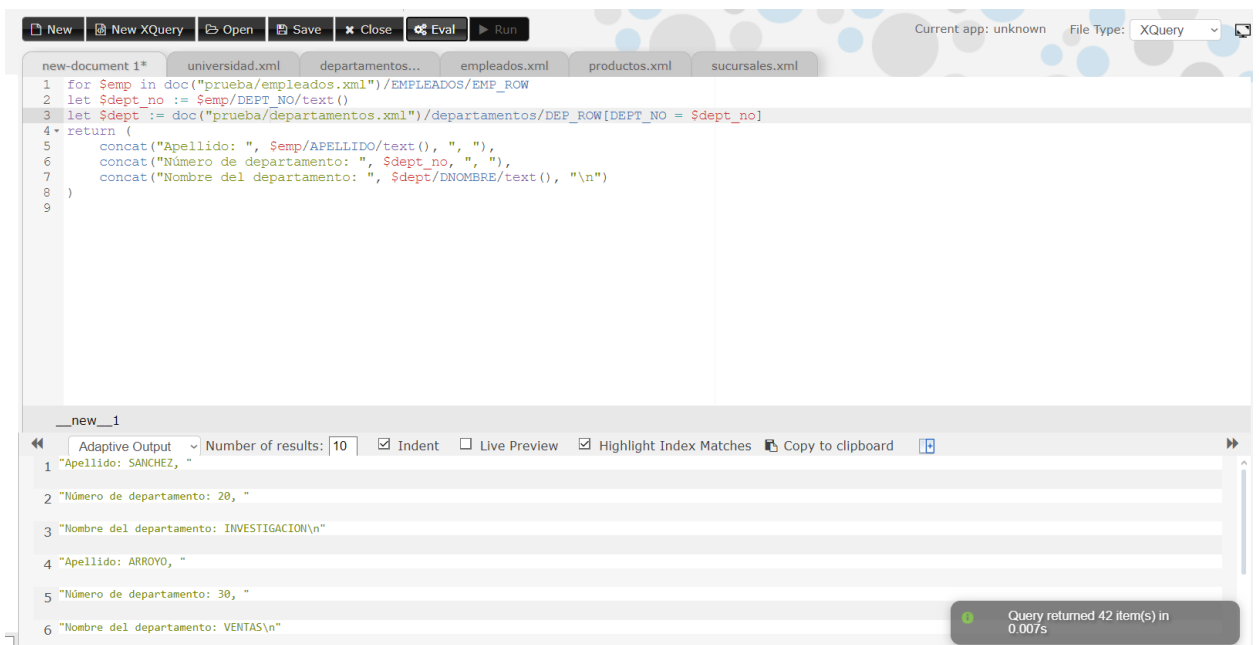
Aprendiendo consultas complejas, joins y varios for.

Resuelve las consultas dadas en teoría:

## Ejemplos de Joins de documentos.

a) Visualizar por cada empleado del documento **empleados.xml**, su apellido, su número de departamento y nombre del departamento que se encuentra en el documento **departamentos.xml**.

```
for $emp in doc("prueba/empleados.xml")/EMPLEADOS/EMP_ROW
let $dept_no := $emp/DEPT_NO/text()
let                                     $dept                                     :=
doc("prueba/departamentos.xml")/departamentos/DEP_ROW[DEPT_NO
= $dept_no]
return (
    concat("Apellido: ", $emp/APELLIDO/text(), ", ", " "),
    concat("Número de departamento: ", $dept_no, ", ", " "),
    concat("Nombre del departamento: ", $dept/DNOMBRE/text(), "\n")
)
```

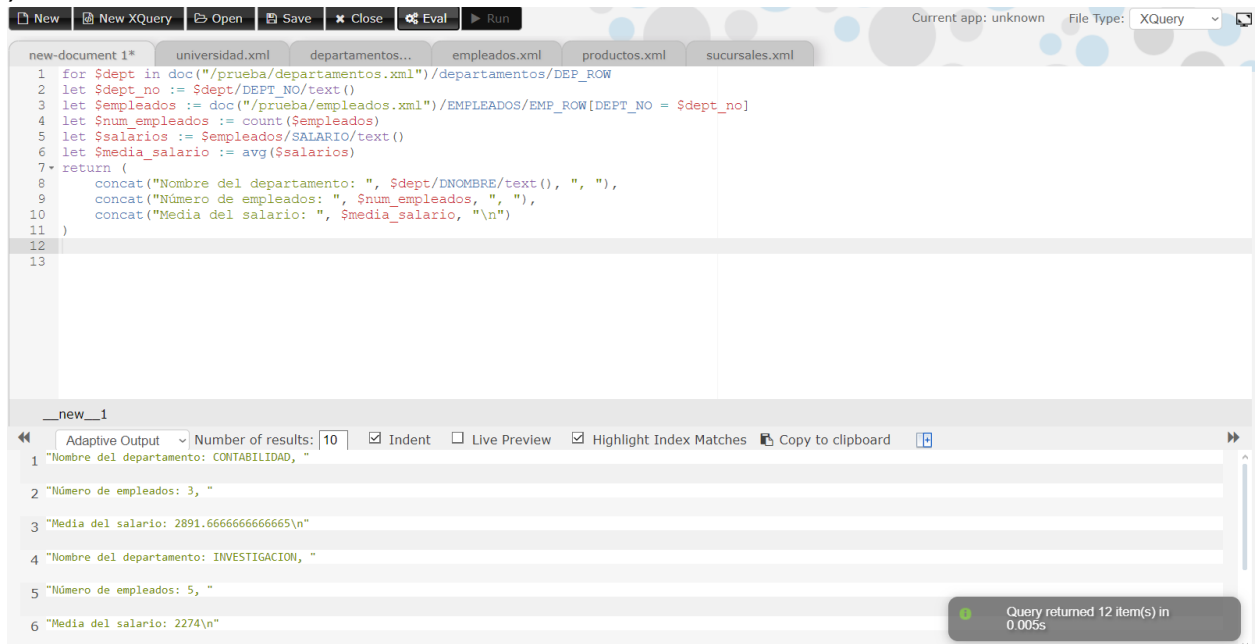


b) Obtener por cada departamento, el nombre del departamento, el número de empleados y la media del salario.

```

for $dept in
doc("/prueba/departamentos.xml")/departamentos/DEP_ROW
let $dept_no := $dept/DEPT_NO/text()
let $empleados :=
doc("/prueba/empleados.xml")/EMPLEADOS/EMP_ROW[DEPT_NO =
$dept_no]
let $num_empleados := count($empleados)
let $salarios := $empleados/SALARIO/text()
let $media_salario := avg($salarios)
return (
  concat("Nombre del departamento: ", $dept/DNOMBRE/text(), ", ", " "),
  concat("Número de empleados: ", $num_empleados, ", ", " "),
  concat("Media del salario: ", $media_salario, "\n")
)

```



c) Obtener por cada departamento el nombre del empleado que más gana.

```

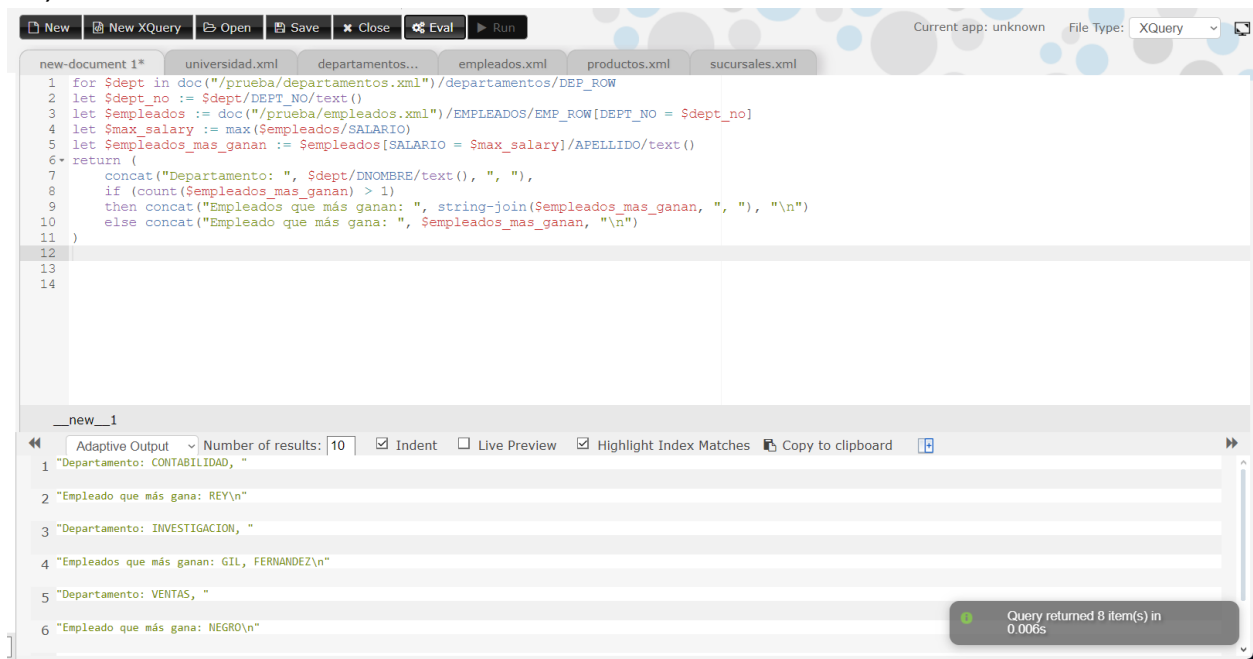
for $dept in
doc("/prueba/departamentos.xml")/departamentos/DEP_ROW
let $dept_no := $dept/DEPT_NO/text()
let $empleados :=
doc("/prueba/empleados.xml")/EMPLEADOS/EMP_ROW[DEPT_NO =
$dept_no]
let $max_salary := max($empleados/SALARIO)

```

```

let $empleados_mas_ganan := $empleados[SALARIO =
$max_salary]/APELLIDO/text()
return (
  concat("Departamento: ", $dept/DNOMBRE/text(), ", "),
  if (count($empleados_mas_ganan) > 1)
  then concat("Empleados que más ganan: ", string-
join($empleados_mas_ganan, ", "), "\n")
  else concat("Empleado que más gana: ", $empleados_mas_ganan,
"\n")
)

```



## Ejemplos de uso de varios for:

La utilización de varios for es muy útil para consultas en documentos XML anidados y también cuando utilizamos varios documentos unidos por una clausula where como una combinación de tablas en SQL.



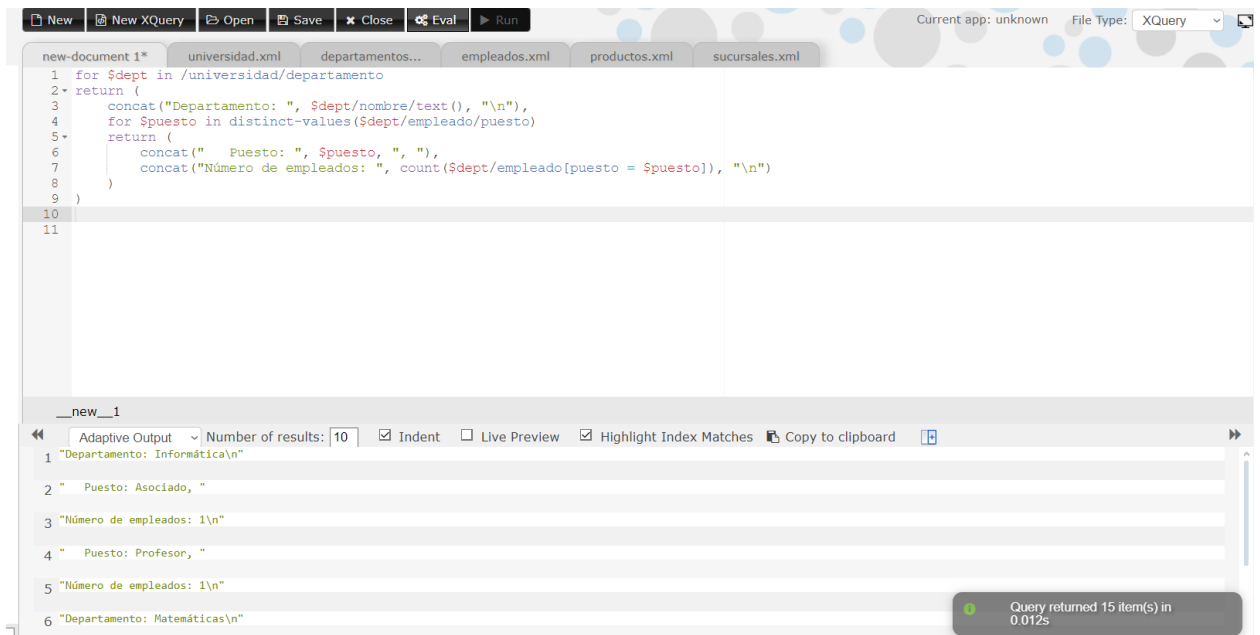
Árbol del documento *universidad.xml*.

a) Visualizar por cada departamento del documento *universidad.xml*, **el número de empleados que hay en cada puesto de trabajo**. Se usa un `for` para obtener los nodos `departamento` y un segundo `for` para obtener los distintos puestos de cada departamento.

```

for $dept in /universidad/departamento
return (
  concat("Departamento: ", $dept/nombre/text(), "\n"),
  for $puesto in distinct-values($dept/empleado/puesto)
  return (
    concat(" Puesto: ", $puesto, ", "),
    concat("Número de empleados: ",
count($dept/empleado[puesto = $puesto]), "\n")
  )
)

```



b) Visualizar por cada departamento del documento universidad.xml, **el salario máximo y el empleado que tiene ese salario**. El primer for obtiene los nodos departamento y el segundo for los empleados de cada departamento. Para sacar el máximo, en la salida preguntamos si el salario es el máximo.

**for \$dept in /universidad/departamento**

**return (**

**concat("Departamento: ", \$dept/nombre/text(), "\n"),**

**let \$max\_salary := max(\$dept/empleado/@salario)**

**return (**

**concat("Salario máximo: ", \$max\_salary, ", "),**

**for \$emp in \$dept/empleado**

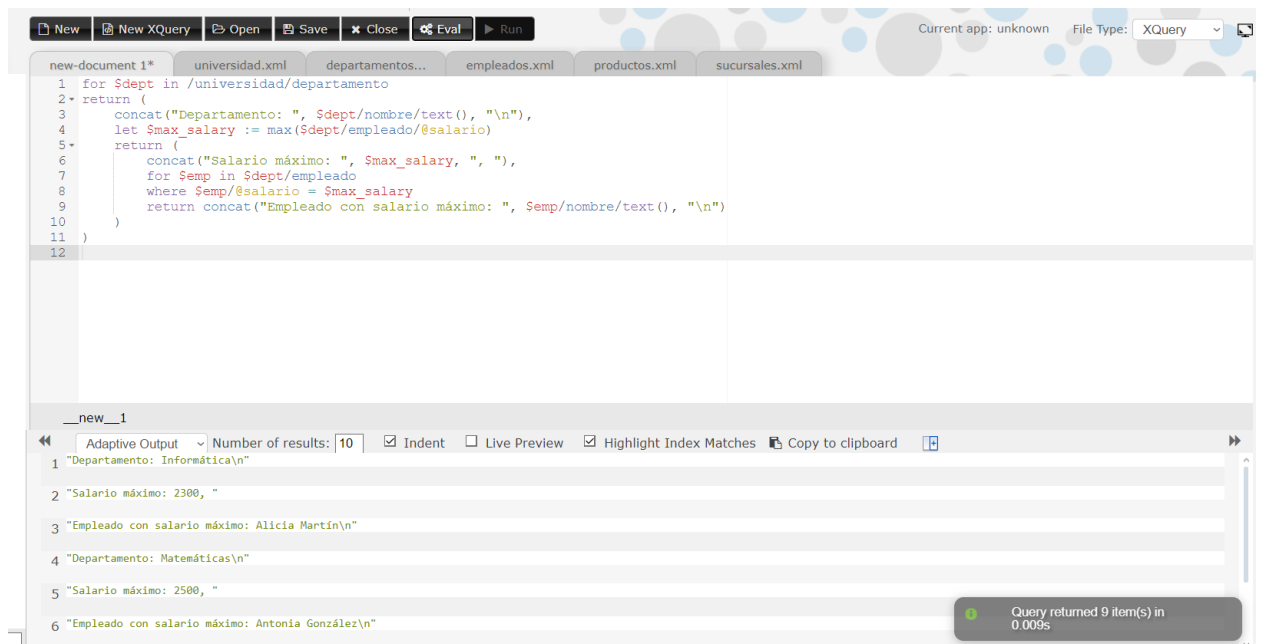
**where \$emp/@salario = \$max\_salary**

**return concat("Empleado con salario máximo: ", \$emp/nombre/text(),**

**"\n")**

**)**

**)**



- c) Visualiza por cada puesto del documento universidad.xml, el **empleado con salario máximo y ese salario**. El primer for obtiene los distintos puestos de trabajo y el segundo for obtiene los empleados que tienen ese puesto de trabajo. En el if se pregunta si el salario del empleado es el salario máximo de los empleados del oficio del primer for.

```

for          $puesto          in          distinct-
values(/universidad/departamento/empleado/puesto)
return (
  concat("Puesto: ", $puesto, "\n"),
  let          $max_salary          :=
max(/universidad/departamento/empleado[puesto
$puesto]/@salario)
  for $semp in /universidad/departamento/empleado[puesto =
$semp]
  where $semp/@salario = $max_salary
  return (
    concat("Empleado          con          salario          máximo:          ",
$semp/nombre/text(), ", "),
    concat("Salario máximo: ", $max_salary, "\n")
  )
)

```

The screenshot shows an XQuery editor with a menu bar (New, New XQuery, Open, Save, Close, Eval, Run) and a toolbar. The editor has tabs for 'new-document 1\*', 'universidad.xml', 'departamentos...', 'empleados.xml', 'productos.xml', and 'sucursales.xml'. The main text area contains the following XQuery code:

```

1 for $puesto in distinct-values(/universidad/departamento/empleado/puesto)
2- return (
3   concat("Puesto: ", $puesto, "\n"),
4   let $max_salary := max(/universidad/departamento/empleado[puesto = $puesto]/@salario)
5   for $emp in /universidad/departamento/empleado[puesto = $puesto]
6   where $emp/@salario = $max_salary
7- return (
8   concat("Empleado con salario máximo: ", $emp/nombre/text(), ", "),
9   concat("Salario máximo: ", $max_salary, "\n")
10 )
11 )
12

```

Below the editor, the results pane shows the output of the query. It includes a toolbar with 'Adaptive Output', 'Number of results: 10', 'Indent', 'Live Preview', 'Highlight Index Matches', and 'Copy to clipboard'. The results are as follows:

```

1 "Puesto: Asociado\n"
2 "Empleado con salario máximo: Mario García, "
3 "Salario máximo: 2200\n"
4 "Puesto: Profesor\n"
5 "Empleado con salario máximo: Alicia Martín, "
6 "Salario máximo: 2300\n"

```

A status bar at the bottom right indicates: "Query returned 14 item(s) in 0.013s".

La consulta a) del apartado anterior, joins de documentos, también se podría haber resuelto con dos for y where. ¿Cómo lo harías?

```

for
    $dept in doc("/prueba/departamentos.xml")/departamentos/DEP_ROW
return (
    concat("Nombre del departamento: ", $dept/DNOMBRE/text(), "\n"),
    for $emp in doc("/prueba/empleados.xml")/EMPLEADOS/EMP_ROW
    where $emp/DEPT_NO = $dept/DEPT_NO
    return (
        concat(" Apellido: ", $emp/APELLIDO/text(), ", "),
        concat("Número de departamento: ", $dept/DEPT_NO/text(), "\n")
    )
)

```

NewNew XQueryOpenSaveCloseEvalRun

Current app: unknownFile Type: XQuery

new-document 1\*universidad.xmldepartamentos...empleados.xmlproductos.xmlsucursales.xml

```
1 for $dept in doc("/prueba/departamentos.xml")/departamentos/DEPT_ROW
2 return (
3   concat("Nombre del departamento: ", $dept/DNOMBRE/text(), "\n"),
4   for $emp in doc("/prueba/empleados.xml")/EMPLEADOS/EMP_ROW
5   where $emp/DEPT_NO = $dept/DEPT_NO
6   return (
7     concat("  Apellido: ", $emp/APELLIDO/text(), ", "),
8     concat("Número de departamento: ", $dept/DEPT_NO/text(), "\n")
9   )
10 )
11
```

\_\_new\_\_1

Adaptive OutputNumber of results: 10IndentLive PreviewHighlight Index MatchesCopy to clipboard

```
1 "Nombre del departamento: CONTABILIDAD\n"
2 "  Apellido: CEREZO, "
3 "Número de departamento: 10\n"
4 "  Apellido: REY, "
5 "Número de departamento: 10\n"
6 "  Apellido: MUÑOZ, "
```

Query returned 32 item(s) in 0.004s