

Tenemos el documento XML sobre Sucursales cuya estructura es (se adjunta el primer dato, el resto se pasa el documento completo sucursales.xml)

```
<sucursales>
  <sucursal telefono="123456789" codigo="SUC1">
    <director>Juan Perez</director>
    <poblacion>Madrid</poblacion>
    <cuenta tipo="AHORRO">
      <nombre>Cuenta Ahorro 1</nombre>
      <numero>0001</numero>
      <saldohaber>5000</saldohaber>
      <saldodebe>1000</saldodebe>
    </cuenta>
    <cuenta tipo="CORRIENTE">
      <nombre>Cuenta Corriente 1</nombre>
      <numero>0002</numero>
      <saldohaber>15000</saldohaber>
      <saldodebe>3000</saldodebe>
    </cuenta>
    <cuenta tipo="PENSIONES">
      <nombre>Cuenta Pensiones 1</nombre>
      <numero>0003</numero>
      <saldohaber>8000</saldohaber>
      <saldodebe>2000</saldodebe>
    </cuenta>
  </sucursal>
</sucursales>
```

1. Obtener los datos de las cuentas bancarias cuyo tipo sea AHORRO.

```
<cuenta tipo="AHORRO">
  <nombre>Cuenta Ahorro 1</nombre>
  <numero>0001</numero>
  <saldohaber>5000</saldohaber>
  <saldodebe>1000</saldodebe>
</cuenta>
<cuenta tipo="AHORRO">
  <nombre>Cuenta Ahorro 2</nombre>
  <numero>0004</numero>
  <saldohaber>12000</saldohaber>
  <saldodebe>4000</saldodebe>
</cuenta>
<cuenta tipo="AHORRO">
  <nombre>Cuenta Ahorro 3</nombre>
  <numero>0005</numero>
  <saldohaber>3000</saldohaber>
  <saldodebe>500</saldodebe>
</cuenta>
<cuenta tipo="AHORRO">
  <nombre>Cuenta Ahorro 4</nombre>
  <numero>0010</numero>
  <saldohaber>6000</saldohaber>
  <saldodebe>800</saldodebe>
</cuenta>
<cuenta tipo="AHORRO">
  <nombre>Cuenta Ahorro 5</nombre>
  <numero>0011</numero>
  <saldohaber>8000</saldohaber>
  <saldodebe>1000</saldodebe>
</cuenta>
<cuenta tipo="AHORRO">
  <nombre>Cuenta Ahorro 6</nombre>
  <numero>0014</numero>
  <saldohaber>10000</saldohaber>
  <saldodebe>2000</saldodebe>
</cuenta>
<cuenta tipo="AHORRO">
  <nombre>Cuenta Ahorro 7</nombre>
  <numero>0015</numero>
  <saldohaber>4000</saldohaber>
  <saldodebe>1000</saldodebe>
</cuenta>
<cuenta tipo="AHORRO">
  <nombre>Cuenta Ahorro 8</nombre>
  <numero>0020</numero>
  <saldohaber>2500</saldohaber>
```

/sucursales/sucursal/cuenta[@tipo = "AHORRO"]

2. Obtener por cada sucursal la concatenación de su código, y el número de cuentas del tipo AHORRO que tiene

```
Result
SUC1 1
SUC2 2
SUC3 1
SUC4 1
SUC5 2
SUC6 1

for $sucursal in /sucursales/sucursal return
concat($sucursal/@codigo," ",count(/$sucursal/cuenta[@tipo =
"AHORRO"]))
```

3. Obtener las cuentas de tipo PENSIONES de la sucursal con código SUC3.

```
/sucursales/sucursal[@codigo = "SUC3"]/cuenta[@tipo =
"PENSIONES"]
```




```
Result
<cuanta tipo="PENSIONES">
  <nombre>Cuenta Pensiones 3</nombre>
  <numero>0008</numero>
  <saldohaber>11000</saldohaber>
  <saldodebe>3000</saldodebe>
</cuanta>
```

4. Obtener por cada sucursal la concatenación de los datos, código sucursal, director, y total saldo haber

```
Result
SUC1 Juan Perez 28000
SUC2 Ana Gomez 38000
SUC3 Maria Lopez 19000
SUC4 Pedro Sanchez 25000
SUC5 Lucia Ramirez 36000
SUC6 Carlos Ruiz 36500

for $sucursal in /sucursales/sucursal
let $totalSaldoHaber := sum($sucursal/cuenta/saldohaber)
return
concat($sucursal/@codigo, ' - ', $sucursal/director, ' - ',
$totalSaldoHaber)
```

5. Obtener todos los elementos de las sucursales con más de 3 cuentas.



Result

```
<sucursal telefono="987654321" codigo="SUC2">
  <director>Ana Gomez</director>
  <poblacion>Barcelona</poblacion>
  <cuenta tipo="AHORRO">
    <nombre>Cuenta Ahorro 2</nombre>
    <numero>0004</numero>
    <saldohaber>12000</saldohaber>
    <saldodebe>4000</saldodebe>
  </cuenta>
  <cuenta tipo="AHORRO">
    <nombre>Cuenta Ahorro 3</nombre>
    <numero>0005</numero>
    <saldohaber>3000</saldohaber>
    <saldodebe>500</saldodebe>
  </cuenta>
  <cuenta tipo="PENSIONES">
    <nombre>Cuenta Pensiones 2</nombre>
    <numero>0006</numero>
    <saldohaber>7000</saldohaber>
    <saldodebe>1500</saldodebe>
  </cuenta>
  <cuenta tipo="CORRIENTE">
    <nombre>Cuenta Corriente 2</nombre>
    <numero>0007</numero>
    <saldohaber>16000</saldohaber>
    <saldodebe>2500</saldodebe>
  </cuenta>
</sucursal>
<sucursal telefono="444555666" codigo="SUC5">
  <director>Lucia Ramirez</director>
  <poblacion>Madrid</poblacion>
  <cuenta tipo="AHORRO">
    <nombre>Cuenta Ahorro 6</nombre>
    <numero>0014</numero>
    <saldohaber>10000</saldohaber>
    <saldodebe>2000</saldodebe>
  </cuenta>
  <cuenta tipo="AHORRO">
    <nombre>Cuenta Ahorro 7</nombre>
    <numero>0015</numero>
    <saldohaber>4000</saldohaber>
    <saldodebe>1000</saldodebe>
  </cuenta>
  <cuenta tipo="PENSIONES">
    <nombre>Cuenta Pensiones 5</nombre>
    <numero>0016</numero>
    <saldohaber>0000</saldohaber>
    <saldodebe>0000</saldodebe>
  </cuenta>
</sucursal>
```

for \$sucursal in /sucursales/sucursal
where count(\$sucursal/cuenta) > 3
return \$sucursal




6. Obtener todos los elementos de las sucursales con más de 3 cuentas del tipo AHORRO.

```
<sucursal telefono="987654321" codigo="SUC2">
  <director>Ana Gomez</director>
  <poblacion>Barcelona</poblacion>
  <cuenta tipo="AHORRO">
    <nombre>Cuenta Ahorro 2</nombre>
    <numero>0004</numero>
    <saldohaber>12000</saldohaber>
    <saldodebe>4000</saldodebe>
  </cuenta>
  <cuenta tipo="AHORRO">
    <nombre>Cuenta Ahorro 3</nombre>
    <numero>0005</numero>
    <saldohaber>3000</saldohaber>
    <saldodebe>500</saldodebe>
  </cuenta>
  <cuenta tipo="PENSIONES">
    <nombre>Cuenta Pensiones 2</nombre>
    <numero>0006</numero>
    <saldohaber>7000</saldohaber>
    <saldodebe>1500</saldodebe>
  </cuenta>
  <cuenta tipo="CORRIENTE">
    <nombre>Cuenta Corriente 2</nombre>
    <numero>0007</numero>
    <saldohaber>16000</saldohaber>
    <saldodebe>2500</saldodebe>
  </cuenta>
</sucursal>
<sucursal telefono="444555666" codigo="SUC5">
  <director>Lucia Ramirez</director>
  <poblacion>Madrid</poblacion>
  <cuenta tipo="AHORRO">
    <nombre>Cuenta Ahorro 6</nombre>
    <numero>0014</numero>
    <saldohaber>10000</saldohaber>
    <saldodebe>2000</saldodebe>
  </cuenta>
  <cuenta tipo="AHORRO">
    <nombre>Cuenta Ahorro 7</nombre>
    <numero>0015</numero>
    <saldohaber>4000</saldohaber>
    <saldodebe>1000</saldodebe>
  </cuenta>
  <cuenta tipo="PENSIONES">
    <nombre>Cuenta Pensiones 5</nombre>
    <numero>0016</numero>
```

```
for $sucursal in /sucursales/sucursal
where count($sucursal/cuenta[@tipo =
"AHORRO"]) > 3
return $sucursal
```

7. Obtener los nodos del director y la población de las sucursales con más de 3 cuentas.




```
for $sucursal in /sucursales/sucursal
where count($sucursal/cuenta) > 3
return $sucursal/(director|poblacion)
```



Result

```
<director>Ana Gomez</director>
<director>Lucia Ramirez</director>
```

8. Obtener el número de sucursales cuya población sea Madrid.






Result

2

```
count(sucursales/sucursal[poblacion = "Madrid"])
```

9. Obtener por cada sucursal, su código y la suma de los saldos debes de las cuentas del tipo PENSIONES.






Result

```
SUC1 2000
SUC2 1500
SUC3 3000
SUC4 500
SUC5 1500
SUC6 3000
```

```
for $sucursal in /sucursales/sucursal
let $totalSaldoDebe := sum($sucursal/cuenta[@tipo =
"PENSIONES"]/saldodebe)
return
concat($sucursal/@codigo, ' - ', $totalSaldoDebe)
```

10. Obtener los nodos número de cuenta, nombre de cuenta y el saldo haber de las cuentas con saldo haber mayor de 10000.



Result

```
SUC1 Juan Perez
SUC2 Ana Gomez
SUC3 Maria Lopez
SUC4 Pedro Sanchez
SUC5 Lucia Ramirez
SUC6 Carlos Ruiz
```

```
sucursales/sucursal/cuenta[saldohaber>10000]/(nombre|numero|saldo
haber)
```

11. Obtener las cuentas con saldo haber mayor que 5000 y saldo debe menor que 2000 en todas las sucursales.



Result

```
<cuota tipo="PENSIONES">
  <nombre>Cuenta Pensiones 2</nombre>
  <numero>0006</numero>
  <saldohaber>7000</saldohaber>
  <saldodebe>1500</saldodebe>
</cuota>
<cuota tipo="AHORRO">
  <nombre>Cuenta Ahorro 4</nombre>
  <numero>0010</numero>
  <saldohaber>6000</saldohaber>
  <saldodebe>800</saldodebe>
</cuota>
<cuota tipo="AHORRO">
  <nombre>Cuenta Ahorro 5</nombre>
  <numero>0011</numero>
  <saldohaber>8000</saldohaber>
  <saldodebe>1000</saldodebe>
</cuota>
<cuota tipo="PENSIONES">
  <nombre>Cuenta Pensiones 5</nombre>
  <numero>0016</numero>
  <saldohaber>9000</saldohaber>
  <saldodebe>1500</saldodebe>
</cuota>
```

sucursales/sucursal/cuenta[saldohaber>5000 and
saldodebe < 2000]

12. Obtener el código y el nombre del director de las sucursales con al menos una cuenta del tipo `AHORRO` y una del tipo `PENSIONES`.



Result

```
SUC1 Juan Perez
SUC2 Ana Gomez
SUC3 Maria Lopez
SUC4 Pedro Sanchez
SUC5 Lucia Ramirez
SUC6 Carlos Ruiz
```

```
for $sucursal in /sucursales/sucursal
where $sucursal/cuenta[@tipo = "AHORRO"] and
$sucursal/cuenta[@tipo = "PENSIONES"]
return
concat($sucursal/@codigo, ' - ', $sucursal/director)
```

13. Obtener el código y el total de cuentas por tipo en cada sucursal (concatenando tipo y cantidad).



Result

```
SUC1 AHORRO:1 PENSIONES:1 CORRIENTE:1
SUC2 AHORRO:2 PENSIONES:1 CORRIENTE:1
SUC3 AHORRO:1 PENSIONES:1 CORRIENTE:1
SUC4 AHORRO:1 PENSIONES:1 CORRIENTE:1
SUC5 AHORRO:2 PENSIONES:1 CORRIENTE:1
SUC6 AHORRO:1 PENSIONES:1 CORRIENTE:1
```

```
/sucursales/sucursal/concat(@codigo, "
AHORRO: ",
count(cuenta[@tipo="AHORRO"]), "
PENSIONES: ",
count(cuenta[@tipo="PENSIONES"]), "
CORRIENTE: ",
count(cuenta[@tipo="CORRIENTE"]))
```

14. Obtener la cuenta con el mayor saldo haber en cada sucursal.



Result

```
<cuanta tipo="CORRIENTE">
  <nombre>Cuenta Corriente 6</nombre>
  <numero>0019</numero>
  <saldohaber>20000</saldohaber>
  <saldodebe>4000</saldodebe>
</cuanta>
```

15. Obtener el código de las sucursales donde el saldo total de las cuentas de tipo `CORRIENTE` es mayor que el saldo total de las cuentas de tipo `AHORRO`.



Result

```
codigo="SUC1"
codigo="SUC2"
codigo="SUC4"
codigo="SUC6"
```

16. Obtener el número de cuenta y saldo debe de todas las cuentas de tipo `PENSIONES` con saldo debe mayor al promedio del saldo debe de todas las cuentas de ese tipo.



Result

```
<numero>0003</numero>
<saldodebe>2000</saldodebe>
<numero>0008</numero>
<saldodebe>3000</saldodebe>
<numero>0018</numero>
<saldodebe>3000</saldodebe>
```

17. Obtener los nombres y números de las cuentas cuya sucursal tiene más de una cuenta con saldo haber superior a 10000.



Result

```
<nombre>Cuenta Ahorro 2</nombre>
<numero>0004</numero>
<nombre>Cuenta Ahorro 3</nombre>
<numero>0005</numero>
<nombre>Cuenta Pensiones 2</nombre>
<numero>0006</numero>
<nombre>Cuenta Corriente 2</nombre>
<numero>0007</numero>
<nombre>Cuenta Pensiones 6</nombre>
<numero>0018</numero>
<nombre>Cuenta Corriente 6</nombre>
<numero>0019</numero>
<nombre>Cuenta Ahorro 8</nombre>
<numero>0020</numero>
```

18. Obtener el código de cada sucursal, la cantidad total de cuentas y el promedio del saldo debe en todas las cuentas de la sucursal.



Result

```
SUC1 3 2000
SUC2 4 2125
SUC3 3 1433.3333333333333
SUC4 3 1333.3333333333333
SUC5 4 1750
SUC6 3 2533.3333333333335
```

19. Para cada sucursal, obtener el código de la sucursal junto con la suma de los saldos haber de todas sus cuentas.



Result

```
SUC1 28000
SUC2 38000
SUC3 19000
SUC4 25000
SUC5 36000
SUC6 36500
```

20. Listar el nombre de cada director y el número de cuentas de tipo `CORRIENTE` en su sucursal, solo si la sucursal tiene al menos una cuenta `CORRIENTE`.



Result

```
Juan Perez - Número de cuentas Corriente: 1
Ana Gomez - Número de cuentas Corriente: 1
Maria Lopez - Número de cuentas Corriente: 1
Pedro Sanchez - Número de cuentas Corriente: 1
Lucia Ramirez - Número de cuentas Corriente: 1
Carlos Ruiz - Número de cuentas Corriente: 1
```




21. Para cada sucursal, obtener el código y el saldo promedio de las cuentas `AHORRO`, si existen cuentas de este tipo.






Result

```
SUC1 - Saldo promedio AHORRO: 5000
SUC2 - Saldo promedio AHORRO: 7500
SUC3 - Saldo promedio AHORRO: 6000
SUC4 - Saldo promedio AHORRO: 8000
SUC5 - Saldo promedio AHORRO: 7000
SUC6 - Saldo promedio AHORRO: 2500
```


22. Obtener el código de cada sucursal junto con la diferencia entre el saldo total de `saldohaber` y `saldodebe` para todas sus cuentas.

  	Result
SUC1 - Diferencia Haber-Debe: 22000	
SUC2 - Diferencia Haber-Debe: 29500	
SUC3 - Diferencia Haber-Debe: 14700	
SUC4 - Diferencia Haber-Debe: 21000	
SUC5 - Diferencia Haber-Debe: 29000	
SUC6 - Diferencia Haber-Debe: 28900	

23. Para cada sucursal, mostrar el código y el número de cuentas cuyo saldo debe sea mayor al saldo promedio de debe de todas las cuentas de la sucursal.

  	Result
SUC1 - Cuentas con debe mayor al promedio: 1	
SUC2 - Cuentas con debe mayor al promedio: 2	
SUC3 - Cuentas con debe mayor al promedio: 1	
SUC4 - Cuentas con debe mayor al promedio: 1	
SUC5 - Cuentas con debe mayor al promedio: 2	
SUC6 - Cuentas con debe mayor al promedio: 2	