**EJERCICIO 2**

En la base de datos eXist DB, crea una colección denominada *ejercicio2* y sube el siguiente documento XML: *sucursales.xml*.

Realiza las siguientes consultas con XPath:

 Obtener los datos de las cuentas bancarias cuyo tipo sea ahorro.

//cuenta[@tipo="AHORRO"]

 Obtener por cada sucursal la concatenación de su código y el número de cuentas del tipo ahorro que tiene.

//sucursal/concat(@codigo," | ", count(cuenta[@tipo="AHORRO"]))

 Obtener las cuentas de tipo pensiones de la sucursal con código SUC3.

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

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

//sucursal/concat(@codigo,"|",director/text(),"|",sum(cuenta/saldohaber))

 Obtener todos los elementos de las sucursales con más de tres cuentas.

//sucursal[count(cuenta)>3]

 Obtén en todos los elementos de las sucursales, más de tres cuentas del tipo AHORRO.

//sucursal[count(cuenta[@tipo="AHORRO"])>3]

 Obtener los datos del director y la población de las sucursales con más de tres cuentas.

//sucursal[count(cuenta)>3]/concat(director/text(),"|",poblacion/text())

 Obtener el número de sucursales cuya población sea Madrid.

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

 Obtener por cada sucursal su código y la suma de las aportaciones de las cuentas del tipo de pensiones.

//sucursal/concat(@codigo,"|",sum(cuenta/aportacion))

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

//cuenta[saldohaber>10000]/(numero|nombre|saldohaber)

 Obtener por cada sucursal con más de tres cuentas del tipo AHORRO, su código la suma del saldo debe de esas cuentas.

//sucursal[count(cuenta[@tipo="AHORRO"])>3]/concat(@codigo,"|",sum(cuenta/saldohaber))