



Esta obra está bajo una Licencia Creative Commons Atribución 4.0



LENGUAJES DE CONSULTA (V 6.0)

Prof. José Fager

Montevideo enero de 2020

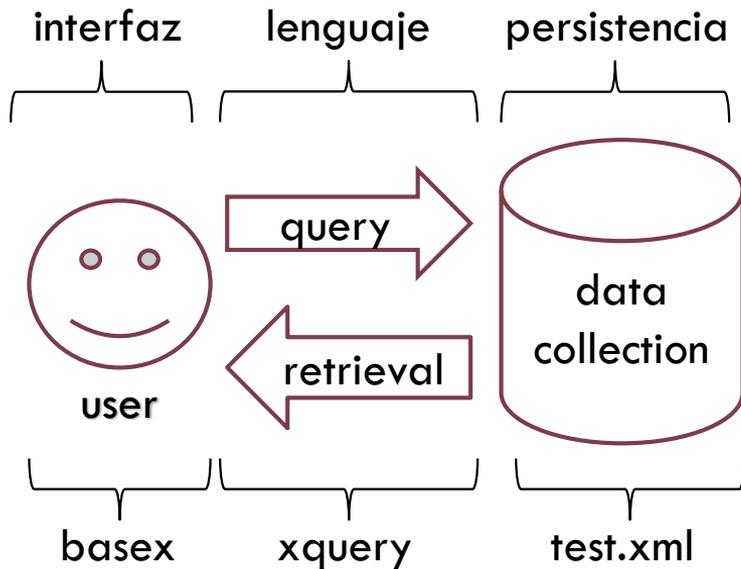
fic

IPAD II

A horizontal decorative bar at the top of the slide, consisting of a red rectangular section on the left and a teal rectangular section on the right.

Consultas con Xquery

Modelo de consulta

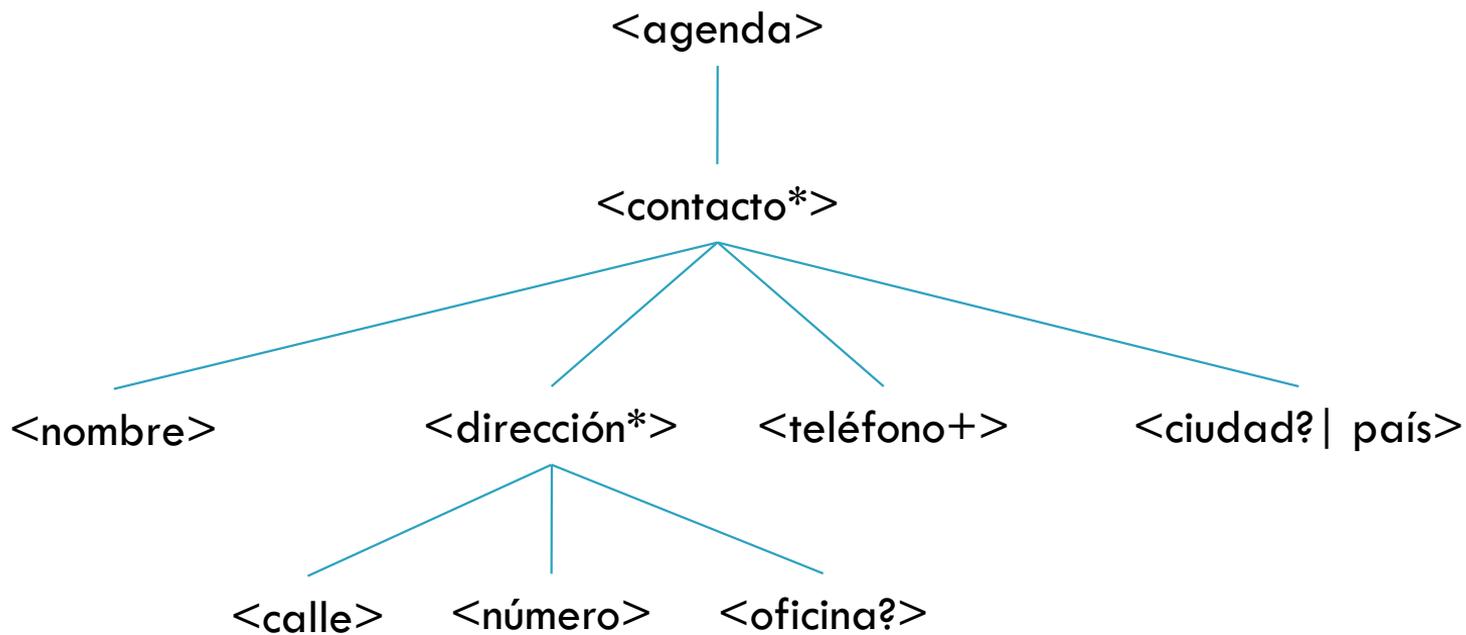


- **Data Collection (colección de datos):** es un conjunto de datos.
- **Query (consulta):** Es una expresión que se aplicada contra una colección de datos.
- **Retreival (recuperación):** Es el resultado de la query, es un sub-conjunto de la colección de datos, un conjunto de tuplas.
- **Tuplas:** Es un tipo de dato, que permite agrupar distintos valores como un mismo valor.

Xquery

- ❑ Xquery es un lenguaje que nos permite hacer consultas que obtienen como resultado un conjunto de tuplas.
- ❑ Nos permite además:
 - Procesar tuplas (for),
 - acumular tuplas (let),
 - filtrar las tuplas con determinados criterios (where),
 - eventualmente ordenar las tuplas (order by) y
 - construir una salida con el resultado obtenido (return).

Estructura de “agenda”



Persistencia "agenda_2.xml"

	Nombre	Direccion*			Telefono+		Ciudad? Pais	
		Calle	Numero	Oficina?			Ciudad?	Pais
a	Marigold	Ansina	2135		9985 9821	2400 2020	Montevideo	
b	Dodinas	Neptuno	6205	A3	9684 9755		Maldonado	
c	Holman				4025- 6617	2400 1010		Brasil
d	Rowan	Ansina	55		485 2519			



NOTA:

El archivo agenda_2.xml es muy similar a agenda2.xml, tiene el mismo contenido pero no usa accidentes gramaticales dentro de las etiquetas. Esto es para evitar problemas que puedan surgir al trabajar en BaseX, por incompatibilidad entre distintos sistemas de codificación.



RETURN

Sentencia "return"

```
return $result
```

- Una vez realizadas las eventuales acciones de obtención, filtrado y ordenado, es necesario retornar el resultado, para eso se usa la sentencia "return".

Ejemplo

- Retornar un resultado (imprime en la salida una cadena de texto)

file* [agenda_2.xml] - BaseX 8.6.7

Database Editor View Visualization Options Help

Editor

```
1 let $resultado:="HoLa MuNdO"  
2 return $resultado  
3
```

Result

HoLa MuNdO

OK 3 : 1



FOR

Sentencia "for"

```
for $resultado in doc('test.xml') path
```

- ▣ La sentencia “for” itera en la persistencia test.xml de una colección de datos y procesa uno a uno en “\$resultado” todas las tuplas de los objetos referenciados en path.

Ejemplo I

- Retornar todos los contactos.

	Nombre	Direccion*			Telefono+		Ciudad? Pais	
		Calle	Numero	Oficina?			Ciudad?	Pais
a	Marigold	Ansina	2135		9985 9821	2400 2020	Montevideo	
b	Dodinas	Neptuno	6205	A3	9684 9755		Maldonado	
c	Holman				4025- 6617	2400 1010		Brasil
d	Rowan	Ansina	55		485 2519			



- Debería retorna todos los datos de todos los contactos.

Ejemplo I – en BaseX

The screenshot displays the BaseX 8.6.7 application window. The title bar reads "file* [agenda_2.xml] - BaseX 8.6.7". The menu bar includes "Database", "Editor", "View", "Visualization", "Options", and "Help".

The interface is split into two main panes:

- Editor:** Contains an XSLT script with the following code:

```
1 for $resultado in doc('agenda_2.xml') //contacto
2 return $resultado
3
```
- Result:** Displays the XML output of the script, listing three contact entries:

```
<contacto>
  <nombre>Marigol</nombre>
  <direccion>
    <calle>Ansina</calle>
    <numero>2135</numero>
  </direccion>
  <telefono>9985 9821</telefono>
  <telefono>2400 2020</telefono>
  <ciudad>Montevideo</ciudad>
</contacto>
<contacto>
  <nombre>Dodinas</nombre>
  <direccion>
    <calle>Neptuno</calle>
    <numero>6205</numero>
    <oficina>A3</oficina>
  </direccion>
  <telefono>9684 9755</telefono>
  <ciudad>Maldonado</ciudad>
</contacto>
<contacto>
  <nombre>Holman</nombre>
  <telefono>4025-6617</telefono>
  <telefono>2400 1010</telefono>
  <pais>Brasil</pais>
</contacto>
<contacto>
  <nombre>Rowan</nombre>
  <direccion>
    <calle>Ansina</calle>
    <numero>55</numero>
  </direccion>
  <telefono>485 2519</telefono>
</contacto>
```

At the bottom left, there is a green checkmark icon and the text "OK". At the bottom right, the text "3 : 1" is displayed.

Ejemplo II

- Retornar todos los nombres de los contactos.

	Nombre	Direccion*			Telefono+		Ciudad? Pais	
		Calle	Numero	Oficina?			Ciudad?	Pais
a	Marigold	Ansina	2135		9985 9821	2400 2020	Montevideo	
b	Dodinas	Neptuno	6205	A3	9684 9755		Maldonado	
c	Holman				4025- 6617	2400 1010		Brasil
d	Rowan	Ansina	55		485 2519			



- Debería retorna solo los datos que están bajo la columna "Nombre".

Ejemplo II – en BaseX

The screenshot displays the BaseX 8.6.7 application window. The title bar reads "file* [agenda_2.xml] - BaseX 8.6.7". The menu bar includes "Database", "Editor", "View", "Visualization", "Options", and "Help". The "Editor" pane contains the following XSLT code:

```
1 for $resultado in doc('agenda_2.xml') //contacto/nombre
2 return $resultado
3
```

The "Result" pane shows the output of the XSLT transformation:

```
<nombre>Marigol</nombre>
<nombre>Dodinas</nombre>
<nombre>Holman</nombre>
<nombre>Rowan</nombre>
```

At the bottom left, there is a green checkmark icon and the text "OK". At the bottom right, the zoom level is indicated as "3 : 1".

Ejemplo III

- Retornar todas las calles de los contactos.

	Nombre	Direccion*			Telefono+		Ciudad? Pais	
		Calle	Numero	Oficina?			Ciudad?	Pais
a	Marigold	Ansina	2135		9985 9821	2400 2020	Montevideo	
b	Dodinas	Neptuno	6205	A3	9684 9755		Maldonado	
c	Holman				4025- 6617	2400 1010		Brasil
d	Rowan	Ansina	55		485 2519			



- Debería retorna solo los datos que están bajo la columna "Calle".

Ejemplo III – en BaseX

The screenshot displays the BaseX 8.6.7 application window. The title bar reads "file* [agenda_2.xml] - BaseX 8.6.7". The menu bar includes "Database", "Editor", "View", "Visualization", "Options", and "Help". The "Editor" pane contains the following XQuery code:

```
1 for $resultado in doc('agenda_2.xml') //contacto/direccion/calle
2 return $resultado
3
```

The "Result" pane shows the output of the query:

```
<calle>Ansina</calle>
<calle>Neptuno</calle>
<calle>Ansina</calle>
```

At the bottom left, there is a green checkmark and the text "OK". At the bottom right, the text "3 : 1" is displayed.



WHERE

Sentencia "where"

```
for $resultado in doc('test.xml') path
where expression
```

- ▣ Filtra las tuplas obtenidas con la sentencia "for". Define una expresión (expression) que evalúa cada tupla obtenida con "for" y si el resultado de la evaluación es falso descarta la tupla, sino la tiene en cuenta.

Ejemplo I

- Retornar todos los contactos que tienen alguna dirección en la calle “Ansina”.

	Nombre	Direccion*			Telefono+		Ciudad? Pais	
		Calle	Numero	Oficina?			Ciudad?	Pais
a	Marigold	Ansina	2135		9985 9821	2400 2020	Montevideo	
b	Dodinas	Neptuno	6205	A3	9684 9755		Maldonado	
c	Holman				4025- 6617	2400 1010		Brasil
d	Rowan	Ansina	55		485 2519			



- Debería retorna solo los datos de las filas “a” y “d”.

Ejemplo I – en BaseX

The screenshot displays the BaseX 8.6.7 application window. The title bar reads "file* [agenda_2.xml] - BaseX 8.6.7". The menu bar includes "Database", "Editor", "View", "Visualization", "Options", and "Help". The "Editor" pane on the left contains the following XQuery code:

```
1 for $resultado in doc('agenda_2.xml') //contacto
2 where $resultado/direccion/calle= 'Ansina'
3 return $resultado
4
```

The "Result" pane on the right shows the XML output of the query:

```
<contacto>
  <nombre>Marigol</nombre>
  <direccion>
    <calle>Ansina</calle>
    <numero>2135</numero>
  </direccion>
  <telefono>9985 9821</telefono>
  <telefono>2400 2020</telefono>
  <ciudad>Montevideo</ciudad>
</contacto>
<contacto>
  <nombre>Rowan</nombre>
  <direccion>
    <calle>Ansina</calle>
    <numero>55</numero>
  </direccion>
  <telefono>485 2519</telefono>
</contacto>
```

At the bottom left, there is a green checkmark icon and the text "OK". At the bottom right, the text "4 : 1" is displayed.

Ejemplo II

- Retornar todos los contactos que tienen más de 1 teléfono.

	Nombre	Direccion*			Telefono+		Ciudad? Pais	
		Calle	Numero	Oficina?			Ciudad?	Pais
a	Marigold	Ansina	2135		9985-9821	2400-2020	Montevideo	
b	Dodinas	Neptuno	6205	A3	9684-9755		Maldonado	
c	Holman				4025-6617	2400-1010		Brasil
d	Rowan	Ansina	55		485 2519			



- Debería retorna solo los datos de las filas “a” y “c”.

Ejemplo II – en BaseX

The screenshot shows the BaseX 8.6.7 application window. The title bar reads "file* [agenda_2.xml] - BaseX 8.6.7". The menu bar includes "Database", "Editor", "View", "Visualization", "Options", and "Help". The interface is split into two main panes: "Editor" on the left and "Result" on the right.

Editor Pane: Contains an XQuery query with the following code:

```
1 for $resultado in doc('agenda_2.xml') //contacto
2 where count($resultado/telefono)>1
3 return $resultado
4
```

The status bar at the bottom left shows a green checkmark and "OK". The status bar at the bottom right shows "4 : 1".

Result Pane: Displays the XML output of the query:

```
<contacto>
  <nombre>Marigol</nombre>
  <direccion>
    <calle>Ansina</calle>
    <numero>2135</numero>
  </direccion>
  <telefono>9985 9821</telefono>
  <telefono>2400 2020</telefono>
  <ciudad>Montevideo</ciudad>
</contacto>
<contacto>
  <nombre>Holman</nombre>
  <telefono>4025-6617</telefono>
  <telefono>2400 1010</telefono>
  <pais>Brasil</pais>
</contacto>
```

Ejemplo III

- Retornar todas las direcciones que tienen como calle a “Ansina”.

	Nombre	Direccion*			Telefono+		Ciudad? Pais	
		Calle	Numero	Oficina?			Ciudad?	Pais
a	Marigold	Ansina	2135		9985 9821	2400 2020	Montevideo	
b	Dodinas	Neptuno	6205	A3	9684 9755		Maldonado	
c	Holman				4025- 6617	2400 1010		Brasil
d	Rowan	Ansina	55		485 2519			



- Debería retornar solo los datos bajo la columna “Direccion” de las filas “a” y “d”.

Ejemplo III – en BaseX

The screenshot displays the BaseX 8.6.7 application window. The title bar reads "file* [agenda_2.xml] - BaseX 8.6.7". The menu bar includes "Database", "Editor", "View", "Visualization", "Options", and "Help". The "Editor" pane on the left contains the following XQuery code:

```
1 for $resultado in doc('agenda_2.xml') //contacto/direccion
2 where $resultado/calle= 'Ansina'
3 return $resultado
4
```

The "Result" pane on the right shows the XML output:

```
<direccion>
  <calle>Ansina</calle>
  <numero>2135</numero>
</direccion>
<direccion>
  <calle>Ansina</calle>
  <numero>55</numero>
</direccion>
```

At the bottom left, there is a green checkmark icon and the text "OK". At the bottom right, the zoom level "4 : 1" is displayed.

A horizontal bar at the top of the page, divided into a red section on the left and a teal section on the right.

OPERADORES

Sentencia "where" con operadores

```
for $resultado in doc('test') path
where expression1 op1 expression2 op2 expression3
```

- ▣ Funciona igual que la anterior, pero el filtrado tiene múltiples expresiones (expression) que se combinan con diferentes operadores (op). Los operadores más comunes son “and”, “or” y “not”. Se suele separar las expresiones con paréntesis curvos para mejor comprensión del criterio de filtrado.

Ejemplo AND

- Retorna todos los contactos que tiene alguna dirección en la calle “Ansina” y más de 1 teléfono.

	Nombre	Direccion*			Telefono+		Ciudad? Pais	
		Calle	Numero	Oficina?			Ciudad?	Pais
a	Marigold	Ansina	2135		9985 9821	2400 2020	Montevideo	
b	Dodinas	Neptuno	6205	A3	9684 9755		Maldonado	
c	Holman				4025- 6617	2400 1010		Brasil
d	Rowan	Ansina	55		485 2519			



- Debería retorna solo los datos de la fila “a”.

Ejemplo AND – en BaseX

The screenshot displays the BaseX 8.6.7 application window. The title bar reads "file* [agenda_2.xml] - BaseX 8.6.7". The menu bar includes "Database", "Editor", "View", "Visualization", "Options", and "Help". The interface is split into two main panes: "Editor" on the left and "Result" on the right.

Editor Pane: Contains an XQuery query with four lines:

```
1 for $resultado in doc('agenda_2.xml') //contacto
2 where $resultado/direccion/calle='Ansina' and
3 count($resultado/telefono)>1
4 return $resultado
```

Result Pane: Displays the XML output of the query:

```
<contacto>
  <nombre>Marigol</nombre>
  <direccion>
    <calle>Ansina</calle>
    <numero>2135</numero>
  </direccion>
  <telefono>9985 9821</telefono>
  <telefono>2400 2020</telefono>
  <ciudad>Montevideo</ciudad>
</contacto>
```

At the bottom left of the window, there is a green checkmark icon and the text "OK". At the bottom right, the zoom level "4 : 1" is displayed.

Ejemplo OR

- Retorna todos los contactos que tiene alguna dirección en la calle “Ansina” o más de 1 teléfono.

	Nombre	Direccion*			Telefono+		Ciudad? Pais	
		Calle	Numero	Oficina?			Ciudad?	Pais
a	Marigold	Ansina	2135		9985-9821	2400-2020	Montevideo	
b	Dodinas	Neptuno	6205	A3	9684-9755		Maldonado	
c	Holman				4025-6617	2400-1010		Brasil
d	Rowan	Ansina	55		485 2519			



- Debería retorna solo los datos de las filas “a”, “c” y “d”.

Ejemplo OR – en BaseX

The screenshot shows the BaseX 8.6.7 interface with two panes: Editor and Result.

Editor Pane: Contains the following XQuery code:

```
1 for $resultado in doc('agenda_2.xml') //contacto
2 where $resultado/direccion/calle='Ansina' or
   count($resultado/telefono)>1
3 return $resultado
4
```

Result Pane: Displays the XML output of the query:

```
<contacto>
  <nombre>Marigol</nombre>
  <direccion>
    <calle>Ansina</calle>
    <numero>2135</numero>
  </direccion>
  <telefono>9985 9821</telefono>
  <telefono>2400 2020</telefono>
  <ciudad>Montevideo</ciudad>
</contacto>
<contacto>
  <nombre>Holman</nombre>
  <telefono>4025-6617</telefono>
  <telefono>2400 1010</telefono>
  <pais>Brasil</pais>
</contacto>
<contacto>
  <nombre>Rowan</nombre>
  <direccion>
    <calle>Ansina</calle>
    <numero>55</numero>
  </direccion>
  <telefono>485 2519</telefono>
</contacto>
```

The status bar at the bottom left shows a green checkmark and the text "OK". The bottom right corner of the editor pane shows "4 : 1".

Ejemplo NOT

- Retorna todos los contactos que tiene alguna dirección en la calle “Ansina” y más de 1 teléfono.

	Nombre	Direccion*			Telefono+		Ciudad? Pais	
		Calle	Numero	Oficina?			Ciudad?	Pais
a	Marigold	Ansina	2135		9985 9821	2400 2020	Montevideo	
b	Dodinas	Neptuno	6205	A3	9684 9755		Maldonado	
c	Holman				4025- 6617	2400 1010		Brasil
d	Rowan	Ansina	55		485 2519			



- Debería retorna solo los datos de la fila “a”.

Ejemplo NOT – en BaseX

The screenshot displays the BaseX 8.6.7 application window. The title bar reads "file* [agenda_2.xml] - BaseX 8.6.7". The menu bar includes "Database", "Editor", "View", "Visualization", "Options", and "Help". The interface is split into two main panes: "Editor" on the left and "Result" on the right.

Editor Pane: Contains an XQuery query with the following code:

```
1 for $resultado in doc('agenda_2.xml') //contacto
2 where $resultado/direccion/calle='Ansina' and
3 not (count($resultado/telefono)<=1)
4 return $resultado
```

Result Pane: Displays the XML output of the query:

```
<contacto>
  <nombre>Marigol</nombre>
  <direccion>
    <calle>Ansina</calle>
    <numero>2135</numero>
  </direccion>
  <telefono>9985 9821</telefono>
  <telefono>2400 2020</telefono>
  <ciudad>Montevideo</ciudad>
</contacto>
```

At the bottom left of the window, there is a green checkmark icon and the text "OK". At the bottom right, the zoom level "4 : 1" is displayed.

A horizontal bar at the top of the slide, divided into a red section on the left and a teal section on the right. The text 'Redefinición de variables' is written in white on the teal section.

Redefinición de variables

Redefinición de variables

A

```
let $resultado:=$resultado/path
```

B

```
return $resultado/path
```

- ▣ Permite cambiar el tipo de objeto contenido en la variable.
- ▣ Se puede redefinir una variables usando “let”. (A)
- ▣ Se puede redefinir la variable usando “return”. (B)

Ejemplo usando “let”

- Retorna todas las calles de los contactos que tiene más de 1 teléfono.

	Nombre	Direccion*			Telefono+		Ciudad? Pais	
		Calle	Numero	Oficina?			Ciudad?	Pais
a	Marigold	Ansina	2135		9985 9821	2400 2020	Montevideo	
b	Dodinas	Neptuno	6205	A3	9684 9755		Maldonado	
c	Holman				4025- 6617	2400 1010		Brasil
d	Rowan	Ansina	55		485 2519			



- Debería retorna solo los datos bajo la columna “Calle” de las filas “a” y “c”.

Ejemplo usando "let" – en BaseX

The screenshot displays the BaseX 8.6.7 application window. The title bar reads "file* [agenda_2.xml] - BaseX 8.6.7". The menu bar includes "Database", "Editor", "View", "Visualization", "Options", and "Help". The interface is split into two main panes: "Editor" on the left and "Result" on the right.

Editor Pane: Contains an XQuery script with four lines of code:

```
1 for $resultado in doc('agenda_2.xml') //contacto
2 where count($resultado/telefono)>1
3 let $resultado := $resultado/direccion/calle
4 return $resultado
5
```

The status bar at the bottom left shows a green checkmark and the text "OK". The status bar at the bottom right shows "5 : 1".

Result Pane: Displays the output of the query as a single XML element:

```
<calle>Ansina</calle>
```

Ejemplo usando “return”

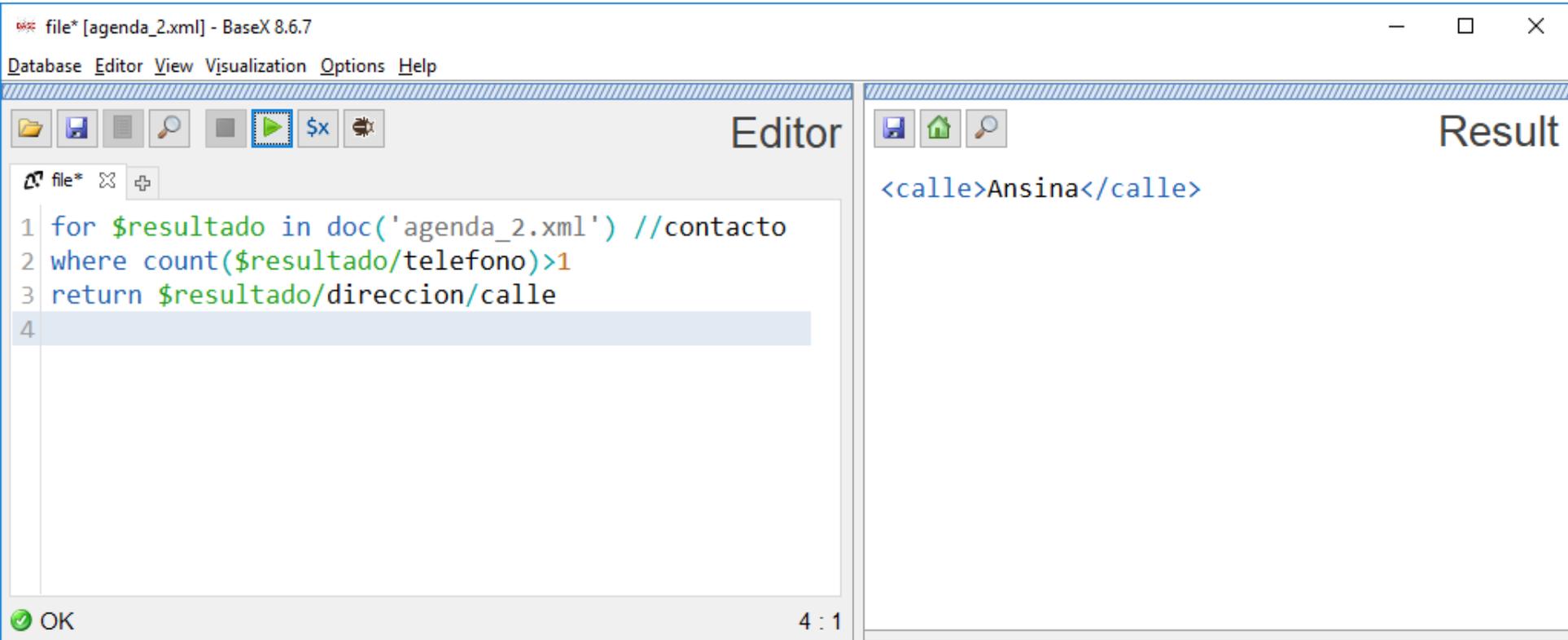
- Retorna todas las calles de los contactos que tiene más de 1 teléfono.

	Nombre	Direccion*			Telefono+		Ciudad? Pais	
		Calle	Numero	Oficina?			Ciudad?	Pais
a	Marigold	Ansina	2135		9985 9821	2400 2020	Montevideo	
b	Dodinas	Neptuno	6205	A3	9684 9755		Maldonado	
c	Holman				4025- 6617	2400 1010		Brasil
d	Rowan	Ansina	55		485 2519			



- Debería retorna solo los datos bajo la columna “Calle” de las filas “a” y “c”.

Ejemplo usando "return" – en BaseX



The screenshot displays the BaseX 8.6.7 application window. The title bar reads "file* [agenda_2.xml] - BaseX 8.6.7". The menu bar includes "Database", "Editor", "View", "Visualization", "Options", and "Help". The interface is split into two main panes: "Editor" on the left and "Result" on the right.

Editor Pane: Contains an XQuery script with four lines:

```
1 for $resultado in doc('agenda_2.xml') //contacto
2 where count($resultado/telefono)>1
3 return $resultado/direccion/calle
4
```

Result Pane: Displays the output of the query as a single XML element: `<calle>Ansina</calle>`.

At the bottom left, there is a green checkmark icon and the text "OK". At the bottom right, the text "4 : 1" is visible.



ORDER BY

Sentencia "order by"

```
for $resultado in doc('test.xml') path
where expression
order by condition
```

- ▣ Además de obtener y filtra las tuplas obtenidas, también podemos ordenarlas imponiendo una condición (condition).

Ejemplo I

- Retorna todos los nombres ordenados alfabéticamente.

	Nombre	Direccion*			Telefono+		Ciudad? Pais	
		Calle	Numero	Oficina?			Ciudad?	Pais
a	Marigold	Ansina	2135		9985 9821	2400 2020	Montevideo	
b	Dodinas	Neptuno	6205	A3	9684 9755		Maldonado	
c	Holman				4025- 6617	2400 1010		Brasil
d	Rowan	Ansina	55		485 2519			



- Debería retorna solo los datos bajo la columna “Nombre” ordenados alfabéticamente.

Ejemplo I – en BaseX

The screenshot displays the BaseX 8.6.7 application window. The title bar reads "file* [agenda_2.xml] - BaseX 8.6.7". The menu bar includes "Database", "Editor", "View", "Visualization", "Options", and "Help". The "Editor" pane contains the following XQuery script:

```
1 for $resultado in doc('agenda_2.xml') //contacto/nombre
2 order by $resultado
3 return $resultado
4
```

The "Result" pane shows the output of the query as XML fragments:

```
<nombre>Dodinas</nombre>
<nombre>Holman</nombre>
<nombre>Marigol</nombre>
<nombre>Rowan</nombre>
```

At the bottom left, there is a green checkmark icon and the text "OK". At the bottom right, the zoom level is indicated as "4 : 1".

Ejemplo II

- Retorna todos los nombres de los contactos que no tienen número de oficina, ordenados alfabéticamente.

	Nombre	Direccion*			Telefono+		Ciudad? Pais	
		Calle	Numero	Oficina?			Ciudad?	Pais
a	Marigold	Ansina	2135		9985 9821	2400 2020	Montevideo	
b	Dodinas	Neptuno	6205	A3	9684 9755		Maldonado	
c	Holman				4025- 6617	2400 1010		Brasil
d	Rowan	Ansina	55		485 2519			



- Debería retorna solo los datos bajo la columna “Nombre”; de las filas “a”, “c” y “d”; ordenados alfabéticamente.

Ejemplo II – en BaseX

The screenshot shows the BaseX 8.6.7 application window. The title bar reads "file* [agenda_2.xml] - BaseX 8.6.7". The menu bar includes "Database", "Editor", "View", "Visualization", "Options", and "Help". The interface is split into two main panes: "Editor" on the left and "Result" on the right.

Editor Pane: Contains an XQuery query with the following text:

```
1 for $resultado in doc('agenda_2.xml') //contacto
2 where count($resultado/direccion/oficina)=0
3 order by $resultado/nombre
4 return $resultado/nombre
5
```

Result Pane: Displays the output of the query as XML fragments:

```
<nombre>Holman</nombre>
<nombre>Marigol</nombre>
<nombre>Rowan</nombre>
```

At the bottom left, there is a green checkmark icon and the text "OK". At the bottom right, the zoom level "5 : 1" is displayed.

Ejemplo III

- Retorna todas las direcciones, ordenadas alfabéticamente por la calle.

	Nombre	Direccion*			Telefono+		Ciudad? Pais	
		Calle	Numero	Oficina?			Ciudad?	Pais
a	Marigold	Ansina	2135		9985-9821	2400-2020	Montevideo	
b	Dodinas	Neptuno	6205	A3	9684-9755		Maldonado	
c	Holman				4025-6617	2400-1010		Brasil
d	Rowan	Ansina	55		485 2519			



- Debería retorna solo los datos bajo la columna “Direccion”, ordenados alfabéticamente por la “Calle”.

Ejemplo III – en BaseX

The screenshot displays the BaseX 8.6.7 application window. The title bar reads "file* [agenda_2.xml] - BaseX 8.6.7". The menu bar includes "Database", "Editor", "View", "Visualization", "Options", and "Help". The interface is split into two main panes: "Editor" on the left and "Result" on the right.

Editor Pane: Contains an XQuery script with the following code:

```
1 for $resultado in doc('agenda_2.xml') //contacto/direccion
2 order by $resultado/calle
3 return $resultado
4
```

Result Pane: Displays the XML output of the query, showing three <direction> elements with their respective street names and numbers:

```
<direccion>
  <calle>Ansina</calle>
  <numero>2135</numero>
</direccion>
<direccion>
  <calle>Ansina</calle>
  <numero>55</numero>
</direccion>
<direccion>
  <calle>Neptuno</calle>
  <numero>6205</numero>
  <oficina>A3</oficina>
</direccion>
```

The status bar at the bottom left shows a green checkmark and the text "OK". The bottom right corner of the editor pane shows "4 : 1".