

```
for $x in doc("menu3")//PLATO[@category="POSTRE"]
return $x
```

BaseX 9.6.4 interface showing a query result for 'menu3' with 3 results. The query is: `for $x in doc("menu3")//PLATO[@category="POSTRE"] return $x`. The result pane shows XML snippets for 'tarta de queso', 'macedonia de frutas', and 'helado de vainilla'. The info pane shows the compiled query and optimized query.

```
for $x in doc("menu3")//PLATO[price<10]
return $x
```

BaseX 9.6.4 interface showing a query result for 'menu3' with 6 results. The query is: `for $x in doc("menu3")//PLATO[price<10] return $x`. The result pane shows XML snippets for 'ensalada mixta', 'sopa de marisco', and 'tarta de queso'. The info pane shows the compiled query and optimized query.

```
for $x in doc ("menu")// INGREDIENTES = pescado
```

File [menu] - BaseX 9.6.4

Database Editor View Visualization Options Help

Find Find...

Columns: PhyDesktop\basex\base\

\*.xml \*.xml\*

Find contents...

bin

data

etc

io

logo

src

webapp

Event.log (4044 kb)

CHANGELOG (46 kb)

LICENSE (1524 kb)

readme.txt (1602 kb)

1 for \$x in doc ("menu")//INGREDIENTES= pescado

2

3 return \$x

OK

3 : 10

Editor

menu.xml

PLATO RESTAURANTE

name

ensalada mixta

price

4

ingredientes

name

sopa de marisco

price

9

ingredientes

name

ensalada de pasta

price

6

ingredientes

name

paella de marisco

price

14

ingredientes

name

paella de marisco

price

14

ingredientes

name

tarta de queso

price

8

ingredientes

name

helado de vainilla

price

4

ingredientes

1 Result, 5 b

false

Result

Compiling:

rewrite fn doc(uri) to document-node() item: doc("menu") -> db open-pre("menu", 0)

remove step without results: INGREDIENTES

rewrite cached step to empty sequence: INGREDIENTES -> ()

rewrite cached path to empty sequence: db open-pre("menu", 0)/descendant-or-self::node()()-> ()

remove step without results: pescado

rewrite cached step to empty sequence: pescado -> ()

rewrite cached path to empty sequence: db open-pre("menu", 0)/()-> ()

rewrite empty sequence to xs:boolean item: () -> false()

rewrite fn boolean(items) to xs:boolean item: boolean(false()) -> false()

rewrite = comparison to xs:boolean item: () = () -> false()

rewrite for to let: for \$x, 0 in false()

inline let \$x, 0 = false()

simplify FLWOR expression: false()

Optimized Query:

false()

Query:

for \$x in doc ("menu")//INGREDIENTES= pescado return \$x

Result:

db open("menu", "menu.xml")//MENU RESTAURANTE//PLATO//ingredientes/text()

team.microsoft.com está compartiendo tu pantalla. [Dejar de compartir](#) [Ocultar](#)

23:10

18:13

10/02/2022

The image shows a Windows desktop with a Microsoft Edge browser window open. The browser's address bar shows the file path 'C:\Users\Pedro\Desktop\basen\basen.js'. The Visual Studio Code editor is open, displaying a JavaScript file named 'menu.js'. The code in the editor is as follows:

```
for $x in doc('menu')//PLATO[last()]
1
2
3 return $x
```

The editor's status bar at the bottom indicates '1 Result, 192 b'. The Microsoft Edge DevTools console is open, showing the output of the query. The console displays a table of restaurant menu items. The table has columns for 'Plato', 'Precio', 'Ingredientes', and 'Imagen'. The data is as follows:

Plato	Precio	Ingredientes	Imagen
ensalada mixta	4		
ensalada de pasta	6		
paella de marisco	14		
tarta de queso	8		
helado de vainilla	4		

The console also shows the 'Query' and 'Result' sections, indicating that the query returned 1 item and that the result was updated, printed, and read. The status bar at the bottom of the browser shows 'Time required: 67.52 ms'.

File\* [menu] - BaseX 8.6.4

Database Editor View Visualization Options Help

Find Find...

Editor

```

1. for $x in doc('menu')//name
2. return $x

```

9 Results, 335 b

```

<name>ensalada mixta</name>
<name>ensalada de pasta</name>
<name>sopa de marisco</name>
<name>filetes de rosada a la plancha con patatas cocidas</name>
<name>coliflor de cerdo con salsa roquefort</name>
<name>paella de marisco</name>
<name>tarta de queso</name>
<name>helado de vainilla</name>
<name>sopa de marisco</name>

```

Result

Compiling:

- rewrite fn doc(uri) to document-node() item: doc("menu") -> db:open-pre("menu", 0)
- merge: descendant: name
- convert to child steps: descendant: name
- inline for \$x, 0 in db:open-pre("menu", 0)\*MENU\_RESTAURANTE\*PLATO\*name
- simplify FLWOR expression: db:open-pre("menu", 0)\*MENU\_RESTAURANTE\*PLATO\*name

Optimized Query:

db:open-pre("menu", 0)\*MENU\_RESTAURANTE\*PLATO\*name

Query:

for \$x in doc('menu')//name return \$x

Result:

- Hits: 9 items
- Updated: 0 items
- Printed: 335 b
- Read Locking: menu
- Write Locking: (none)

Timing:

- Parsing: 0.35 ms
- Compiling: 1.05 ms

### 3 OPCIONALES

for \$x in doc ("menu")//name | //ingredientes [position()=1]  
return \$x

File\* [menu] - BaseX 8.6.4

Database Editor View Visualization Options Help

Find Find...

Editor

```

1. for $x in doc('menu')//name | //ingredientes [position()=1]
2. return $x

```

18 Results, 1134 b

```

<name>ensalada mixta</name>
<ingredientes>lechuga, tomate, huevo, zanahoria, atún</ingredientes>
<name>ensalada de pasta</name>
<ingredientes>macarrones, tomate natural, queso, atún</ingredientes>
<name>sopa de marisco</name>
<ingredientes>patatas, nape, gambas, almejas, mejillones, arroz</ingredientes>
<name>filetes de rosada a la plancha con patatas cocidas</name>
<ingredientes>filetes de rosada fresca patatas de la huerta cocidas con un pimiento</ingredientes>
<name>coliflor de cerdo con salsa roquefort</name>
<ingredientes>filetes de cerdo ibérico con salsa roquefort de la casa con patatas fritas</ingredientes>
<name>paella de marisco</name>
<ingredientes>arroz con gambas almejas mejillones y carne de cerdo además de un sofrito de verduras</ingredientes>
<name>tarta de queso</name>
<ingredientes>tarta de queso casera con mermelada de fresa y nata</ingredientes>
<name>helado de vainilla</name>
<ingredientes>trozos de fruta fresca cortada en rodajas</ingredientes>
<name>sopa de marisco</name>
<ingredientes>helado de vainilla hecho a mano con un toque de sabor de nuestro chef</ingredientes>

```

Result

Compiling:

- rewrite fn doc(uri) to document-node() item: doc("menu") -> db:open-pre("menu", 0)
- merge: descendant: name
- convert to child steps: descendant: name
- rewrite content value to document-node() item: -> db:open-pre("menu", 0) -> db:open-pre("menu", 0)
- rewrite uri root(nodes) to document-node() item: uri root(db:open-pre("menu", 0)) -> db:open-pre("menu", 0)
- rewrite = comparison to positional access: (position() = 1) -> fn:position() = 1
- inline for \$x, 0 in db:open-pre("menu", 0)\*MENU\_RESTAURANTE\*PLATO\*name union db:open-pre("menu", 0)descendant-or-self:node(//ingredientes[fn:position() = 1])
- simplify FLWOR expression: (db:open-pre("menu", 0)\*MENU\_RESTAURANTE\*PLATO\*name union db:open-pre("menu", 0)descendant-or-self:node(//ingredientes[fn:position() = 1])

Optimized Query:

(db:open-pre("menu", 0)\*MENU\_RESTAURANTE\*PLATO\*name union db:open-pre("menu", 0)descendant-or-self:node(//ingredientes[fn:position() = 1])

Query:

for \$x in doc('menu')//name | //ingredientes [position()=1] return \$x

Result:

- Hits: 18 items
- Updated: 0 items

for \$x in doc ("menu")//PLATO[@category='ENTRANTE']  
return \$x

File [menu] - BaseX 8.6.4

Database Editor View Visualization Options Help

Find Find...

Context: db:open("menu")

Editor

```

1 for $x in doc ("menu")//PLATO[@category='ENTRANTE']
2
3 return $x

```

3 Results, 499 b

Result

```

<PLATO category="ENTRANTE">&lt;<name>ensalada mixta</name>
<price>4</price>
<ingredientes>lechuga, tomate, huevo, zanahoria, atún</ingredientes>
</PLATO>
<PLATO category="ENTRANTE">
<name>ensalada de pasta</name>
<price>6</price>
<ingredientes>macarrones, tomate natural, queso, atún</ingredientes>
</PLATO>
<PLATO category="ENTRANTE">
<name>sopa de marisco</name>
<price>9</price>
<ingredientes>patatas, rape, gambas, almejas, mejillones, arroz</ingredientes>
</PLATO>

```

3 Results

menu.xml

PLATO	name	price	ingredientes
ensalada mixta	4	lechuga, tomate, huevo, zanahoria, atún	
ensalada de pasta	6	macarrones, tomate natural, queso, atún	
sopa de marisco	9	patatas, rape, gambas, almejas, mejillones, arroz	

Compiling:

- rewrite in doc(uri) to document-node() term: doc("menu") -> db:open-pre("menu", 0)
- merge: descendant: PLATO[@category = "ENTRANTE"]
- apply attribute index for "ENTRANTE"
- inline for \$x, 0 in db:attribute("menu", "ENTRANTE")/self:attribute(category)/parent: PLATO
- simplify FLWOR expression: db:attribute("menu", "ENTRANTE")/self:attribute(category)/parent: PLATO

Optimized Query:

db:attribute("menu", "ENTRANTE")/self:attribute(category)/parent: PLATO

Query:

for \$x in doc ("menu")//PLATO[@category='ENTRANTE'] return \$x

Result:

- Hits: 3 items
- Updated: 0 items
- Printed: 499 b
- Read Locking: none
- Write Locking: none

Timing:

- Parsing: 0.32 ms
- Compiling: 28.98 ms

Time required: 32.25 ms

tramu.microsoft.com está compartiendo tu pantalla. Dejar de compartir Ocultar

for \$x in doc ("menu")//PLATO[@category='PRINCIPAL']  
return \$x

File [menu] - BaseX 8.6.4

Database Editor View Visualization Options Help

Find Find...

Context: db:open("menu")

Editor

```

1 for $x in doc ("menu")//PLATO[@category='PRINCIPAL']
2
3 return $x

```

3 Results, 665 b

Result

```

<PLATO category="PRINCIPAL">
<name>Filetes de rosada a la plancha con patatas cocidas</name>
<price>12</price>
<ingredientes>Filetes de rosada fresca patatas de la huerta cocidas con un perejil</ingredientes>
</PLATO>
<PLATO category="PRINCIPAL">
<name>solomillo de cerdo con salsa roquefort</name>
<price>18</price>
<ingredientes>Filetes de cerdo ibérico con salsa roquefort de la casa con patatas fritas</ingredientes>
</PLATO>
<PLATO category="PRINCIPAL">
<name>paella de marisco</name>
<price>14</price>
<ingredientes>arroz con gambas almejas mejillones y carne de cerdo además de un sofrito de verduras</ingredientes>
</PLATO>

```

3 Results

menu.xml

PLATO	name	price	ingredientes
ensalada mixta	4	lechuga, tomate, huevo, zanahoria, atún	
ensalada de pasta	6	macarrones, tomate natural, queso, atún	
sopa de marisco	9	patatas, rape, gambas, almejas, mejillones, arroz	
Filetes de rosada a la plancha con patatas cocidas	12	Filetes de rosada fresca patatas de la huerta cocidas con un perejil	
solomillo de cerdo con salsa roquefort	18	Filetes de cerdo ibérico con salsa roquefort de la casa con patatas fritas	
paella de marisco	14	arroz con gambas almejas mejillones y carne de cerdo además de un sofrito de verduras	

Compiling:

- rewrite in doc(uri) to document-node() term: doc("menu") -> db:open-pre("menu", 0)
- merge: descendant: PLATO[@category = "PRINCIPAL"]
- apply attribute index for "PRINCIPAL"
- inline for \$x, 0 in db:attribute("menu", "PRINCIPAL")/self:attribute(category)/parent: PLATO
- simplify FLWOR expression: db:attribute("menu", "PRINCIPAL")/self:attribute(category)/parent: PLATO

Optimized Query:

db:attribute("menu", "PRINCIPAL")/self:attribute(category)/parent: PLATO

Query:

for \$x in doc ("menu")//PLATO[@category='PRINCIPAL'] return \$x

Result:

- Hits: 3 items
- Updated: 0 items
- Printed: 665 b
- Read Locking: none
- Write Locking: none

Timing:

- Parsing: 0.35 ms
- Compiling: 2.06 ms

Time required: 3.69 ms

tramu.microsoft.com está compartiendo tu pantalla. Dejar de compartir Ocultar