

EJERCICIO 1 U5

Database Editor View Visualization Options Help

XQuery /productos/producto[1]

C:\Program Files (x86)\Base

Find contents...

EJERCICIOS

Videojuegos.xml Productos.xml

```
<nombre>Altavoces Sound Pivody</nombre>
<fabricante>Pivody</fabricante>
<precio>24.99</precio>
</producto>
<producto>
  <nombre>Cámara web Log C70</nombre>
  <fabricante>Log</fabricante>
  <precio>34.99</precio>
</producto>
</productos>
```

OK 53 : 1

1 Result, 132 b

Result

```
<producto>
  <nombre>Portátil Lenpad 3</nombre>
  <fabricante>Lenpad</fabricante>
  <precio>699.99</precio>
</producto>
```

Optimizing:

- rewrite context value: .-> db.get-pre("Productos", 0)
- rewrite util.root(nodes): util.root(db.get-pre("Productos", 0)) -> db.get-pre("Productos", 0)

Optimized Query:

```
db.get-pre("Productos", 0)/productos/producto[1]
```

Query:

```
/productos/producto[1]
```

Result:

productos.xml

productos

producto

nombre

precio

699.99

fabricante

Lenpad

producto

nombre

precio

99.99

fabricante

Queens

producto

nombre

precio

169.99

fabricante

GL

producto

nombre

precio

19.99

fabricante

Logisur

producto

nombre

precio

229.99

fabricante

Trek

producto

nombre

precio

14.99

fabricante

Micro

producto

nombre

precio

69.99

fabricante

Seahard

producto

nombre

precio

24.99

fabricante

Pivody

producto

nombre

precio

34.99

fabricante

Log

Database Editor View Visualization Options Help

XQuery /productos/producto[fabricante='Apple' or fabricante='Pivody']

C:\Program Files (x86)\Base

Find contents...

EJERCICIOS

Videojuegos.xml Productos.xml

```
<nombre>Altavoces Sound Pivody</nombre>
<fabricante>Pivody</fabricante>
<precio>24.99</precio>
</producto>
<producto>
  <nombre>Cámara web Log C70</nombre>
  <fabricante>Log</fabricante>
  <precio>34.99</precio>
</producto>
</productos>
```

OK 53 : 1

1 Result, 135 b

Result

```
<producto>
  <nombre>Altavoces Sound Pivody</nombre>
  <fabricante>Pivody</fabricante>
  <precio>24.99</precio>
</producto>
```

Compiling:

- rewrite list to xs:string sequence: ("Apple", "Pivody")
- simplify or: ((fabricante = "Apple") or (fabricante = "Pivody"))
- rewrite fn:boolean(input): boolean(fabricante = ("Apple", "Pivody")) -> (fabricante = ("Apple", "Pivody"))

Optimizing:

- rewrite context value: .-> db.get-pre("Productos", 0)
- rewrite util.root(nodes): util.root(db.get-pre("Productos", 0)) -> db.get-pre("Productos", 0)

productos.xml

productos

producto

nombre

precio

699.99

fabricante

Lenpad

producto

nombre

precio

99.99

fabricante

Queens

producto

nombre

precio

169.99

fabricante

GL

producto

nombre

precio

19.99

fabricante

Logisur

producto

nombre

precio

229.99

fabricante

Trek

producto

nombre

precio

14.99

fabricante

Micro

producto

nombre

precio

69.99

fabricante

Seahard

producto

nombre

precio

24.99

fabricante

Pivody

```
40 <precio>79.99</precio>
41 </producto>
42 <producto>
43 <nombre>Altavoces Sound Pivody</nombre>
44 <fabricante>Pivody</fabricante>
45 <precio>24.99</precio>
46 </producto>
47 <producto>
48 <nombre>Cámara web Log C78</nombre>
49 <fabricante>Log</fabricante>
50 <precio>34.99</precio>
```

OK 43 : 35

1 Result, 22 b Result Total Time: 1.8 ms Info

<precio>24.99</precio>

Optimizing:
- rewrite context value: . -> db:get-pre("Productos", 0)
- rewrite util:root(nodes): util:root(db:get-pre("Productos", 0)) -> db:get-pre("Productos", 0)
Optimized Query:
db:get-pre("Productos", 0)/productos/producto[
contains(nombre, "Altavoces Sound Pivody")
]
/precio
Query:
/productos/producto[contains(nombre, 'Altavoces Sound Pivody')]/precio

Productos.xml			
productos			
producto	nombre	precio	producto
	Lenpad	699.99	
producto	fabricante	Logisur	
producto	nombre	precio	producto
	GL	169.99	
producto	fabricante	Trek	
producto	nombre	precio	producto
	Micro	14.99	
producto	fabricante	Seahard	
producto	nombre	precio	producto
	Pivody	24.99	
producto	fabricante	Log	
producto	nombre	precio	producto
	Queens	99.99	
producto	fabricante	Wort	
producto	nombre	precio	producto
	Cámara	79.99	
producto	fabricante	Log	

C:\Program Files (x86)\BaseX\data\EJERCICIOS1\Productos.xml [Productos] - BaseX 10.7

Database Editor View Visualization Options Help

XQuery 9 Results

Context: db:get("..." Editor

Videojuegos.xml Productos.xml

40 <precio>79.99</precio>
41 </producto>
42 </producto>
43 <nombre>Altavoces Sound Pivody</nombre>
44 <fabricante>Pivody</fabricante>
45 <precio>24.99</precio>
46 </producto>
47 <producto>
48 <nombre>Cámara web Log C70</nombre>
49 <fabricante>Log</fabricante>
50 <precio>34.99</precio>
51 </producto>
52
53

OK 43 : 35

9 Results, 396 b

Result

<nombre>Monitor GL 2M60M-B</nombre>
<nombre>Teclado Logisur K20</nombre>
<nombre>Mouse inalámbrico Micro 150</nombre>
<nombre>Disco duro externo Seahard Expansion</nombre>
<nombre>Memoria RAM Queenston HyperX force</nombre>
<nombre>Tablet Trek Galaxy Tab A7</nombre>
<nombre>Impresora Wortp DeskJet 20</nombre>
<nombre>Altavoces Sound Pivody</nombre>

Compiling:
- rewrite < comparison: (precio < 500) ->
precio <= 499.99999999999994

Optimizing:
- rewrite context value: .-> db:get-pre("Productos", 0)
- rewrite util:root(nodes): util:root(db:get-pre("Productos", 0)) -> db:get-pre("Productos", 0)

Optimized Query:
db:get-pre("Productos", 0)/productos/producto[precio <= 499.99999999999994]/nombre

Query:

Productos.xml

productos

producto

nombre

precio

fabricante

Lenpad

699.99

Logisur

19.99

Seahard

69.99

Micro

14.99

Queenston

99.99

Pivody

24.99

Log

34.99

Wortp

79.99

Database Editor View Visualization Options Help

XQuery 3 Results

Context: db:get("..." Editor

Videojuegos.xml Productos.xml

44 <nombre>Altavoces Sound Pivody</nombre>
45 <fabricante>Pivody</fabricante>
46 <precio>24.99</precio>
47 </producto>
48 <producto>
49 <nombre>Cámara web Log C70</nombre>
50 <fabricante>Log</fabricante>
51 <precio>34.99</precio>
52 </producto>
53 </productos>

OK 53 : 1

3 Results, 401 b

Result

<producto>
>
<nombre>Portátil Lenpad 3</nombre>
<fabricante>Lenpad</fabricante>
<precio>699.99</precio>
</producto>
<producto>
<nombre>Monitor GL 2M60M-B</nombre>
<fabricante>GL</fabricante>
<precio>169.99</precio>

Compiling:
- rewrite > comparison: (precio > 100) ->
precio >= 100.00000000000001

Optimizing:
- rewrite context value: .-> db:get-pre("Productos", 0)
- rewrite util:root(nodes): util:root(db:get-pre("Productos", 0)) -> db:get-pre("Productos", 0)

Optimized Query:
db:get-pre("Productos", 0)/productos/producto[precio >= 100]

productos.xml

productos

producto

nombre

precio

fabricante

Lenpad

699.99

Logisur

19.99

Seahard

69.99

Micro

14.99

Queenston

99.99

Pivody

24.99

Log

34.99

Wortp

79.99

Trek

229.99