



# TECNOLOGÍAS XML

## XPATH & XSLT

### Descripción breve

Proyecto del Segundo Trimestre sobre las Tecnologías XML (XSD,XPath,XSLT,XQuery).  
En este documento se representara las rutas XPATH realizadas, así como el XSLT que se ha diseñado.

Iván Hernández Fuentes

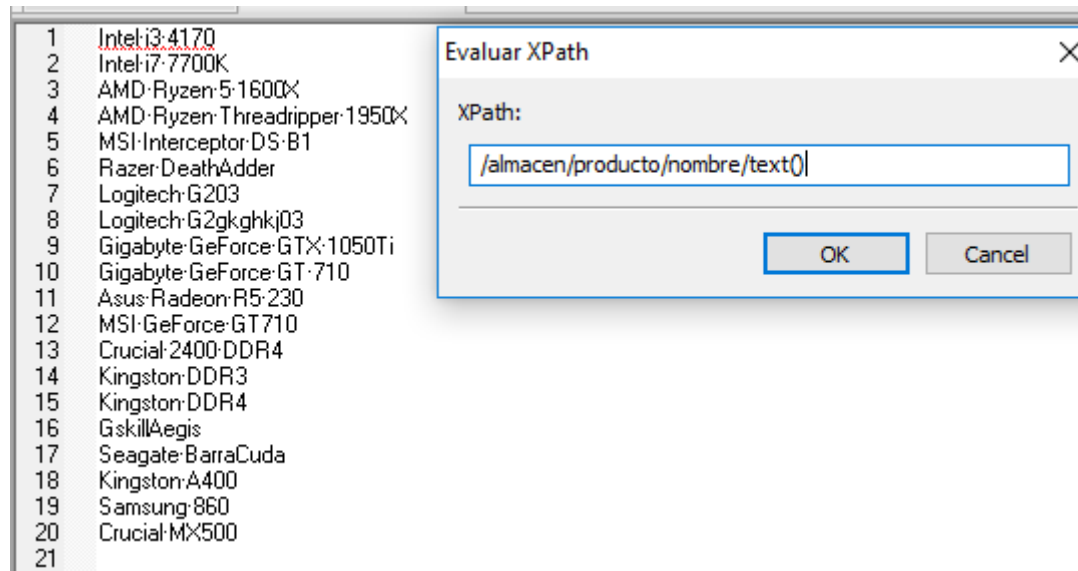
1º CFGS Desarrollo de Aplicaciones Web

# Índice

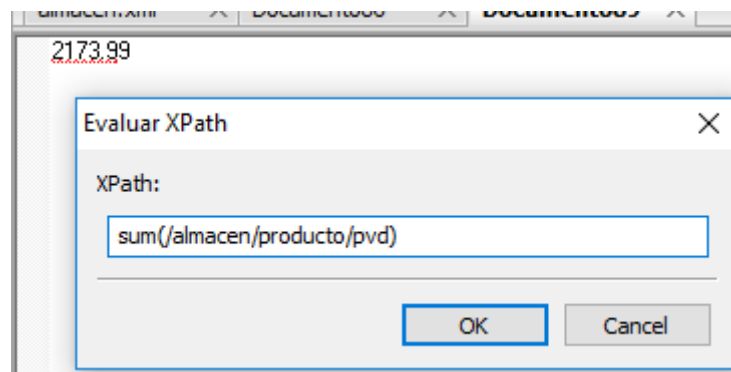
1. XPATH
2. XSLT

# XPATH

Vamos a mostrar los nombres de todos los artículos:

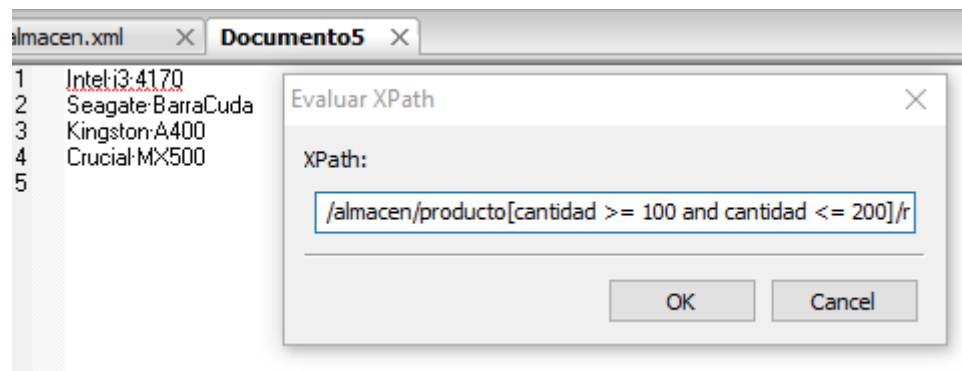


Ahora mostraremos el precio total de los productos:

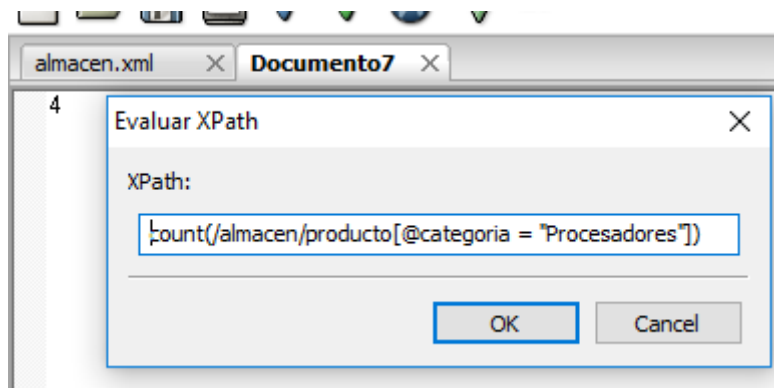


A continuación mostraremos los productos que tienen una cantidad entre 100 y 200

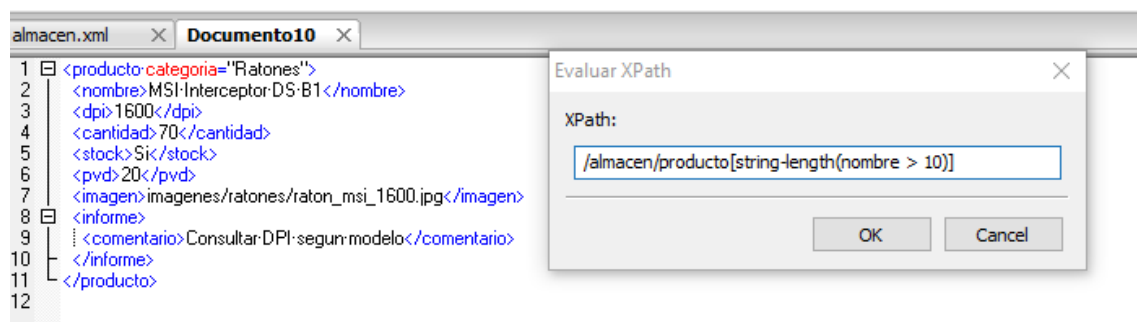
`/almacen/producto[cantidad >= 100 and cantidad <= 200]/nombre/text()`



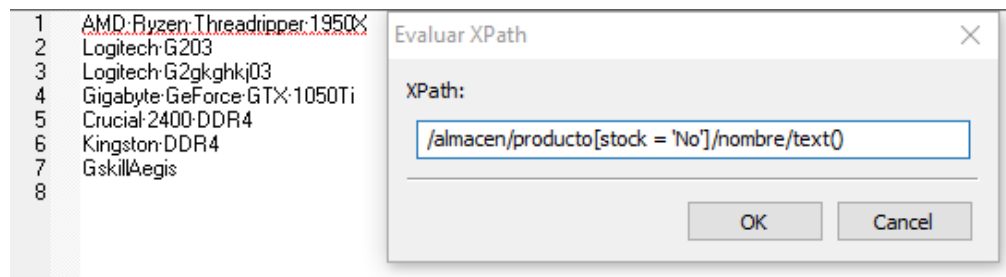
Contaremos la cantidad de nodos que tienen la categoría "Procesador"



Sacamos todos los datos de los productos con un nombre mayor a 10 caracteres

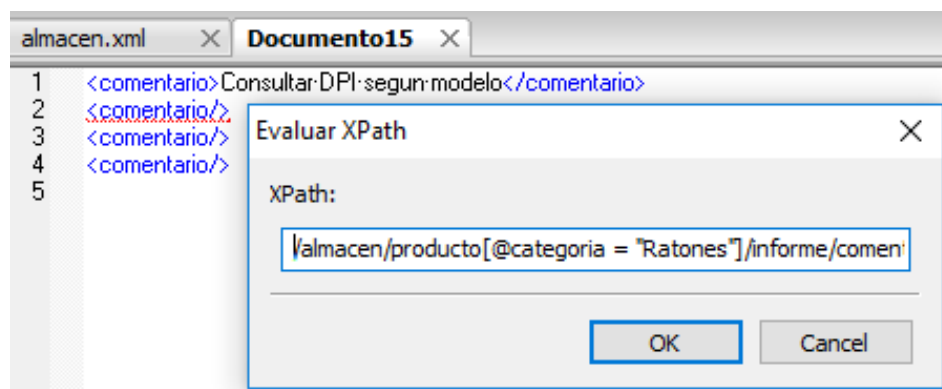


La siguiente consulta que haremos será la de saber el nombre de los productos que no tienen stock.



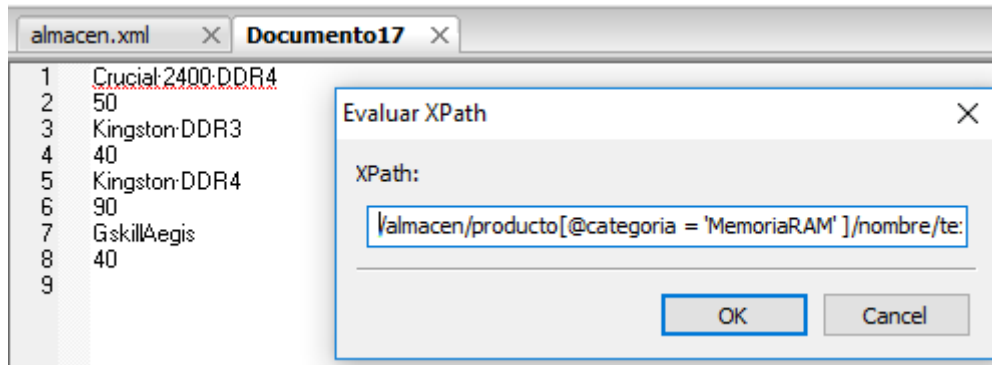
Vamos a obtener los comentarios de la categoría "Ratones"

`/almacen/producto[@categoria = "Ratones"]/informe/comentario`



Por último hemos probado a concatenar una consulta, donde hemos obtenido el nombre y el precio del producto.

```
/almacen/producto[@categoria = 'MemoriaRAM' ]/nombre/text() |  
/almacen/producto[@categoria = 'MemoriaRAM' ]/pvd/text()
```



# XSLT

El XSLT que he diseñado ha sido orientado a una especie de web de consulta del “almacén”

En el primer XSLT se representan todos los productos del XML mediante un “for”

```
<section>
  <h2 class="resultados"> Todos los Productos </h2>
<div class="tabla">
  <table>
    <tr>
      <xsl:for-each select="/almacen/producto[@categoria='Procesadores']">
        <td>
          <img>
            <xsl:attribute name="src">
              <xsl:value-of select="imagen"/>
            </xsl:attribute>
          </img>
          <p><xsl:value-of select="nombre"/></p>

          <xsl:if test="stock='Si'">
            <p class="disponible">Disponible:
              <xsl:value-of select="stock"/>
            </p>
          </xsl:if>

          <xsl:if test="cantidad='0' and stock='No'">
            <p class="no_disponible">Disponible:
              <xsl:value-of select="stock"/>
            </p>
          </xsl:if>
        </td>
      </xsl:for-each>
    </tr>
  </table>
</div>
```

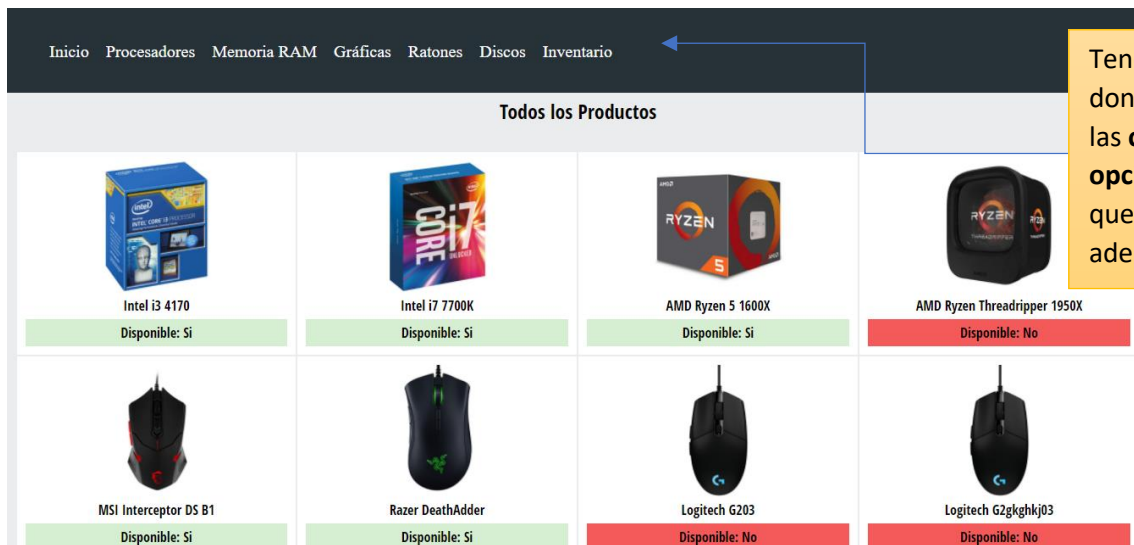
Si bien podemos observar hemos creado una tabla y hemos dicho que nos muestre todos los productos que tienen la **categoría “Procesadores”**, **mostrando una imagen**, en donde tenemos que utilizar un **atribute “src”** dentro de un **img** para que nos coja **ese elemento como una imagen en html “<img src.....>”**

También hemos seleccionado el valor **“nombre”**, además hemos incluido un **“xsl:if”** para que cuando **“stock tenga un valor = si”** el texto **“disponible”** será representado con una clase en concreto.

Lo mismo si **“cantidad = 0 y además stock = No”**, aplicamos otro estilo al texto **“disponible”** (ambos estilos aplican únicamente un color, verde o rojo).

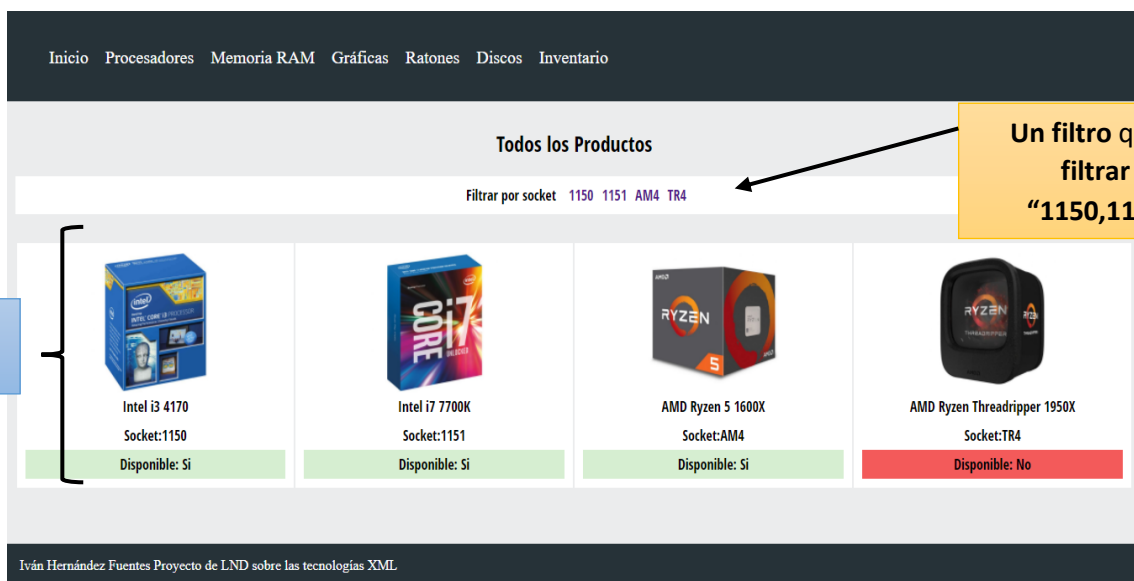
El **“for”** anterior lo hemos **repetido para todas las categorías**, en donde luego de aplicar diferentes estilos obtenemos el siguiente resultado como **“Página principal”(index.html)**

Al final obtenemos lo deseado una tabla que muestre **todos los productos**, el nombre, una imagen y un color según este disponible o no.



Tenemos un menú donde se muestran las **categorías** y una opción **"Inventario"** que veremos más adelante.

En la segunda opción del menú (**Procesadores**) nos muestra la siguiente página:



La sección de procesadores nos muestra **"todos los procesadores existentes"**, cosa que básicamente hicimos anteriormente, un **"for"** donde nos muestre **todos los productos de la categoría "procesadores"** (Véase figura 1)

```

</section>
<h2 class="resultados">Todos los Productos</h2>
<div class="filtro">
  <p>Filtrar por socket</p>
  <a href="procesador_socket_1150.html">1150</a>
  <a href="procesador_socket_1151.html">1151</a>
  <a href="procesador_socket_AM4.html">AM4</a>
  <a href="procesador_socket_TR4.html">TR4</a>
</div>
<div class="tabla">
  <table>
    <tr>
      <xsl:for-each select="/almacen/producto[@categoria='Procesadores']">
        <td>
          <img>
            <xsl:attribute name="src">
              <xsl:value-of select="imagen"/>
            </xsl:attribute>
          </img>
          <p><xsl:value-of select="nombre"/></p>
          <p>Socket:<xsl:value-of select="socket"/></p>
          <xsl:if test="stock='Si'">
            <p class="disponible">Disponible:
              <xsl:value-of select="stock"/>
            </p>
          </xsl:if>
          <xsl:if test="cantidad='0' and stock='No'">
            <p class="no_disponible">Disponible:
              <xsl:value-of select="stock"/>
            </p>
          </xsl:if>
        </td>
      </xsl:for-each>
    </tr>
  </table>
</div>
</section>

```

Filtro que veremos a continuación

**Figura 1:** utilizamos un “For each” para seleccionar todos los productos de la categoría “Procesadores”.

En la sección procesadores tenemos un filtro que nos filtraba por socket, cosa que hemos hecho con el siguiente XSLT.

```

<table>
<tr>
  <xsl:for-each select="/almacen/producto[socket='1150']">
    <td>
      <xsl:if test="stock='Si'">
        <p class="disponible">Disponible:
          <xsl:value-of select="stock"/> &#160;Cantidad: <xsl:value-of select="cantidad"/>
        </p>
      </xsl:if>
      <img>
        <xsl:attribute name="src">
          <xsl:value-of select="imagen"/>
        </xsl:attribute>
      </img>
      <p><xsl:value-of select="nombre"/></p>
      <p>Socket:<xsl:value-of select="socket"/></p>
    </td>
  </xsl:for-each>
</tr>
</table>

```

Filtro



Hemos indicado en el **for each** que **seleccione el socket “1150”**, seguidamente mostrará lo siguiente(*Véase figura 2*)



**Figura 2:** Nos muestra todos los procesadores, con el **tipo de socket “1150”** en este caso solo tenemos uno.

**Como novedad mostramos elementos nuevos de nuestro XML:**

Disponible: Si Cantidad: 100

Cantidad existente



Intel i3 4170

Socket:1150

Precio (impuestos no incluidos)100.99 €

Precio + IVA 122 €

Precio + IGIC 108 €

Comentario:

El precio sin impuestos,  
con IVA e IGIC (*calculados en  
el xslt véase figura 3*)

Además del **Comentario**

**Figura 3:** La segunda parte de nuestro xslt anterior (*figura 2*) es la siguiente:

```
<xsl:if test="cantidad='0' and stock='No'">
  <p class="no_disponible">Disponible:
    <xsl:value-of select="stock"/><p>Cantidad: <xsl:value-of select="cantidad"/></p>
  </p>
</xsl:if>
<p>Precio (impuestos no incluidos) <xsl:value-of select="pvd"/> €</p>
<p>Precio + IVA: <xsl:value-of select="round(pvd*21 div 100 + pvd)"/> €</p>
<p>Precio + IGIC: <xsl:value-of select="round(pvd*7 div 100 + pvd)"/> €</p>
<p>Comentario: <xsl:value-of select="informe"/></p>
</td>
</xsl:for-each>
</tr>
</table>
```

Aquí mostramos la cantidad del producto, su **Precio sin impuestos, y luego con impuestos**, donde básicamente hemos **multiplicado y dividido**, además **hemos trabajado sobre la función round()** para redondear el precio.

Por último nos muestra el comentario alojado en el elemento informe.

La estructura general de nuestro XSLT es la explicada anteriormente, para el resto de categorías se ha filtrado según sus características.




- Para la **memoria RAM, el tamaño.**

Todos los Productos			
Filtrar por Memoria 4GB 8GB			
 <p>Crucial 2400 DDR4 Memoria:8GB</p> <p>Disponible: No</p>	 <p>Kingston DDR3 Memoria:8GB</p> <p>Disponible: Si</p>	 <p>Kingston DDR4 Memoria:4GB</p> <p>Disponible: No</p>	 <p>Gskill Aegis Memoria:4GB</p> <p>Disponible: No</p>

- Para las **Gráficas el tipo de memoria.**

Todos los Productos			
Filtrar por Tipo de Memoria GDDR5 DDR3 GDDR3			
 <p>Gigabyte GeForce GTX 1050Ti Tipo:GDDR5</p> <p>Disponible: No</p>	 <p>Gigabyte GeForce GT 710 Tipo:GDDR3</p> <p>Disponible: Si</p>	 <p>Asus Radeon R5 230 Tipo:DDR3</p> <p>Disponible: Si</p>	 <p>MSI GeForce GT710 Tipo:GDDR3</p> <p>Disponible: Si</p>

- Para los **Ratones según el DPI**

Todos los Productos			
Filtrar por DPI 1600 16000 6000			
			
MSI Interceptor DS B1 DPI:1600	Razer DeathAdder DPI:16000	Logitech G203 DPI:6000	Logitech G2gkghkj03 DPI:6000
Disponible: Si	Disponible: Si	Disponible: No	Disponible: No

- Para los **discos duros el tipo según sea HDD O SSD**

Todos los Productos			
Filtrar por tipo de Disco SSD HDD			
			
Seagate BarraCuda TIPO:HDD	Kingston A400 TIPO:SSD	Samsung 860 TIPO:SSD	Crucial MX500 TIPO:SSD
Disponible: Si	Disponible: Si	Disponible: Si	Disponible: Si

**\*Si hay dos productos que pasan el filtro se representaría de la siguiente manera:**

Todos los Productos	
Disponible: Si Cantidad: 90	Disponible: Si Cantidad: 11
	
Gigabyte GeForce GT 710 Tipo:GDDR3	MSI GeForce GT710 Tipo:GDDR3
Precio (impuestos no incluidos)40 €	Precio (impuestos no incluidos)59 €
Precio + IVA 48 €	Precio + IVA 71 €
Precio + IGIC 43 €	Precio + IGIC 63 €
Comentario:	Comentario:

Para finalizar la ultima sección que tenemos en el menú **“Inventario”**, mostramos una lista general de todos los productos con su cantidad.

```
<!-- V12 class= lista de productos V12 -->
<section>
<div class="lista">
<ol>
<xsl:for-each select="almacen/producto">
<xsl:sort select="nombre" order="ascending"/>
<li><xsl:value-of select="nombre"/>:-----Cantidad:<xsl:value-of select="cantidad"/></li>
</xsl:for-each>
</ol>
</div>
</section>
<footer>
<p>Iván Hernández Fuentes Proyecto de LND sobre las tecnologías XML</p>
</footer>
</div>
</body>
</html>
```

Hemos utilizado un **for each** que seleccione el nodo **“almacén/producto”**, y luego un **“sort”** para ordenar **“nombre”** de manera **ascendente**, por último en un **“ol”, “li”** mostramos el **“nombre”** y la **“Cantidad”**.

**En el HTML aplicando estilos se vería de la siguiente manera:**

## Inventario

AMD Ryzen 5 1600X:	Cantidad:10
AMD Ryzen Threadripper 1950X:	Cantidad:0
Asus Radeon R5 230:	Cantidad:11
Crucial 2400 DDR4:	Cantidad:0
Crucial MX500:	Cantidad:110
Gigabyte GeForce GT 710:	Cantidad:90
Gigabyte GeForce GTX 1050Ti:	Cantidad:0
GskillAegis:	Cantidad:0
Intel i3 4170:	Cantidad:100
Intel i7 7700K:	Cantidad:75
Kingston A400:	Cantidad:110