

<b>Nombre función:</b>	<b>GetSaldoFisicoProductoEnBodegas (datajson, controlkey, iapp, random) : json</b>				
<b>Resultado:</b>	Retorna la existencia física del producto en cada bodega donde tiene saldo.				
<b>Programa o servidor:</b>	Agente CP.	<b>Seguridad:</b>	No aplica		
<b>Descripción de la función:</b>	Retorna un json con la lista de bodegas y cantidad del elemento, donde éste tiene saldo físico (Saldo contable + Recepciones - Remisiones).				
<b>Desarrollado por:</b>	ESuarez	<b>Documentado por:</b>	ESuarez	<b>Aprobado por:</b>	PAizate
<b>Fecha creación:</b>	15/1/2015	<b>Fecha modificación:</b>	26/05/15	<b>Versión:</b>	1.0

<b>Descripción de parámetros que recibe la función</b>			
<b>Nombre parám.</b>	<b>Tipo</b>	<b>Descripción</b>	<b>Ejemplo</b>
datajson	Json	Este Json contiene en su interior la siguiente estructura:  <b>irecurso:</b> Código del elemento de inventario al que se le quiere hacer la consulta de saldos por bodega. ( <i>requerido</i> ).  <b>iiinventario:</b> Código de la bodega. ( <i>opcional</i> ). - Si es dado iiinventario, se retorna el saldo del irecurso para la bodega especificada. - Si no es dado iiinventario, se retorna el saldo del irecurso de todas las bodegas.	{ "irecurso":"102030" "iiinventario":"010" }
controlkey	Varchar	Corresponde al keyagente obtenido en el logueo. ( <i>requerido</i> )	"366989185577850"
iapp	Varchar	Código que identifica a la aplicación que interactúa con el Agente. ( <i>requerido</i> )	"1985"
random	Varchar	Cadena aleatoria que se crea en el lado del cliente, esto con el fin de que las peticiones no sean cacheadas por el navegador Internet Explorer (para aplicaciones web). ( <i>Opcional</i> ).	"6935968966323469"

<b>Ejemplo de la URL del llamado de la función</b>
<a "="" 1985="" 366989185577850="" 6935968966323469="" href="http://190.248.152.10:8011/datasnap/rest/TInventarios/GetSaldoFisicoProductoEnBodegas/{" irecurso":"102030"}="">http://190.248.152.10:8011/datasnap/rest/TInventarios/GetSaldoFisicoProductoEnBodegas/{"irecurso":"102030"}/366989185577850/1985/6935968966323469/</a>

### Formato del Json que retorna la función en caso de ser exitosa

```
{
  "result": [
    {
      "encabezado": {
        "resultado": "true", "imensaje": "", "mensaje": "", "tiempo": "49"
      },
      "respuesta": {
        "datos": [
          {"iinventario": "1", "ninventario": "Bodega 1", "qproducto": "39"},
          {"iinventario": "2", "ninventario": "Bodega 2", "qproducto": "25"}
        ]
      }
    }
  ]
}
```

### Descripción del Json que retorna la función en caso de ser exitosa

Nombre parámetro	Tipo	Descripción
encabezado	Json	<p>Json que contiene en su interior los siguientes datos:</p> <p><b>resultado (varchar):</b> Retorna true siempre que la petición se ejecute satisfactoriamente.</p> <p><b>imensaje (varchar):</b> Código del mensaje de eventualidad o error en caso de presentarse.</p> <p><b>mensaje (varchar):</b> Mensaje de eventualidad o error en caso de presentarse.</p> <p><b>tiempo (varchar):</b> Tiempo que se tardó el Agente en resolver la petición, este tiempo está dado en milisegundos.</p>
respuesta	Json	<p>Json que contiene en su interior el arreglo de objetos "datos" que será descrito a continuación:</p> <p><b>datos (arreglo de objetos):</b> Cada objeto está conformado por:</p> <ul style="list-style-type: none"> <li>- <b>iinventario:</b> código de la bodega.</li> <li>- <b>ninventario:</b> nombre o descripción de la bodega.</li> <li>- <b>qproducto:</b> saldo del producto en la bodega.</li> </ul>

### Descripción de resultados en caso de eventualidades o errores

<b>Descripción:</b>	<p>Para esta función se pueden presentar las siguientes eventualidades o errores, un ejemplo del Json que retorna la función cuando se genera una eventualidad es el siguiente.</p> <pre>{   "result":[{     "encabezado":{"resultado":"false","imensaje":"40","mensaje":"Usuario no logueado."},     "respuesta":{"datos":""}   }] }</pre>
<b>Eventualidades</b>	
<b>Generales:</b>	<p>0: Error en la aplicación, errores no controlados.</p> <p>1: Mensaje que le indica al usuario que debe corregir errores (errores controlados).</p> <p>10: No se ingresó un Json como parámetro.</p> <p>40: Usuario no logueado.</p>

