

Anexo II: Atributos

Ejemplo	Significado
Nombre Entidad.Nombre Atributo	Atributo de Entidad
Nombre Entidad.Nombre Atributo Referencia.Nombre Atributo	Atributo que se obtiene a través de otro atributo de categoría referencia. Ejemplo: Factura.Cliente minorista.Nombre
Nombre Entidad.Nombre Atributo Referencia Inversa.Nombre Atributo en la entidad referenciada inversamente	Cliente.Cliente comprador en Factura.Nro de Factura
Nombre Entidad.Nombre Atributo de la Colección Referenciada.Nombre Atributo	Atributo que se obtiene a través de otro atributo de categoría referencia a elemento de colección. Ejemplo: Factura.Dirección del cliente.Calle

Colección a texto enriquecido (Colección)

Permite convertir un conjunto de ítems de una colección en una lista de elementos de tipo Texto Enriquecido Internacionalizable.

En esta función podemos utilizar como parámetro de entrada tanto un atributo de tipo Colección como una expresión de búsqueda “En <Colección> Buscar <Atributo>”, siempre y cuando luego de la instrucción “En” se indique una Colección, por ejemplo, si se desea obtener la lista de direcciones de un cliente, éstas pueden obtenerse del siguiente modo En .Domicilios Buscar Dirección.

Adicionalmente, la función admite que se pueda solicitar más de un elemento en la expresión de búsqueda, para esto, al momento de escribir la expresión se deberán indicar los elementos de la colección que se desean convertir separándolos con ‘;’, por ejemplo; En .Domicilios Buscar Dirección; Código postal.

Sintaxis válidas de la función:

```
Colección a Texto Enriquecido(<Colección>)
Colección a Texto Enriquecido(En <Colección> Buscar <Atributo>)
Colección a Texto Enriquecido(En <Colección> Buscar <Atributo 1>;<Atributo 2>;<Atributo 3>)
```

Comportamientos:

- ✓ Si el parámetro de la función es un atributo de tipo Colección, se convertirán todos los atributos de la misma a Texto Enriquecido. Por otra parte, si el parámetro es una expresión de búsqueda se convertirán solamente los elementos indicados en la misma.
- ✓ En los casos en los que los atributos a presentar sean de tipo numérico, la función contemplará los formatos de cantidad de decimales expresados en el formato visual del atributo.
- ✓ Si el valor de un atributo que se está convirtiendo es nulo, no se presenta un error y concatena el separador de elemento.

1.1 Funciones de consumo de servicios sincrónicos

Las funciones que consumen dispositivos o servicios web realizan el consumo de forma sincrónica. Esto implica que, a diferencia de los consumos que tienen como origen la acción de una Regla de negocio, el mismo se efectúa **por cada unidad de cambio**.

Dichos parámetros se obtienen de los siguientes lugares:

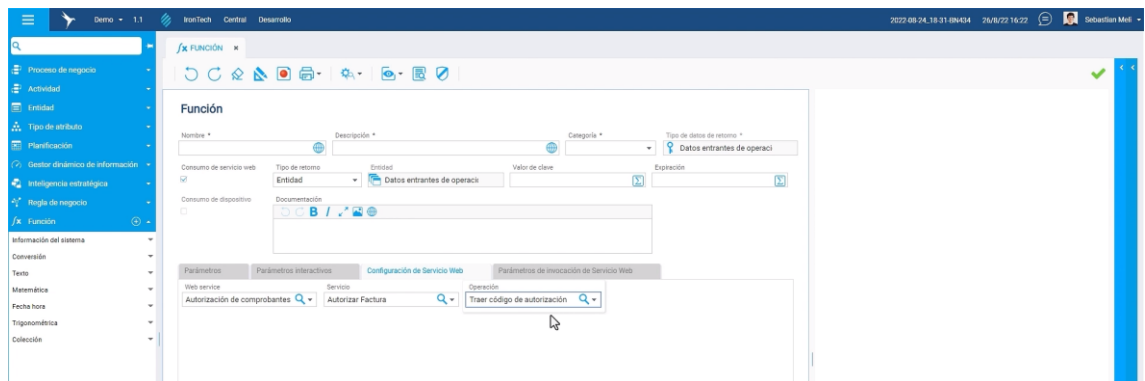
1.1.1 Consumo de operación de Servicio web.

Se obtienen los parámetros de entrada de la operación seleccionada y el sistema, además, debe crearlos con los valores predeterminados que fueron definidos en la misma.

- **Mediante agente:** Permite que el consumo del servicio web se realice a través del agente para que pueda comunicarse con el dispositivo físico final. Al activar esta opción, el sistema adicionalmente presenta las siguientes opciones, las cuales se detalla su funcionamiento en el siguiente ítem *Consumo de dispositivo*:
 - **Tipo de dispositivo.** Tener en cuenta que solo se presentarán los tipos de dispositivos los cuales tengan activa la propiedad Consumo por web service.
 - ✓ **Origen de dispositivo.**
 - **Dispositivo fijo.**
 - Dispositivo
 - **Dispositivo Predeterminado**
 - Predeterminado del usuario autenticado
 - Predeterminado del Sitio
 - Predeterminado del dominio
 - **Dispositivo mediante Expresión**
 - Expresión

Para más información acerca del consumo de web service mediante dispositivos, ver [especificación funcional de dispositivos](#).

Supongamos, ahora, que debemos cargar una factura a un cliente y para ello necesitamos el código que nos provee AFIP. Existen distintas formas de conseguir este valor, pero nosotros lo haremos mediante una función que consuma un web service. Es importante mencionar que este debe estar previamente creado.



En la segunda, visualizamos los parámetros de invocación del servicio web, los cuales tienen como valores iniciales aquellos definidos en la operación del servicio, que pueden redefinirse, sin problemas.

1.2 Manejo de nulos

Es importante destacar que el comportamiento es diferente cuando una expresión es nula (es decir que no se configuró la misma) y cuando el resultado de la evaluación es nulo.

Por ejemplo, cuando una Regla de negocio se ejecute y una condición no se pueda evaluar porque alguno de los atributos a evaluar dentro de la expresión es nulo (no posee valor), las acciones dentro de esta condición no se ejecutarán.

De todas maneras, en las condiciones se puede especificar que se evalúe si el atributo no es nulo, por ejemplo: si se desea ejecutar una acción si “mes(fecha) = 12” y en algunas instancias la fecha es nula, se puede cambiar la condición a “no Esnulo(fecha) y mes(fecha) = 12”.

En todas las operaciones sobre colecciones, si alguno de los elementos que aplican a la operación es nulo, se ignora. Por ejemplo, si monto suma todos los subtotales de los ítems de factura, si existe un ítem cuyo subtotal está vacío, se ignora para la sumatoria. Esto no se deberá confundir con los valores 0.

Supongamos el siguiente ejemplo: Se tiene una colección de valores enteros y se obtiene: Suma y Cuenta de los mismos. Veamos 2 ejemplos que a simple vista parecen similares, pero varían en sus resultados.

Colección 1:

Valor entero
20
40
(nulo)
60

Suma: 120

Cuenta: 3

Colección 2:

Valor entero
20
40
0
60

Suma: 120

Cuenta: 4

En la suma, ambos valores son iguales y para la cuenta de ítems, en el caso de la colección 1, el valor nulo se omite y en el caso de la colección 2 el valor 0 se tiene en cuenta.

Otro caso a considerar es en los atributos de tipo booleano, si no tienen valor inicial toman valor nulo, sin embargo, al visualizarlos se ven de la misma manera que el falso, por lo tanto es sumamente importante validar que no sea nulo antes de utilizarlo, o darle un valor inicial y de esta manera evitar que tome el valor nulo.

Las siguientes son las reglas al momento que comparar valores nulos. Es importante considerarlas ya que podrían resolverse algunas expresiones y puede que no necesariamente el modelador tenga que siempre completar cada uno de los atributos.

Y	Verdadero	Nulo	Falso
Verdadero	Verdadero	Nulo	Falso
Nulo	Nulo	Nulo	Falso
Falso	Falso	Falso	Falso

O	Verdadero	Nulo	Falso
Verdadero	Verdadero	Verdadero	Verdadero
Nulo	Verdadero	Nulo	Nulo
Falso	Verdadero	Nulo	Falso

NO	
Verdadero	Falso
Nulo	Nulo
Falso	Verdadero

Esnulo()	
Verdadero	Falso
Nulo	Verdadero
Falso	Falso

No Esnulo()	
Verdadero	Verdadero
Nulo	Falso
Falso	Verdadero

Conversión de tipos de atributos

Destino Origen	Simple	Compuesto	Referencia	Referencia a colección	Colección	Enumerado	Secuencia
Simple	No	Si	Si	Si	Si	Si	No
Compuesto	No	Si	Si	Si	Si	Si	No
Referencia	Si	Si	Si	Si	Si	Si	No
Referencia a Colección	Si	Si	Si	Si	Si	Si	No
Colección	Si	No	No	No	No	Si	No
Enumerado	Si	Si	Si	Si	Si	Si	No
Secuencia	No	No	No	No	No	No	No

Detalle de funciones:

Función	Descripción	Ansi-92
Convertir Referencia a Simple (tipo de atributo)	Convierte un tipo de atributo de categoría Referencia a un tipo atributo de categoría Simple. Si se aplica a un atributo que contiene un valor, se copia el atributo descriptor de la referencia en el atributo simple.	Ejemplo: Convertir Referencia a Simple (Cliente; Nombre de cliente)
Convertir Referencia a Colección a Simple (tipo de atributo)	Convierte un tipo de atributo de categoría Referencia a Colección a un tipo atributo de categoría Simple. Si se aplica a un atributo que contiene un valor, se copia el atributo descriptor de la colección referenciada en el atributo simple.	Ejemplo: Convertir Referencia a Simple (Direcciones de Cliente; Dirección del Cliente)

Consideraciones para atributos basados en tipo fecha

En particular para las operaciones entre los diferentes tipos de Fecha (Solo Fecha, Solo Hora, Fecha y Hora, Instante, Duración) para la suma:

	Instante	Fecha Y Hora	Solo Fecha	Solo Hora	Duración
Instante	No permitido	No permitido	No permitido	No permitido	OK
Fecha Y Hora		No permitido	No permitido	No permitido	OK
Solo Fecha			No permitido	No Permitido	OK
Solo Hora				No Permitido	No Permitido
Duración					OK

Para el caso de la resta se tendrán las siguientes operaciones permitidas:

	Instante	Fecha Y Hora	Solo Fecha	Solo Hora	Duración
Instante	OK (duración)	OK (duración)	OK (duración)	No permitido	OK (instante)

Fecha y Hora	Y	OK (duración)	OK (duración)	OK (duración)	No permitido	OK (fecha y hora)
Solo Fecha		OK (duración)	OK (duración)	OK (duración)	No Permitido	OK (sólo fecha)
Solo Hora		No Permitido	No Permitido	No Permitido	No Permitido	No Permitido
Duración		No Permitido	No Permitido	No Permitido	No Permitido	OK (duración)

De las tablas anteriores se deberá tener en cuenta las siguientes consideraciones:

- ✓ Para el caso de la suma o resta de Instante, Fecha y Hora, Solo Fecha con una Duración se deberá validar que la Duración tenga una unidad compatible con la operación. Es decir la Duración deberá tener una unidad mayor que el tipo con el que se hace la operación (Entendiendo que la unidad mayor es el año y la unidad menor es el nanosegundo). Por ejemplo, a un Instante se le puede sumar o restar una Duración con cualquier unidad, pero a un Fecha y Hora no se le podrá sumar o restar una Duración que está en milisegundos, o a un Solo Fecha no se le podrá sumar o restar una Duración que está en Horas.
- ✓ Para el caso de operaciones de suma o resta entre Duraciones, las mismas podrán no tener la misma unidad de medida, siempre que ambas duraciones se encuentren en uno de los siguientes grupos y el resultado es siempre la unidad de medida menor:
 - *Grupo 1: Semana o inferior:* las unidad de tiempo posibles son semana, día, horas, minutos o segundos. Ejemplo válido: {2 días} - {4 horas} devuelve {44 horas}. Ejemplo inválido: {4 meses} + {2 días} no se puede realizar porque no se sabe cuántos días tiene cada mes. Lo mismo ocurre con la cantidad de días de un año.
 - *Grupo 2: Mes o superior:* las unidad de tiempo posibles son mes o año. Ejemplo válido: {1 año} + {4 meses} devuelve {16 meses}. Ejemplo inválido: {2 años} – {14 minutos} no se puede realizar por el mismo motivo (la cantidad de días varía de mes a mes y de año a año).

✓

Contenedor: Numérico

Tipo de dato	Numérico	Texto	Booleano	Fechas	Duración
Numérico	2	3	3	7	7
Texto		3	3	3	3
Booleano			3	3	3
Fechas				7	7
Duración					7

Contenedor: Texto o Texto Enriquecido

Tipo de dato	Numérico	Texto	Booleano	Fechas	Duración
Numérico	2 y 5	4	4	6 y 5	6 y 5
Texto		8	4	4	4
Booleano			5	4	4
Fechas				5	5
Duración					5

Contenedor: Booleano

Tipo de dato	Númérico	Texto	Booleano	Fechas	Duración
Númérico	3	3	3	3	3
Texto		3	3	3	3
Booleano			1	3	3
Fechas				3	3
Duración					3

Contenedor: Fechas

Tipo de dato	Númérico	Texto	Booleano	Fechas	Duración
Númérico	3	3	3	6	3
Texto		7	3	7	7
Booleano			3	3	3
Fechas				3	1
Duración					3

Contenedor: Duración

Tipo de dato	Númérico	Texto	Booleano	Fechas	Duración
Númérico	6	3	3	3	6
Texto		7	3	3	7
Booleano			3	3	3
Fechas				1 para la resta 3 para la suma	3
Duración					1

Nota de arquitectura

Si en una expresión se repiten 2 o más veces la misma función o la misma operación que involucra datos externos al contexto de la misma, la primera vez que se evalúa, se guarda el resultado en memoria y las demás veces, se obtiene del valor guardado en memoria.

Esto puede ocurrir dentro de la misma expresión o en distintas expresiones evaluadas en la misma instancia.

Por ejemplo, si en la entidad Factura se tienen los siguientes valores iniciales:

Vendedor de la factura = Usuario().Nombre + Usuario().Apellido

Usuario que generó instancia = Usuario().Nombre

Luego de que evalúe el primer valor inicial, siempre se devolverá el mismo nombre, aunque en el medio de las 2 evaluaciones cambie el nombre del usuario actual.

Esto se realiza dentro de una operación de Core.