

## Variables colección

*GeneXus*<sup>™</sup>

Al finalizar el video anterior, hemos mencionado el concepto de variable colección.

## Colecciones:

Colección numérica

1	5	8	7	3
---	---	---	---	---

Colección de textos

Ann	Paul	Susan	Peter
-----	------	-------	-------

Colección de SDTs

1	2	3	4	5
Uruguay	France	United States	Italy	Argentina
Montevideo	Paris	Washington	Rome	Buenos Aires

En todo objeto GeneXus podemos definir variables basadas en tipos de datos simples o tipos de datos estructurados que nos permiten almacenar los datos de una entidad, por ejemplo, valores numéricos, textos o los datos de un país.

Colecciones:

&names →

Name	Type	Is Collection	Description
& Variables			
& Standard Variables			
names	Character(20)	<input checked="" type="checkbox"/>	names

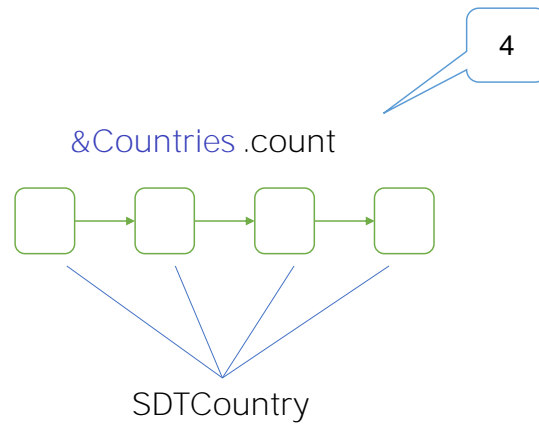
&numbers →

Name	Type	Is Collection	Description
& Variables			
& Standard Variables			
numbers	Numeric(4.0)	<input checked="" type="checkbox"/>	numbers

Hemos visto que al momento de definir una variable, disponemos del check box "IsCollection" para definir que esa variable es una colección.

Las variables colección, independientemente de que sean colecciones numéricas, de caracteres, de SDTs, ofrecerán métodos propios para el manejo de dicha colección.

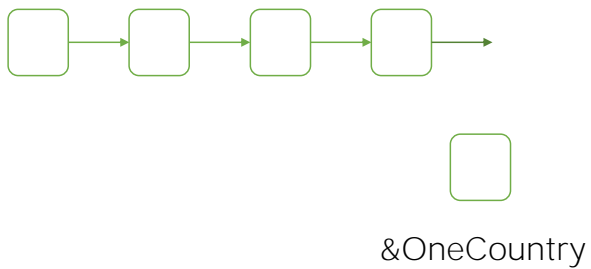
Para contar la cantidad de elementos:



El método `Count`, permite saber la cantidad de elementos que tiene la colección. Devuelve dicha cantidad, por lo que se trata de un método de solo lectura.

Para agregar un elemento a la colección:

```
&Countries .add(&OneCountry)
```



El método Add permite agregar un elemento a la colección. Por defecto dicho elemento se agrega al final de la colección, si bien es posible indicarle la posición.

Para agregar un elemento a la colección:

Ejemplo: Cargar una colección de países

```

For each Country
    &CountryItem = New()
    &CountryItem.Id = CountryId
    &CountryItem.Name = CountryName
    &Countries.Add(&CountryItem)
endfor

```

Estamos reiniciando la variable SDT, y por eso estamos utilizando el operador New

Name	Type	Is Collection
Variables		
Standard Variables		
Countries	SDTCountry	<input checked="" type="checkbox"/>
CountryItem	SDTCountry	<input type="checkbox"/>

Name	Type	Description
SDTCountry		SDTCountry
Id	Numeric(4.0)	Id
Name	Character(20)	Name

Por ejemplo, si queremos cargar la colección de países de la base de datos, ¿cómo hacemos? Si bien en próximo video veremos la forma sencilla de cargar utilizando un Data Provider, veamos ahora esta otra opción.

Cuando se declaran las variables, el programa ya les asigna un espacio de memoria, pero siempre podemos pedir otro espacio de memoria para las variables, para no utilizarlo dos veces.

Es decir, estamos reiniciando la variable SDT, y por eso estamos utilizando el operador New

En este caso se necesita, porque de lo contrario siempre estaremos sobrescribiendo ese espacio de memoria. Entonces si no pedimos nuevo espacio estaremos agregando siempre el mismo ítem a la colección, y tendremos una colección de n ítems, pero con los mismos valores. Es que la variable &countryItem no se reinicia luego de aplicado el Add, como podría pensarse.

Para eliminar un element de la colección:

```
&Countries .remove(3)
```



El método Remove permite eliminar un elemento de la colección a partir de su índice. El rango de los índices comienza en 1 y finaliza con el valor devuelto por el método Count.

Para eliminar todos los elementos de la colección:

```
&Countries .clear()
```



El método Clear permite eliminar todos los elementos de la colección,



Para ordenar los items de una colección:

```
&Countries .sort("Name")
```



Name	Type	Description
SDTCountry		SDTCountry
• Id	Numeric(4,0)	Id
• Name	Character(20)	Name

```
For &CountryItem in &Countries  
    Print Countries  
Endfor
```

El método Sort permite ordenar la colección. Si los elementos son de tipo SDT, entonces podemos indicarle que ordene la colección de acuerdo a uno de los miembros de esa estructura, por ejemplo, de acuerdo al nombre del país.

Luego, para recorrer una colección tenemos el comando For ítem in collection.

Este comando va recorriendo la colección y cargando cada ítem en la variable &countryItem. De esta forma podemos imprimirla en la salida.

Para ordenar los items de una colección:

Ejemplo: Cargar y ordenar una colección de países

```

For each Country
    &CountryItem = New()
    &CountryItem.Id = CountryId
    &CountryItem.Name = CountryName
    &Countries.Add(&CountryItem)
endfor

&Countries.Sort("Name")
For &CountryItem in &Countries
    print Countries
Endfor

```

Name	Type	Description
SDTCountry		SDTCountry
• Id	Numeric(4.0)	Id
• Name	Character(20)	Name

Name	Type	Is Collection
Variables		
• Standard Variables		
• Countries	SDTCountry	<input checked="" type="checkbox"/>
• CountryItem	SDTCountry	<input type="checkbox"/>

Veamos entonces cómo queda finalmente nuestro procedimiento:

*GeneXus*<sup>TM</sup>

[training.genexus.com](http://training.genexus.com)  
[wiki.genexus.com](http://wiki.genexus.com)