Unique Clause

GeneXus™

Let's review the concept and behavior of the Unique clause.

We know this clause allows you to indicate the attribute, or set of attributes, whose value should not be repeated in a query output.

Requirement...

List of categories registered in the attractions. Repeated categories should not be displayed.

Category
- CategoryId
- CategoryName

Attraction
- AttractionId
- AttractionName
- AttractionPhoto
- CategoryId
- CategoryName

```
Print Title
For each Attraction
    unique CategoryId
    print Category
-endfor
```

Title

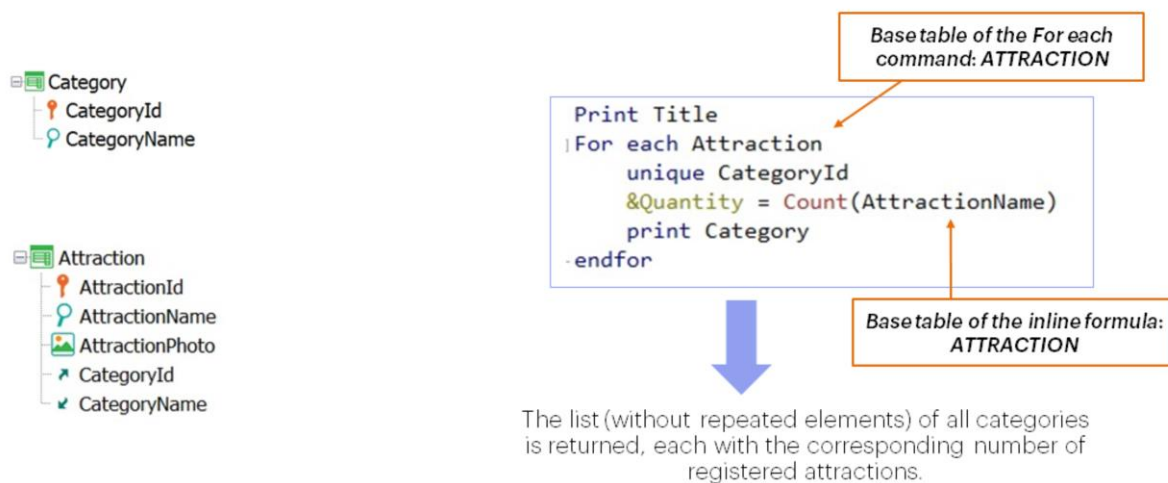**Categories in attractions**

Category

CategoryName

As a review:

A list showing all the categories registered in tourist attractions is requested, but no repeated elements should be displayed.
If we ran through the attractions to show the categories recorded in them, we would see repeated elements.

The Unique clause allows the list not to have these elements repeated.

Unique Clause

It also allows associating information with inline formulas.

```
Print Title
For each Attraction
     unique CategoryId
     &Quantity = Count(AttractionName)
     print Category
endfor
```

Base table of the For each command: ATTRACTION

Base table of the inline formula: ATTRACTION

The list (without repeated elements) of all categories is returned, each with the corresponding number of registered attractions.

Category
- CategoryId
- CategoryName

Attraction
- AttractionId
- AttractionName
- AttractionPhoto
- CategoryId
- CategoryName

The Unique clause also allows associating information with inline formulas.

In this example, the base table of the For each command is the same as the base table of the inline formula.

Therefore, the Count formula will add from the context an implicit condition in its evaluation: It will count all attractions for the attribute declared in the Unique clause – in this case, CategoryId.

In this way, this source will return the list without repetitions, of all the categories, each one with its corresponding number of registered tourist attractions.

Unique Clause in Data Provider

List per day the total amount of registered reservations.

Reservation
- ReservationId
- ReservationDate
- ReservationTotal

Trip
- TripId
- TripDescription
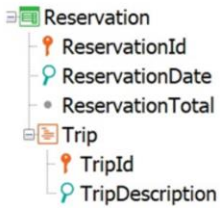
```
ReservationsByDay from Reservation unique ReservationDate
{
        ReservationsByDayItem
        {
                ReservationDate
                ReservationTotal = sum(ReservationTotal)
        }
}
```

```
ReservationsByDay
{
        ReservationsByDayItem from Reservation unique ReservationDate
        {
                ReservationDate
                ReservationTotal = sum(ReservationTotal)
        }
}
```

Now let's see an example of the use of the Unique clause in a Data Provider. For that, let's consider the Reservation transaction displayed. A Data Provider is needed that returns, per day, the total amounts for tour reservations.

We do not want to see the date repeated, so we declare the corresponding Unique clause followed by the ReservationDate attribute. In this way, the Data Provider will return the set of dates, without repeated elements, each one with the accumulated total.

It should be noted that in a Data Provider that has repetitive groups that access the database to retrieve values, declaring the base transaction, and Where, Order, Unique, etc. clauses, at the level of the collection name – i.e. the external group –, is exactly the same as declaring them at the level of the collection item name –i.e. the internal group.

Unique Clause in Data Provider: Navigation List

If we look at the navigation list of this Data Provider, we see that GeneXus adds the corresponding Unique clause with the attribute to be considered.

Unique Clause in the grid of a Web Panel

We have seen the use of the Unique clause in For each commands and in Data Providers. Let's see now that it is also possible to use it in the grid of a Web panel.

So let's consider again the Categories and Tourist Attractions. We are going to design a web panel that shows all the categories that have registered tourist attractions.

In the Web Panel grid we add the CategoryName attribute.

Now let's think about the same thing we did in the example with For each. The base table of this grid must be Attraction; therefore, we indicate it in the Base Transaction property of the grid. But since we see that the grid also has the Unique property, we indicate CategoryId so that repeated categories are not shown.

If we look at the corresponding navigation list, we see that it runs through ATTRACTION and sets CategoryId as unique value.

Complete syntax of the For each command

For Each BaseTransaction

    skip **expression1** count **expression2**
    order **att1, att2, … , attn** [when **condition**]
    order **att1, att2, … , attn** [when **condition**]
    unique att1, att2, … , attn
    using **DataSelector(parm1, parm2, … , parmn)**
    where **condition** [when **condition**]
    where **condition** [when **condition**]
    where **att** IN **DataSelector(parm1, parm2, … , parmn)**
    blocking **N**

        main_code
  When duplicate
        when_duplicate_code
  When none
        when_none_code

Endfor

Now let us look at the **complete** syntax of the For each command, although we should mention that we have not studied all the clauses we are looking at.

It should be noted that the Blocking clause, the updating of attributes in the main code, as well as the When duplicate clause are only valid when the For each command is in the Source of a procedure, because only in those objects it is possible to update the database directly.

The attributes that appear inside the When duplicate clause, as well as those that appear inside the When none clause, will not be taken into account when determining the base table of the For each command.

As for the attributes declared in the Unique clause – including formula attributes – will be processed together; that is, only once in the output. Therefore, if two or more records have the same value for that set of attributes, only one of them will be taken into account.

All the attributes declared in the places mentioned must belong to the extended table of the For each command base table.

Unique Clause: Restrictions

- Expressions cannot be used in the list of attributes of the Unique clause.

    For example: Unique
    ReservationDate.Year()

- Both in the body of the For each command, and in the groups of a Data Provider (outside inline formulas), only attributes that have unique values regarding those declared in the Unique clause can be included.

- Nested For each commands cannot be declared. The Unique clause cannot be used to implement a control break.

```
error spc0211: Unique clause in break group not supported
Failed: Specification
```

Finally, let's look at the restrictions on the use of the Unique clause:

- It is not possible to use expressions in the list of attributes declared in the Unique clause.
- Both in the body of the For each command, and in the groups of a Data Provider – outside inline formulas –, only attributes that have unique values regarding those declared in the Unique clause can be included.
- Another very important restriction to take into account is that the Unique clause cannot be declared in nested For each commands. So far, it cannot be used to implement a control break. If it is done, GeneXus will display a message like the one shown here:

training.genexus.com
wiki.genexus.com
training.genexus.com/certifications