

Description instead of codes

GeneXus™

Applications

- Need for a friendly user interface (UI)

Simple and fast data entry



Um dos desafios que enfrentam os desenvolvedores de aplicações atualmente é desenvolver uma interface de usuário amigável.

Entre outras coisas, uma interface de usuário amigável implica que a entrada de dados seja o mais simples e rápida possível. GeneXus tem propriedades configuráveis para simplificar a entrada de dados e ajudar a desenhar interfaces mais atrativas. Uma delas é a propriedade `InputType`.

Sua função é substituir o que o usuário necessita registrar em um atributo com um valor mais fácil de usar.

The image displays a GeneXus application interface. On the left, a 'Structure' window shows the data model for an 'Airport' entity, listing fields: AirportId, AirportName, CountryId, CountryName, CityId, and CityName. On the right, the 'Airport' form is shown with the following fields: Id (text input with value 0), Name (text input), Country Id (text input with value 0 and a search icon), Country Name (text input), City Id (text input with value 0 and a search icon), and City Name (text input). The form has a 'SELECT' button at the top and 'CONFIRM' and 'CANCEL' buttons at the bottom. The 'Country Id' and 'City Id' fields are highlighted with orange boxes.

Por exemplo, suponhamos que estamos em uma aplicação de uma agência de viagens, onde temos uma transação que inserimos os aeroportos e cada um deles tem um país e uma cidade.

Se declaramos a transação como está definida, por padrão o formulário será este, onde teremos que inserir um código para o país e outro código para a cidade.

A inserção de um dado a partir de um código não é a melhor experiência para um usuário, embora GeneXus forneça automaticamente um formulário prompt onde podemos pesquisar por nome, poderíamos deixar que o usuário insira o nome do país e/ou cidade que necessita e que o programa busque e atribua automaticamente o código correspondente.

Control Info	
Control Type	Edit
Input Type	Descriptions
Suggest	No
Item Values	CountryId
Item Descriptions	CountryName
Conditions	
Instantiated Attribute	
Notify Context Char	false

Airport

« < > » SELECT

Id

Name

Country Name

City Id ↑

City Name

CONFIRM
CANCEL

Conseguimos isto alterando o valor da propriedade InputType do atributo para o valor descriptions, desde que o control type seja edit.

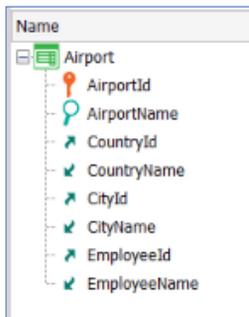
Vamos fazer isso com CountryId, mudamos de Values para Descriptions, e nos possibilita novas propriedades. A que temos que completar obrigatoriamente é Item descriptions.

Como o que queremos é inserir o nome em vez da chave, na propriedade Item descriptions selecionamos CountryName.

Em execução, parece ter desaparecido CountryId e permanece apenas CountryName em modo editável, mas na realidade o atributo que mostra esses valores não é CountryName e sim CountryId “disfarçado”.

Se, por exemplo, queremos selecionar o país Brasil, escrevemos e internamente é realizada uma busca na tabela Country para recuperar o código correspondente, que neste caso será “1”, mas para o usuário isto será transparente. Se formos ao Web Layout da transação, veremos que o atributo que temos efetivamente é CountryId e não CountryName.

É importante que o atributo que escolhemos para a propriedade Item Descriptions seja chave candidata na transação a que pertence, caso contrário pode surgir uma ambiguidade.



Control Info	
Control Type	Edit
Input Type	Descriptions
Suggest	No
Item Values	EmployeeId
Item Descriptions	EmployeeName

Por exemplo, suponhamos que na transação Airport tenhamos que atribuir um funcionário da agência encarregado da coordenação de voos. Para fazer isso, inserimos os atributos EmployeeId e EmployeeName da transação Employee. Se para EmployeeId alterarmos a propriedade InputType para Descriptions, e em Item Descriptions colocamos EmployeeName, para inserir o nome do funcionário em vez da chave, pode ser apresentado um erro de ambiguidade, pois EmployeeName certamente não é uma chave candidata, perfeitamente pode haver dois funcionários com o mesmo nome.

Vemos que no output é mostrado um warning alertando esta situação. Se colocarmos um nome e houver mais de um registro com esse mesmo nome, será apresentado um erro, pois não saberá qual manter.

Por isso, é sempre importante levar isto em consideração. Outra consideração é que o atributo inserido em Item Descriptions deve estar na transação em que usamos esta propriedade, caso contrário, nos lançará um erro. E também deve ser de domínio char ou varchar.

A configuração desta propriedade Input Type e sua associada Item Descriptions, podemos fazer no nível do atributo como acabamos de ver, se for aplicada desta forma terá este comportamento em cada transação que apareça o atributo, exceto em sua própria transação, neste caso Country. Mas também podemos configurar

estas propriedades diretamente no Web Layout de uma transação, e só se aplicaria a esse formulário, neste caso devemos levar em consideração que se perde o dinamismo e que manualmente deverá ser alterado o nome do rótulo e remover o atributo descrição, que será desnecessário. Também podemos fazer uso desta funcionalidade em qualquer Panel ou Web Panel que tenhamos.

Este comportamento é semelhante a quando configuramos um atributo que é chave estrangeira, com o tipo Dynamic Combo Box, com a diferença de que aqui não podemos escrever o valor que queremos, mas o escolhemos entre as opções exibidas.

Control Info	
Control Type	Edit
Input Type	Descriptions
Suggest	Incremental
Item Values	CountryId
Item Descriptions	CountryName
Filter Operator	Begins with
Suggest Max Items	5
Sort Descriptions	True
Case Sensitive	False
Conditions	
Instantiated Attribut	

CountryId's Conditions

```
CountryName LIKE "S%";
```

Airport

« < > » SELECT

Id

Name

Country Name

City Id

City Name

Employee Name

A desvantagem que podemos encontrar em ter que escrever o nome descritivo de um registro que queremos selecionar, é que podemos não lembrar dele exatamente ou cometer erros na hora de escrevê-lo. Para ajudar o usuário na inserção, após a propriedade input Type configurada com o valor descriptions, aparece a propriedade Suggest, por padrão vem definida como “No”, mas se a alteramos para “Incremental” serão sugeridos ao usuário os valores que correspondam à medida que o usuário vá adicionando letras.

Vamos vê-lo em execução, à medida que vamos inserindo letras, vai sugerindo as diferentes opções.

Da mesma forma que vimos com a propriedade Input Type, Suggest pode ser configurado no nível do atributo ou no nível do campo que interesse.

Uma vez alterado o valor da propriedade, são habilitadas novas propriedades, por exemplo a chamada Filter Operator, para configurar se interessa que os registros a serem sugeridos coincidam desde o início com o que foi inserido ou que simplesmente contenham essas letras. Além disso, se queremos que o filtro seja case sensitive ou não, ou até mesmo poder colocar um limite de registros a sugerir, por padrão é definido como 5. Este último pode ajudar a reduzir tempos de carga no caso de muitos registros.

Outra coisa que nos permite é poder colocar condições para esta pesquisa, para dar

um exemplo, se por algum motivo só quisermos que os países a serem exibidos sejam aqueles que começam com a letra “S”, podemos inserir a seguinte condição. Em execução, se inserimos qualquer outra letra, não irá sugerir os registros que existem e coincidem, mesmo se colocamos o nome de um país que existe na base de dados não o encontrará, ou seja, não o sugere nem nos permite registrá-lo.

The image shows three components of the GeneXus IDE:

- Data Models:** Three models are shown: 'Product' (ProductId, ProductName), 'Supplier' (SupplierId, SupplierName, Product, ProductId, ProductName), and 'Claim' (ClaimId, ClaimDate, ClaimDescription, SupplierId, SupplierName, ProductId, ProductName). Arrows indicate relationships between attributes in different models.
- Control Info Table:** A table showing properties for a control. The 'Instantiated Attributes' property is set to 'SupplierId'. The 'Auto correction' property is set to 'False'.
- Runtime Form:** A form with fields for 'Supplier Id' (value 1), 'Supplier Name' (value 'SupplierA'), and 'Product Name' (value 'Pr'). A dropdown menu is open, showing 'Product1' and 'Product2' as suggestions. 'CONFIRM' and 'CANCEL' buttons are at the bottom.

Outra das propriedades que apareceu é “Instantiated Attributes”, aqui temos outra maneira de limitar o conjunto de valores possíveis que serão sugeridos para inserir. É utilizado para filtrar um conjunto de atributos em função de outros atributos que podem ser inferidos diretamente.

Para esclarecer isto, vejamos em um exemplo:

Suponhamos que estamos no desenvolvimento de uma aplicação para uma empresa, onde há uma transação para inserir os produtos que lá são vendidos, outra transação para inserir os fornecedores desses produtos, e por sua vez para cada fornecedor são registrados os produtos que esse fornecedor tem e que a empresa comercializa. E temos outra transação que será usada para inserir reclamações que se tenham de um determinado produto e fornecedor.

Suponhamos que cada produto terá um nome único, e estamos interessados em pesquisar os produtos digitando seu nome, então configuramos o atributo productId com Input Type em Descriptions e em Item Descriptions colocamos ProductName. E deixamos a propriedade Suggest com o valor “Incremental” para que nos sugira os registros.

Para este exemplo, teremos 2 produtos para o fornecedor A e 2 produtos para o fornecedor B.

Vamos inserir uma reclamação, selecionamos o fornecedor, e quando vamos escrever um produto, vemos que sugere também os produtos do fornecedor B, ou seja, está

mostrando todos os produtos da tabela Product. E o que nos interessa é que nos sugira apenas os produtos que são do fornecedor A, ou seja, do que escolhemos, para isso deverá buscar na tabela SupplierProduct apenas os produtos que tenham o fornecedor A.

A propriedade que acabamos de ver (Instantiated Attributes), é justamente para poder limitar o conjunto dos possíveis valores exibidos e/ou inseridos, com base na informação que tenhamos instanciada na transação.

Para que apenas sejam mostrados os produtos do fornecedor escolhido, deveremos indicar o atributo SupplierId como instanciado, na propriedade mencionada. Os atributos aqui indicados participam na determinação da tabela que deverá ser navegada para sugerir e aceitar os valores inseridos pelo usuário. Neste caso, ao inserir SupplierId e ter a necessidade de recuperar os ProductId, a tabela que será navegada será SupplierProduct.

Vemos em execução que agora só nos traz os produtos do fornecedor escolhido.

Control Info	
Control Type	Edit
Input Type	Descriptions
Suggest	Incremental
Item Values	ProductId
Item Descriptions	ProductName
Filter Operator	Begins with
Suggest Max Items	5
Sort Descriptions	True
Case Sensitive	False
Conditions	
Instantiated Attributes	SupplierId
Auto correction	False
Auto capitalization	None
Notify Context Change	False

VS.

Control Info	
Control Type	Dynamic Comb...
Data Source From	Attributes
Item Values	ProductId
Item Descriptions	ProductName
Sort Descriptions	True
Conditions	
Instantiated Attributes	SupplierId
Empty Item	False
Notify Context Change	False

- Allows you to write the value
- You can specify a maximum number of values to display (Max Items)

- It does not allow to write the value, it allows to select it from a list
- Cannot specify a maximum number of values to display

Para os Dynamic Combo box também é possível utilizar a propriedade Instantiated Attributes, bem como também a propriedade Conditions.

Como vimos, é similar o uso dos Dynamic combo box a quando colocamos o Input Type em Descriptions. Entre algumas diferenças, os controles Edit permitem digitar valores, inclusive diferentes dos sugeridos, enquanto os controles combo não. Outra diferença é que nos controles Edit, como vimos, é possível especificar um número máximo de valores oferecidos com a propriedade Max Items e nos combos dinâmicos não temos esta possibilidade, então se a quantidade de registros for muito grande, ainda carregará todos eles, essa é uma vantagem dos controles Edit.

Até aqui vimos um resumo do uso das propriedades vistas, para ampliar os conhecimentos recomendamos pesquisar em nossa Wiki.

GeneXus[™]

training.genexus.com
wiki.genexus.com