**[ABAP 105 - Entendendo o Fieldcatalog](http://www.sonikro.com.br/abapinho/13996561" \o "ABAP 105 - Entendendo o Fieldcatalog)**

Olá, vamos fazer uma aula em texto hoje.

**1. O Que é um Fieldcatalog ?**

R: Um fieldcatalog nada mais é do que um Catalogo de Campos, ou seja, ele é utilizado para evidênciar os campos de uma estrutura ALV e suas respectivas caracteristicas.

**2. Situação:**

Tenho uma tabela interna da SFLIGHT, porém **não** quero mostrar todos os campos, apenas alguns.

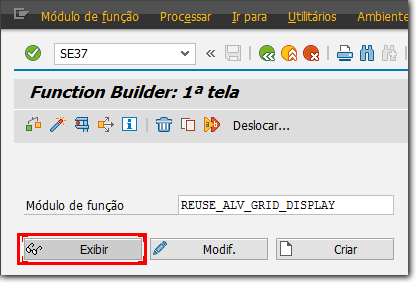
Sendo assim terei que especificar para o ALV, quais serão os campos que aparecerão no relatório, para isso utilizaremos o **FIELDCATALOG.**

**OBS:** Existem diversas maneiras de fazer um ALV, os principais são através da função **REUSE\_ALV\_GRID\_DISPLAY,**ou através das classes **CL\_GUI\_ALV\_GRID** ou **CL\_SALV\_TABLE.**

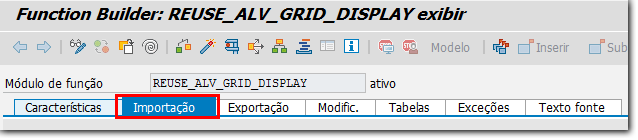
Neste exemplo, estarei utilizando a função **,**pois foi o que eu mostrei na video-aula passado, e por ser o método mais facil para quem está começando.

**3. Análisando parâmetros da função REUSE\_ALV\_GRID\_DISPLAY**

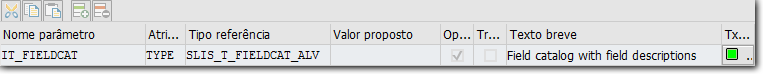
Vamos abrir a função **REUSE\_ALV\_GRID\_DISPLAY,**na transação **SE37** para analisarmos seus parâmetros.



Vá até a aba **Importação** (Parâmetros de entrada na função, ou seja, vai do programa para a função).



E repare que esta função possui um parâmetro chamado **IT\_FIELDCAT**



A partir de agora, já sabemos qual parâmetro devemos preencher na chamada da função no nosso programa.

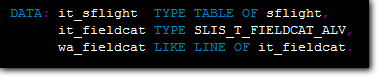
**OBS:**Os tipos de dados que o seu programa vai utilizar para enviar para a função, devem ser compativeis (Iguais), caso contrário ocorrerá um DUMP.

**EX:**A função está preparada para receber um parâmetro do tipo númerico, e você envia um parâmetro do tipo String, neste caso ocorrerá um DUMP.

Para isto não ocorrer, devemos utilizar o mesmo Tipo no nosso programa, ou seja, o tipo **SLIS\_T\_FIELDCAT\_ALV**.

**4. Declarando o Fieldcatalog no nosso programa.**

Vamos primeiro declarar as variáveis que utilizaremos para montar o FIELDCAT no nosso programa:



Repare que o tipo SLIS\_T\_FIELDCAT\_ALV é um tipo tabela, sendo assim a variavel será uma **Internal Table**, mas para preencher os dados da tabela interna, precisamos de uma **WorkArea** com o mesmo tipo de linha, sendo assim declaramos utilizando a sintaxe **LIKE LINE OF**.

**OBS:** Não vou mostrar a parte de popular a tabela IT\_SFLIGHT, pois isso já foi abordado no[**ABAP104 - Criando seu primeiro ALV**](http://sonikro.virb.com/abapinho/13952809).

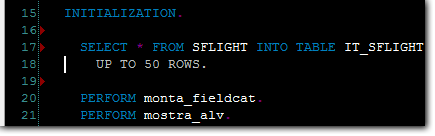
Vou pular para a criação do fieldcat, e a chamada da função.

**5. Populando o Fieldcatalog**

Aqui entra a parte principal. Os campos que serão exibidos no ALV, serão os campos que estiverem no Fieldcatalog (**IT\_FIELDCAT).**

Na nossa situação, eu quero mostrar apenas os campos  CARRID, CONNID, FLDATE e o PRICE.

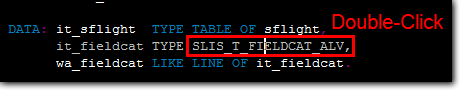
Para nosso programa, vamos criar **2 subrotinas :** **monta\_fieldcat** e **mostra\_alv.**



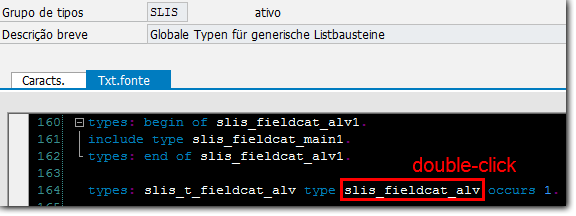
As instruções para preencher a estrutura **IT\_FIELDCAT** é bem simples, basta preencher a Workarea **WA\_FIELDCAT** com os valores necessários e dar um APPEND na tabela **IT\_FIELDCAT.**

Porém você não sabe quais campos a estrutura possui, muitas pessoas copiam e colam da internet os principais campos, mas vou ensinar vocês como descobrir estes campos.

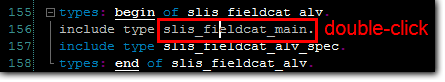
**Clique-Duplo no tipo SLIS\_T\_FIELDCAT\_ALV**



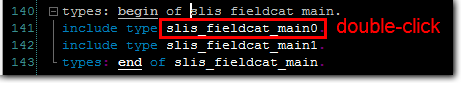
Repare que este tipo SLIS ele é um grupo de Tipos Global, vamos continuar navegando até chegar nos campos desejados:

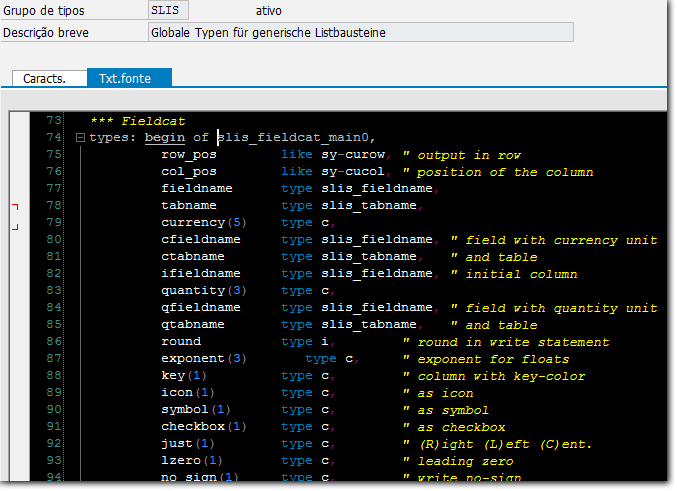


Veja que o TYPE **SLIS\_FIELDCAT\_ALV** é composto por 2 estruturas, vamos navegar para a principal (**\_MAIN)**



Veja novamente que a estrutura **\_main**é composta por outras subestruturas, vamos navegar para a **main0.**

E finalmente chegamos nos principais campos do fieldcatalog:



A maioria dos campos tem nomes auto-explicativos, para começar, vamos utilizar os seguintes campos:

* col\_pos (Posição da Coluna)
* fieldname (Nome técnico do campo)
* emphasize (Dar ênfase)
* fix\_column (Fixar Coluna)
* outputlen (Comprimento de Saida)
* seltext\_l (Texto descritivo LONGO)
* seltext\_m (Texto Descritivo Médio)
* seltext\_s (Texto Descritivo Short)

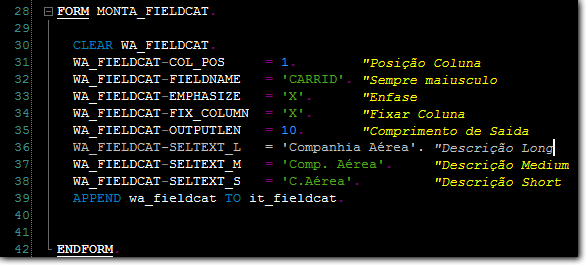
Existem vários outros que vocês podem preencher e descobrir o que acontece :)

Agora que já sabemos os campos que iremos precisar preencher no nosso fieldcat, vamos voltar ao nosso programa e criar o **form monta\_fieldcat.**

**6. Criando FORM monta\_fieldcat.**

Lembrando que os campos que queremos mostrar são : CARRID, CONNID, FLDATE e o PRICE.

Vou montar primeiro apenas para o campo **CARRID**.



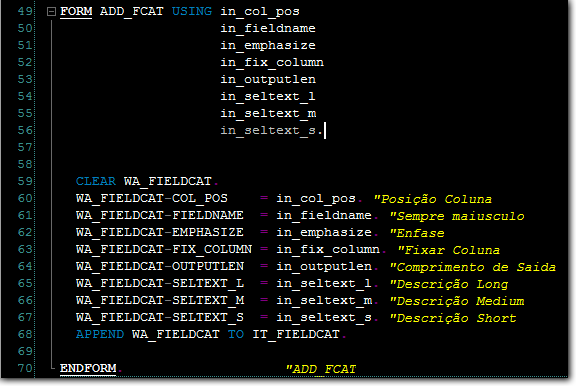
A partir de agora ficou facil, basta copiar e colar isso e ir alterando as informações..... **MAS NÃO FAÇAM ISSO ! É FEIO !**

Vejo muitas pessoas que copiam e colam este bloco e vão alterando as informações, mas esta não é a melhor maneira de fazer isso, pois podemos **modularizar** este problema.

Você reparou que iremos preencher sempre os mesmos campos para criar o fieldcat, e se eu copiar e colar isso pra todo campo que eu quizer fazer estarei repetindo praticamente as mesmas instruções com valores diferentes ? Vamos então **modularizar** nosso problema, criando um **FORM add\_fcat** com **parâmetros** de entrada.

**7. Criando FORM add\_fcat.**

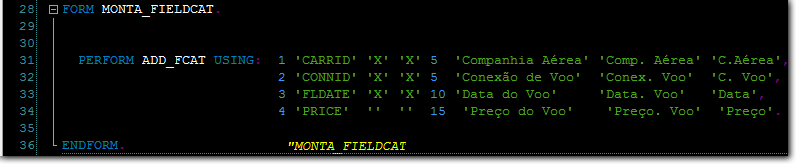
Vamos criar uma subrotina com **parâmetros,**esta subrotina irá receber os valores do fieldcat, e irá preencher a estrutura, e appendar automaticamente na **IT\_FIELDCAT:**



Repare que é exatamente a mesma coisa que fizemos com o campo **CARRID**, porém substituimos os valores fixos, por váriaveis.

**8. Reformulando FORM monta\_fieldcat.**

Agora que já criamos nossa rotina que irá adicionar linhas ao fieldcatalog, vamos reformular nosso form monta\_fieldcat da seguinte maneira:

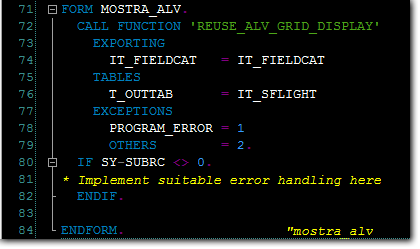


Os parâmetros **USING** na hora do PERFORM são separados por **espaços,** e ao inserir virgulas ele entende que ele deverá executar a subrotina novamente, com outros parâmetros.

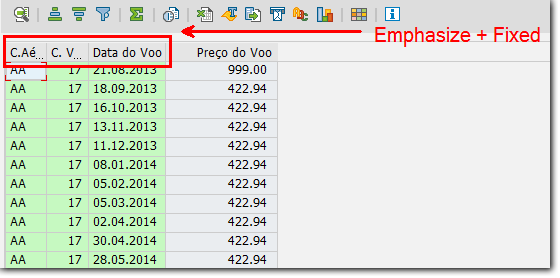
Essa  codificação irá fazer com que a subrotina **add\_fcat** seja executada 4x, cada vez com os parâmetros acima.

**9. FORM mostra\_alv**

Agora vem a parte simples, que é chamar a função **REUSE\_ALV\_GRID\_DISPLAY**, passando a tabela da **IT\_SFLIGHT** e o parâmetro que analisamos anteriormente **IT\_FIELDCAT.**



**10. Execute e veja o resultado:**

****