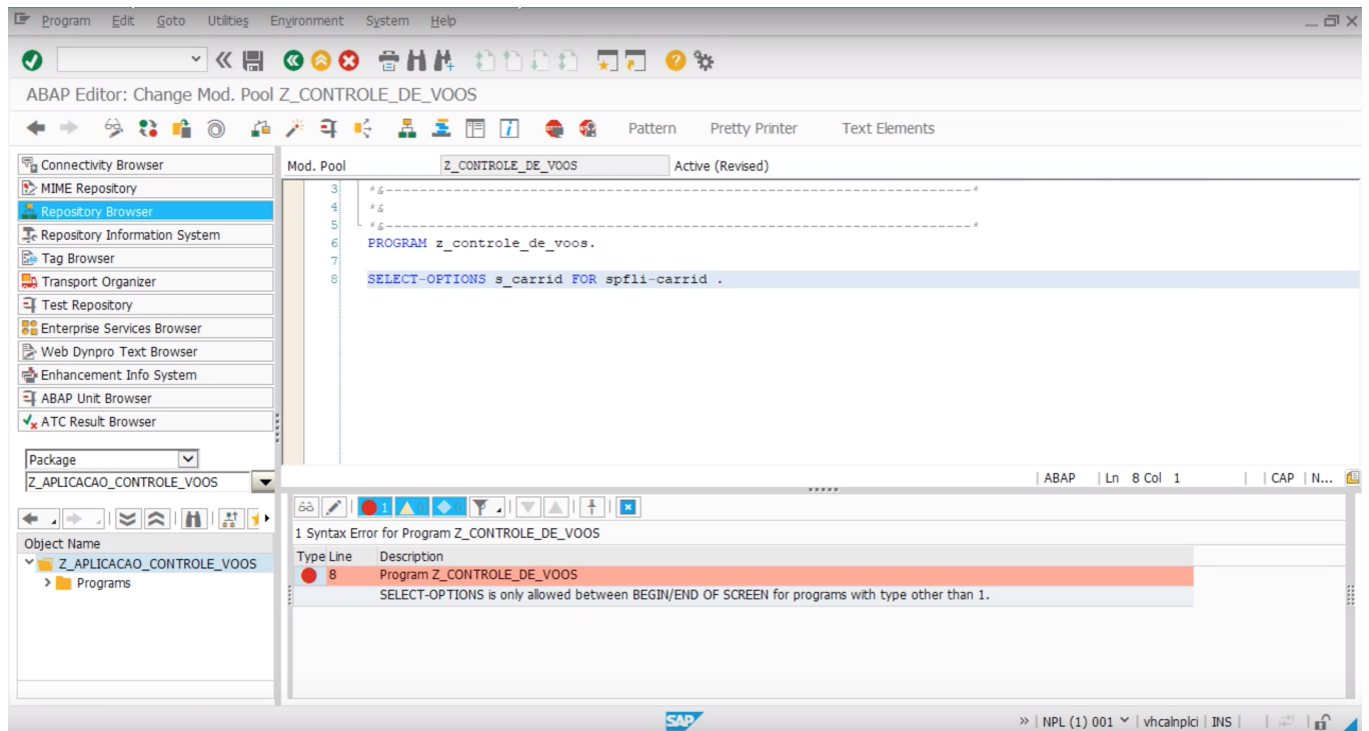
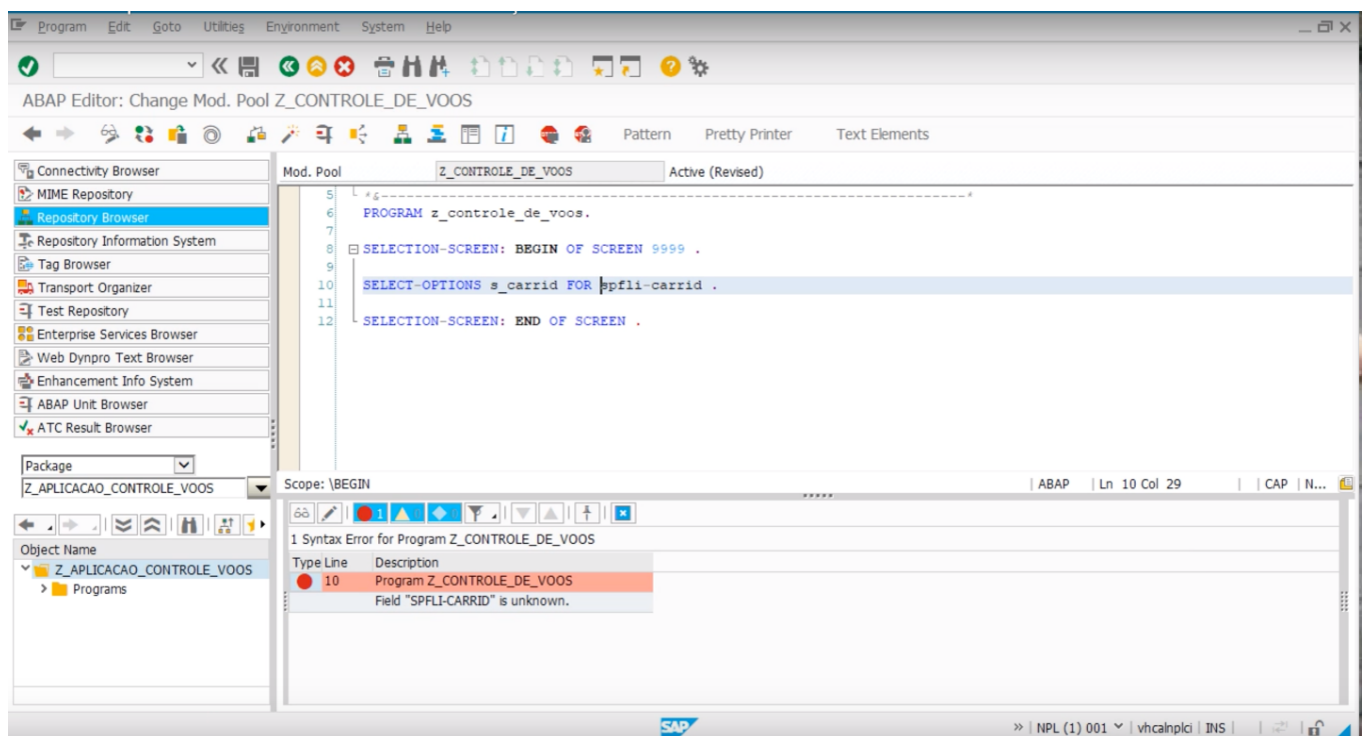


Faça o que eu fiz na aula

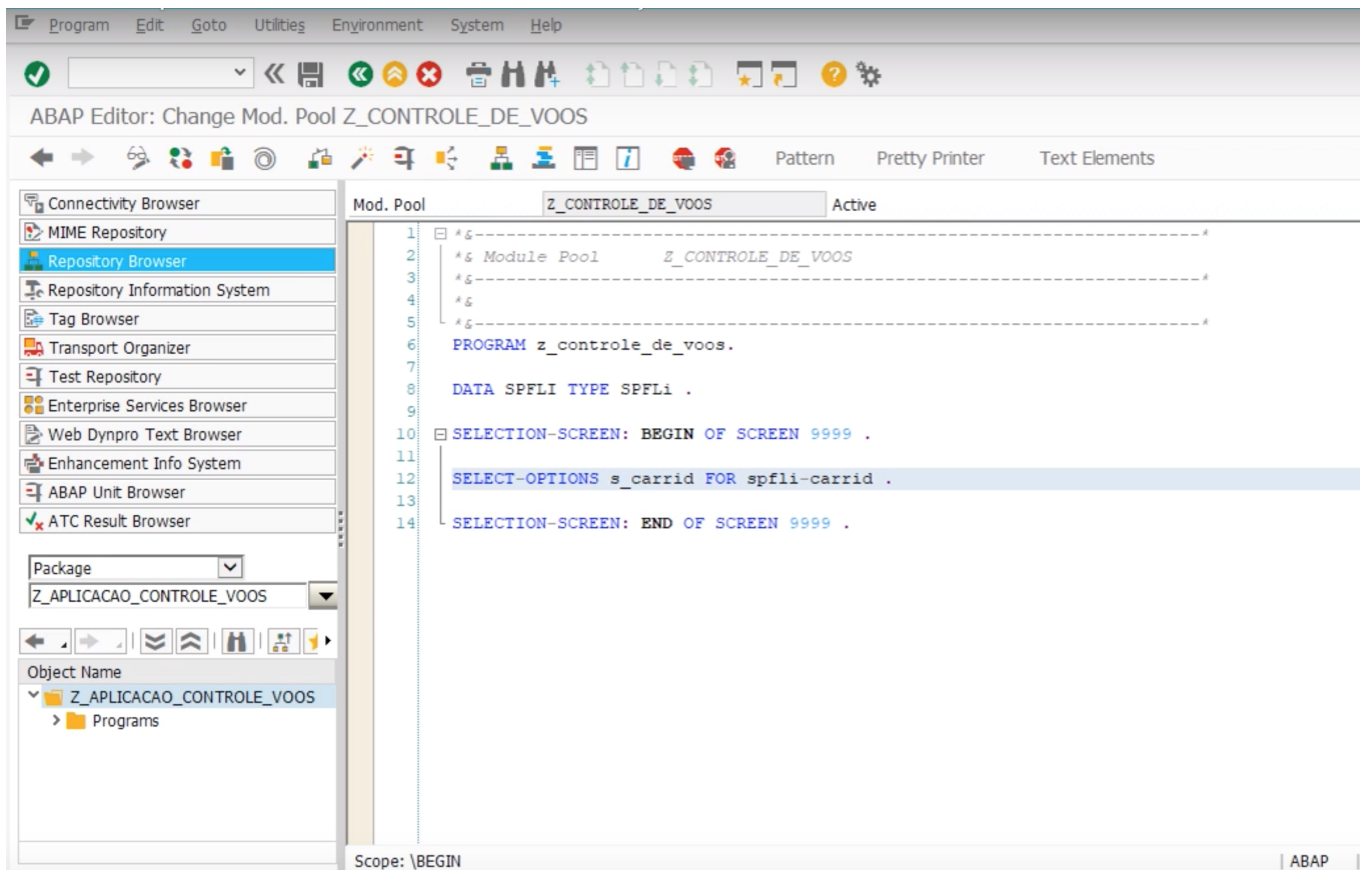
Vamos, então, iniciar o desenvolvimento da nossa tela de seleção.



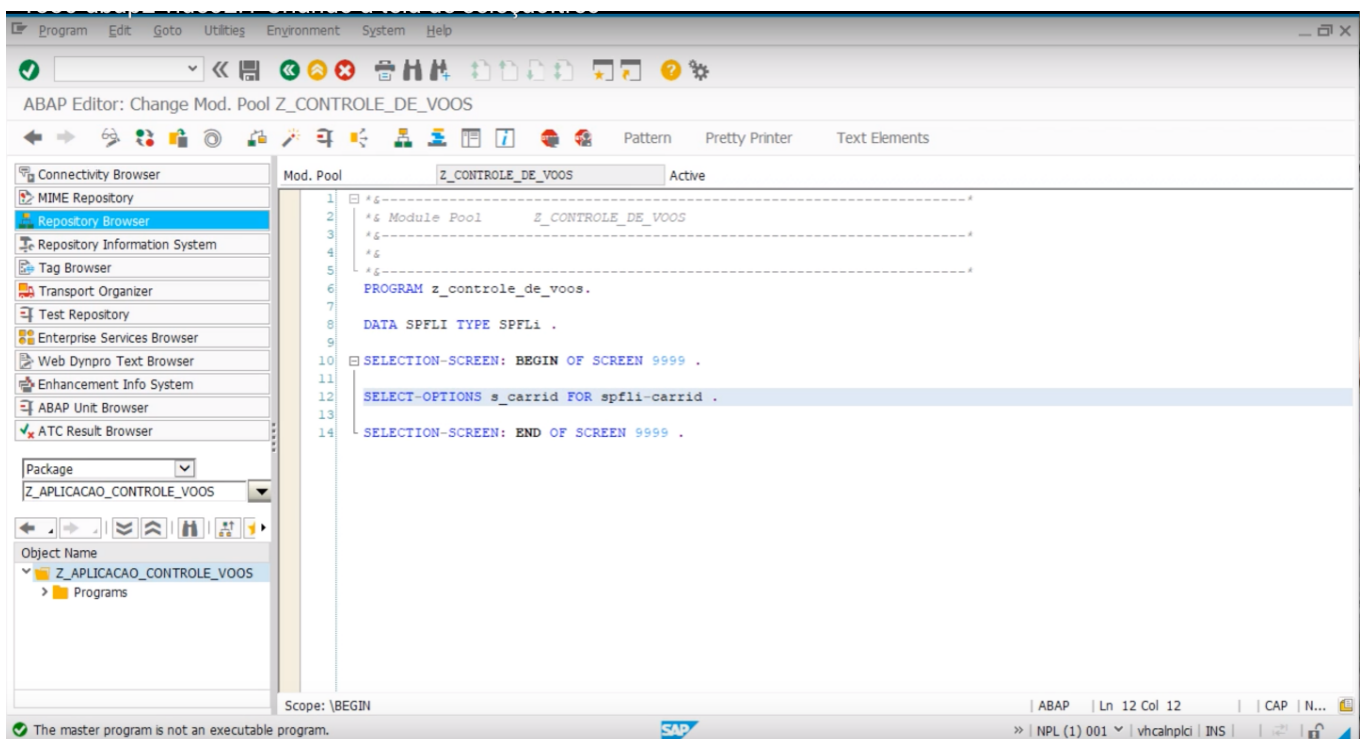
Por estarmos trabalhando com module pool, é necessário referenciar a janela. Após ter declarado a janela, um novo erro acontece por não declararmos a estrutura SPFLI que está sendo utilizada para criar a nossa tela de seleção. Ao tentar ativar o código temos uma nova mensagem de erro:



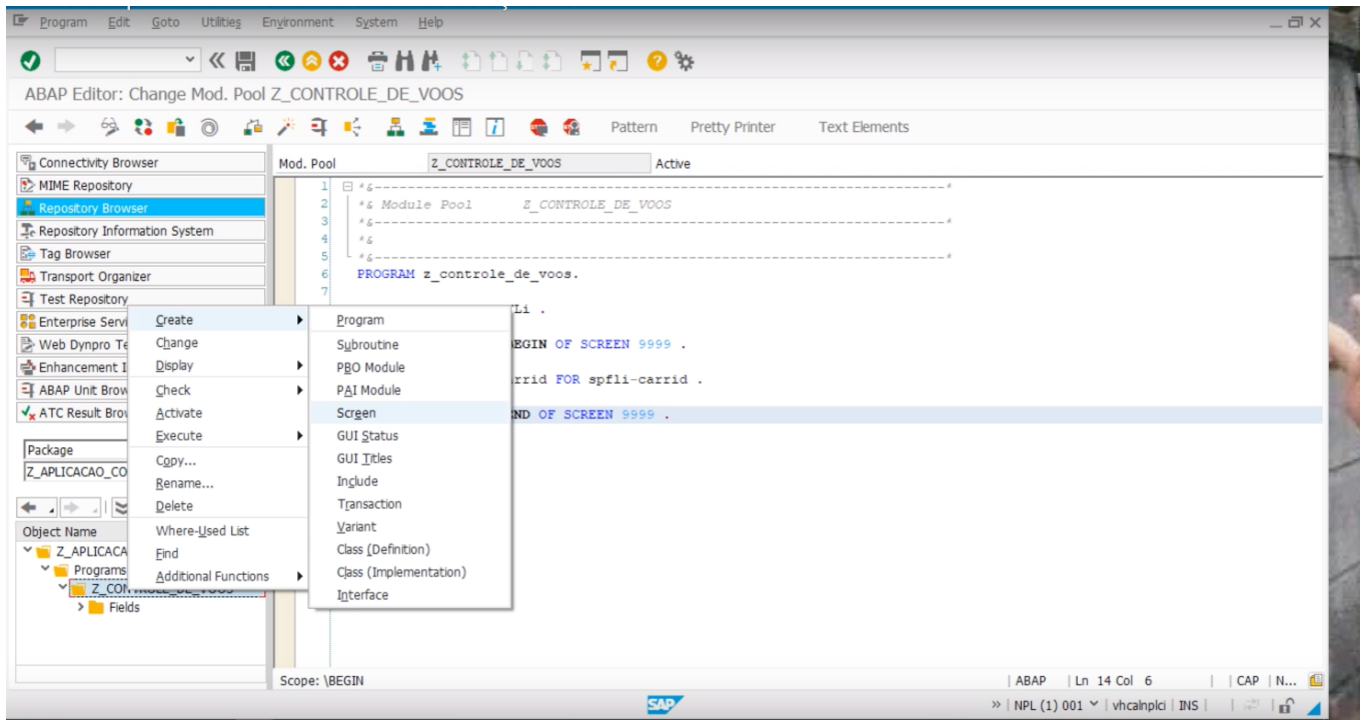
Vamos, então, declarar a estrutura SPFLI:



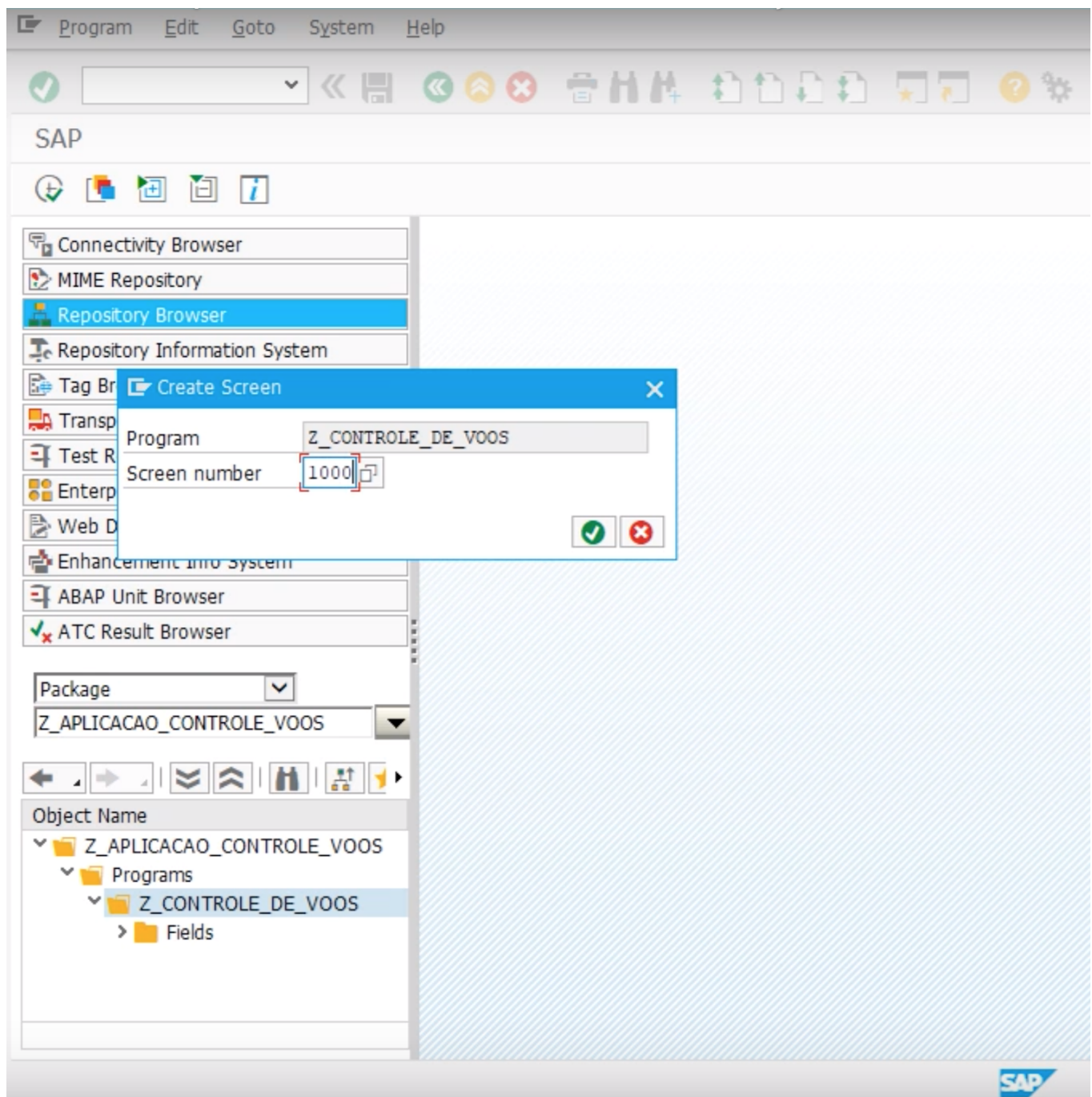
Tentamos executar a aplicação novamente. Entretanto, apenas surgiu uma mensagem informando que o programa Master não é um programa executável. Isso ocorreu pois escolhemos criar um module pool. Então, vamos criar a nossa janela master para referenciarmos os nossos campos do `select-options`.



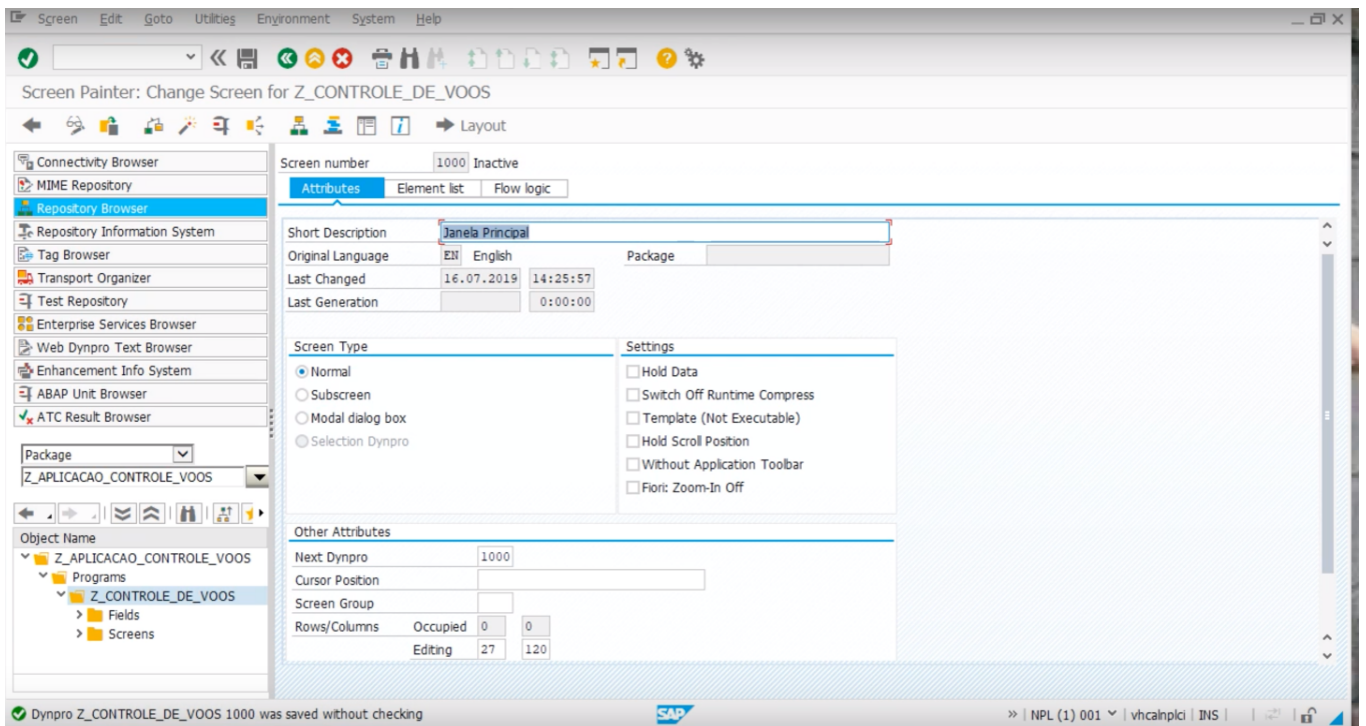
Primeiro, vamos clicar com o botão secundário no nó que representa a nossa aplicação e então selecionamos **Create > Screen**.



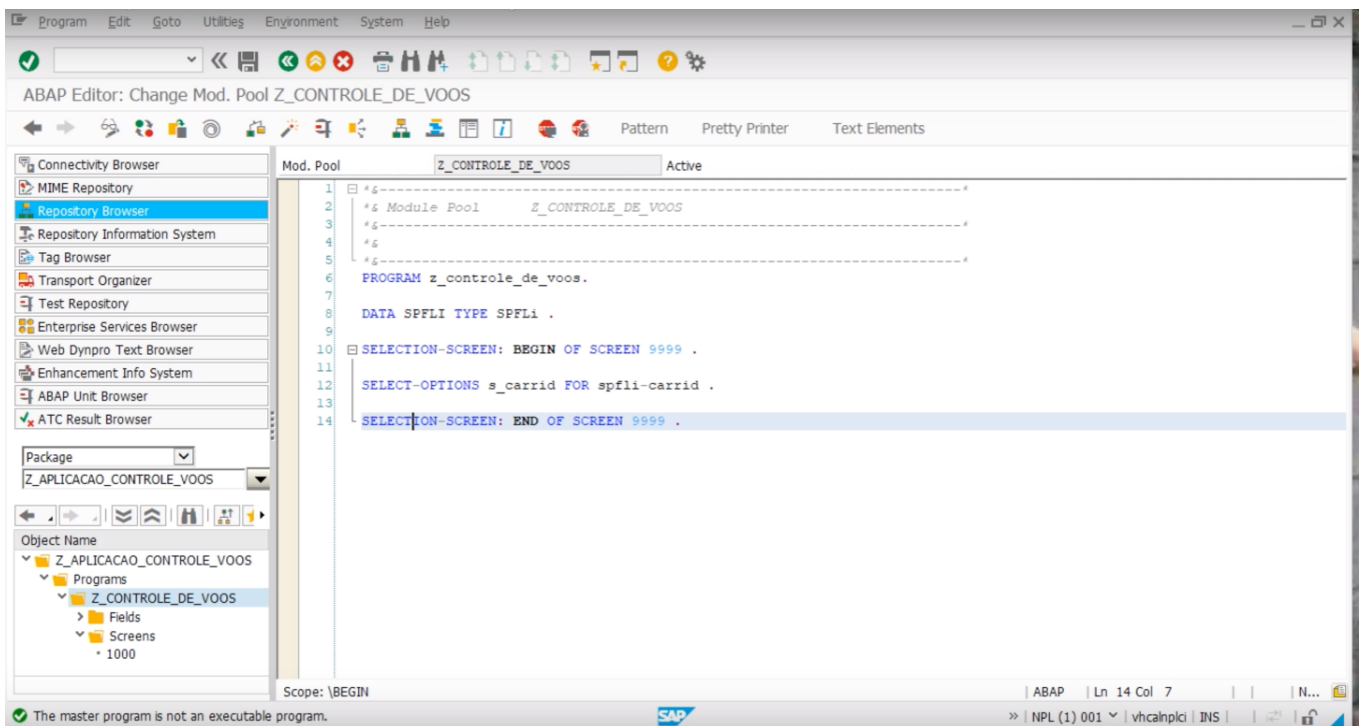
O número da nossa janela será 1000.



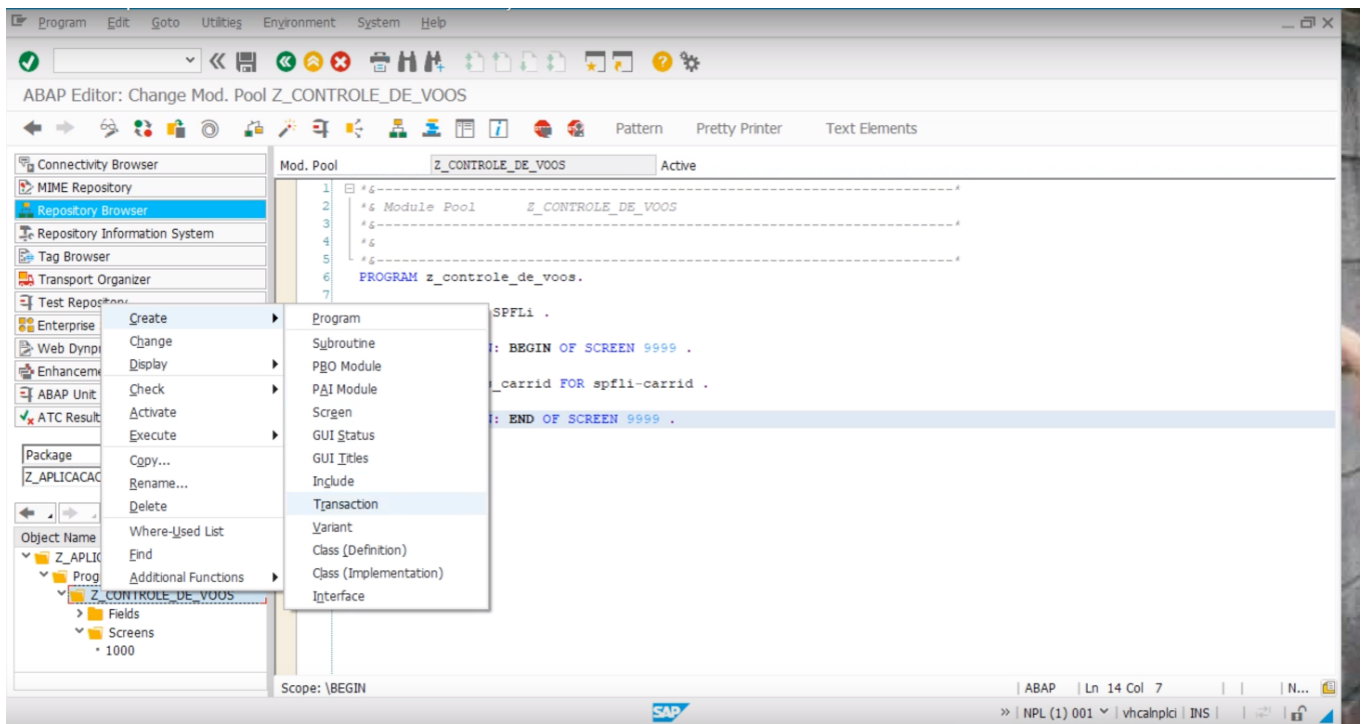
Depois de inserirmos a descrição da nossa janela, vamos deixar as configurações como as sugeridas.



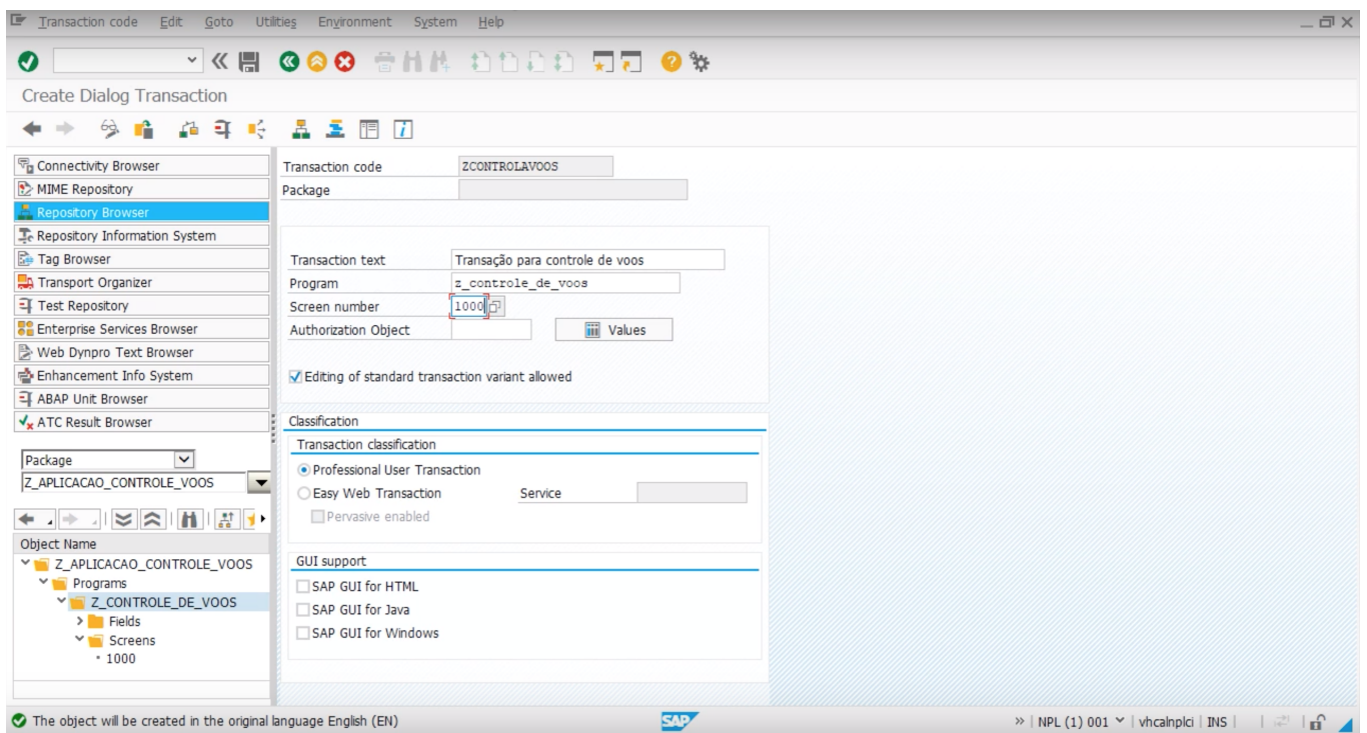
Ao tentar executar novamente a aplicação, a mesma mensagem persiste em aparecer. Isso acontece pois ainda não criamos uma transação para a nossa aplicação.



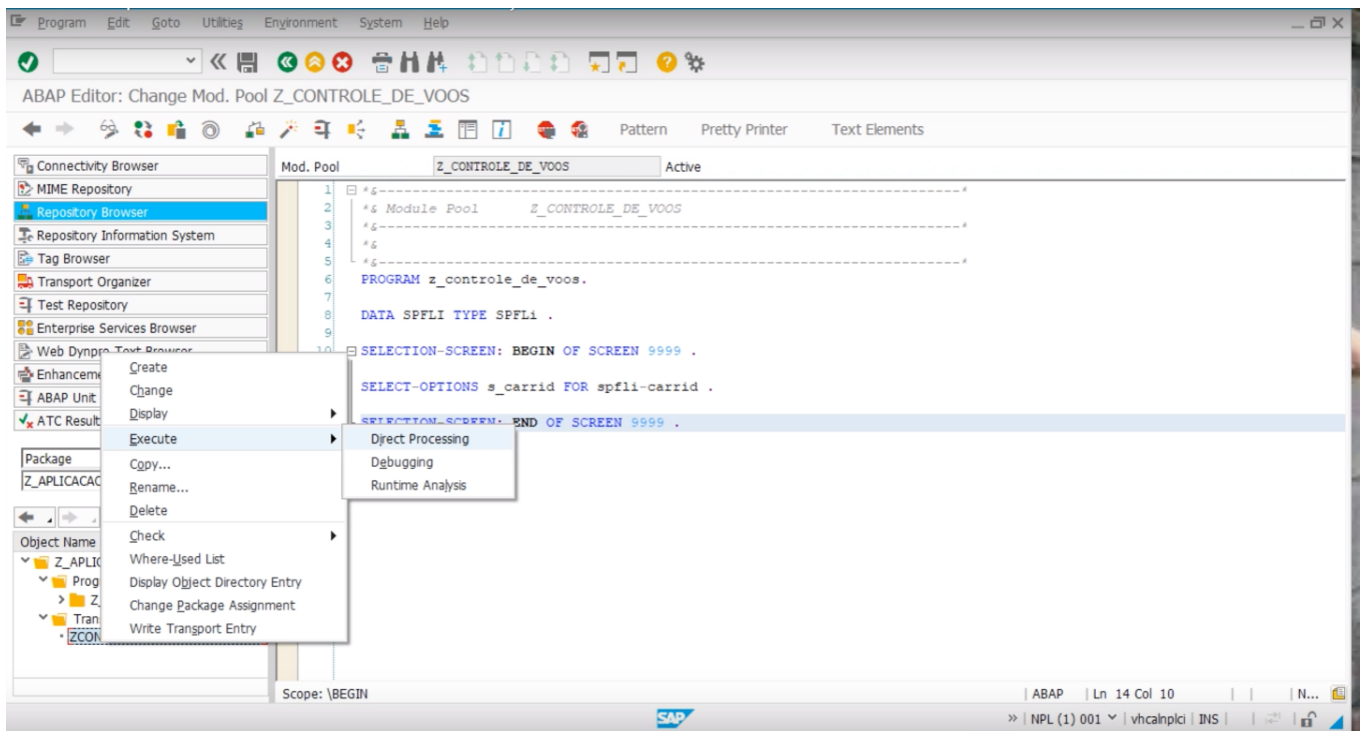
Com o botão secundário, vamos clicar no nome da nossa aplicação e depois em **Create > Transaction**.



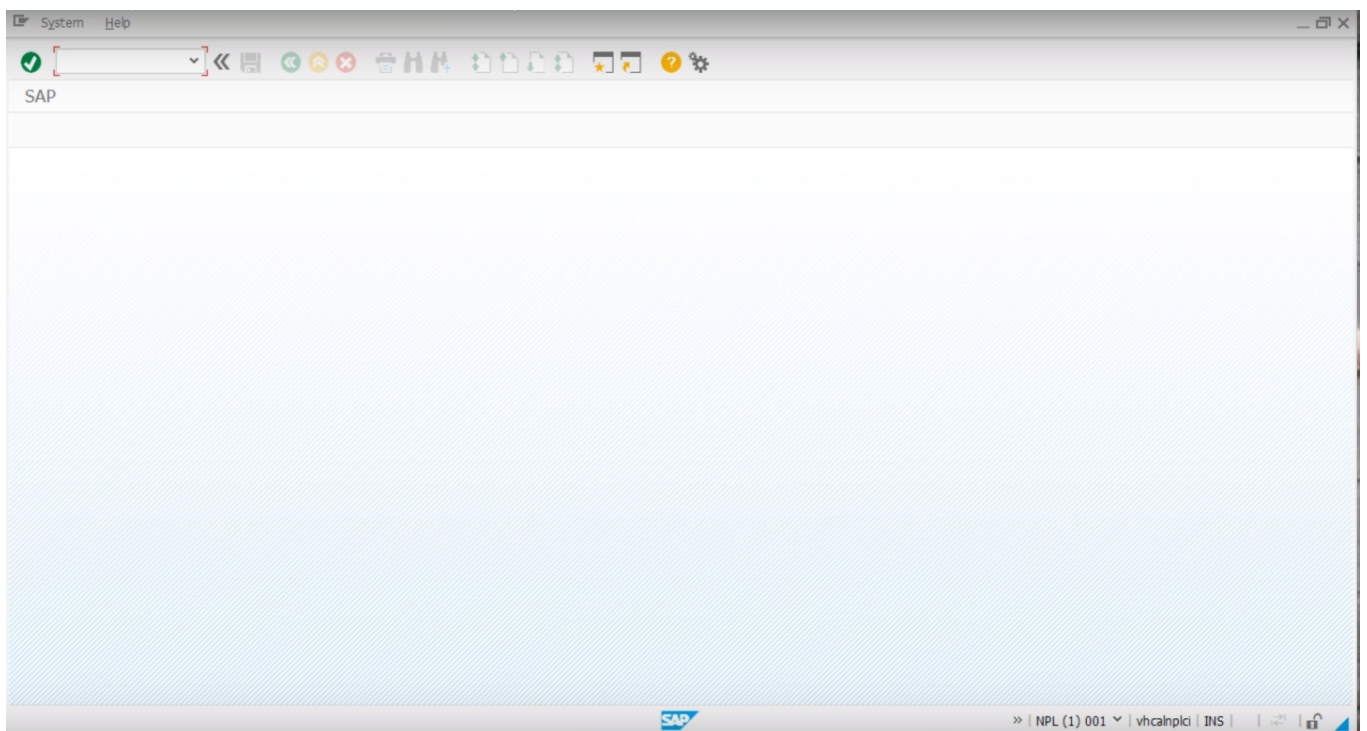
Vamos escolher a primeira opção, pois nesse momento estamos criando uma aplicação do tipo Module Pool, que é também conhecida como Dynpro ou Transação de diálogo.



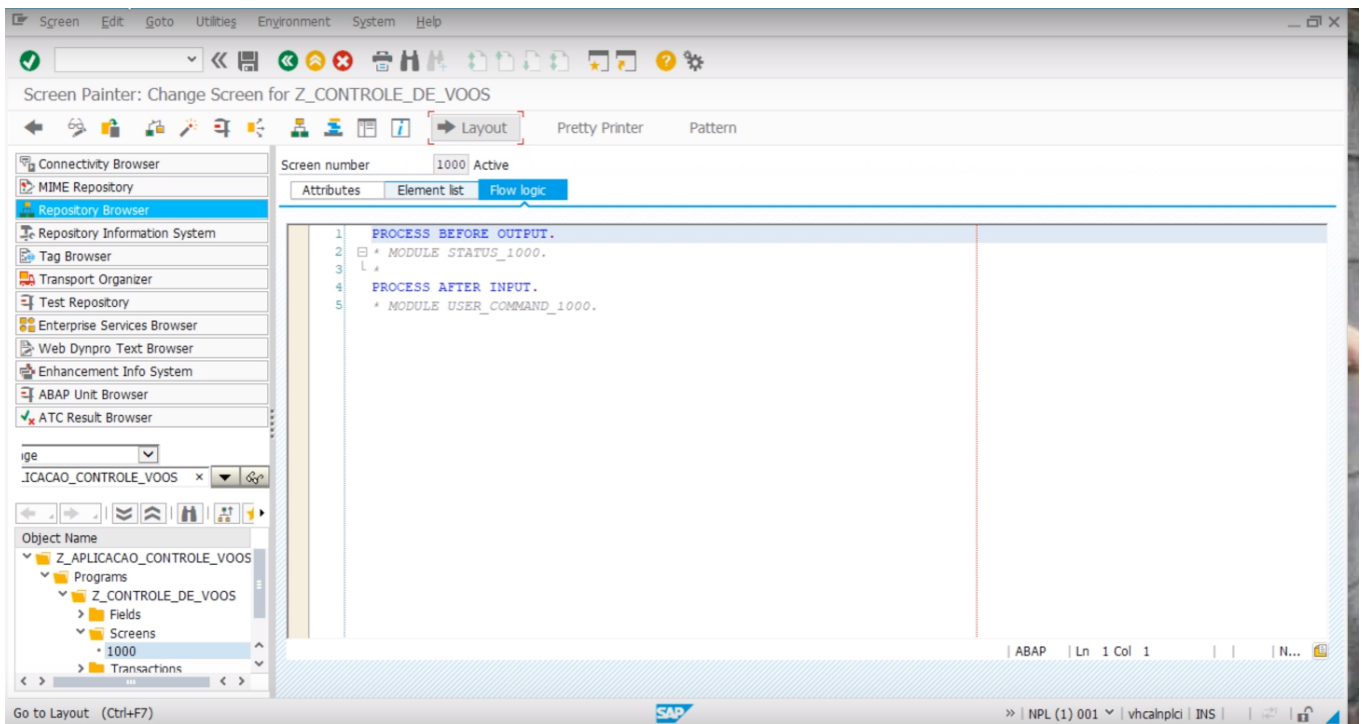
Então, uma nova janela será exibida, e nela informamos o nome da aplicação e o número da janela que deverá ser exibida na execução da aplicação.



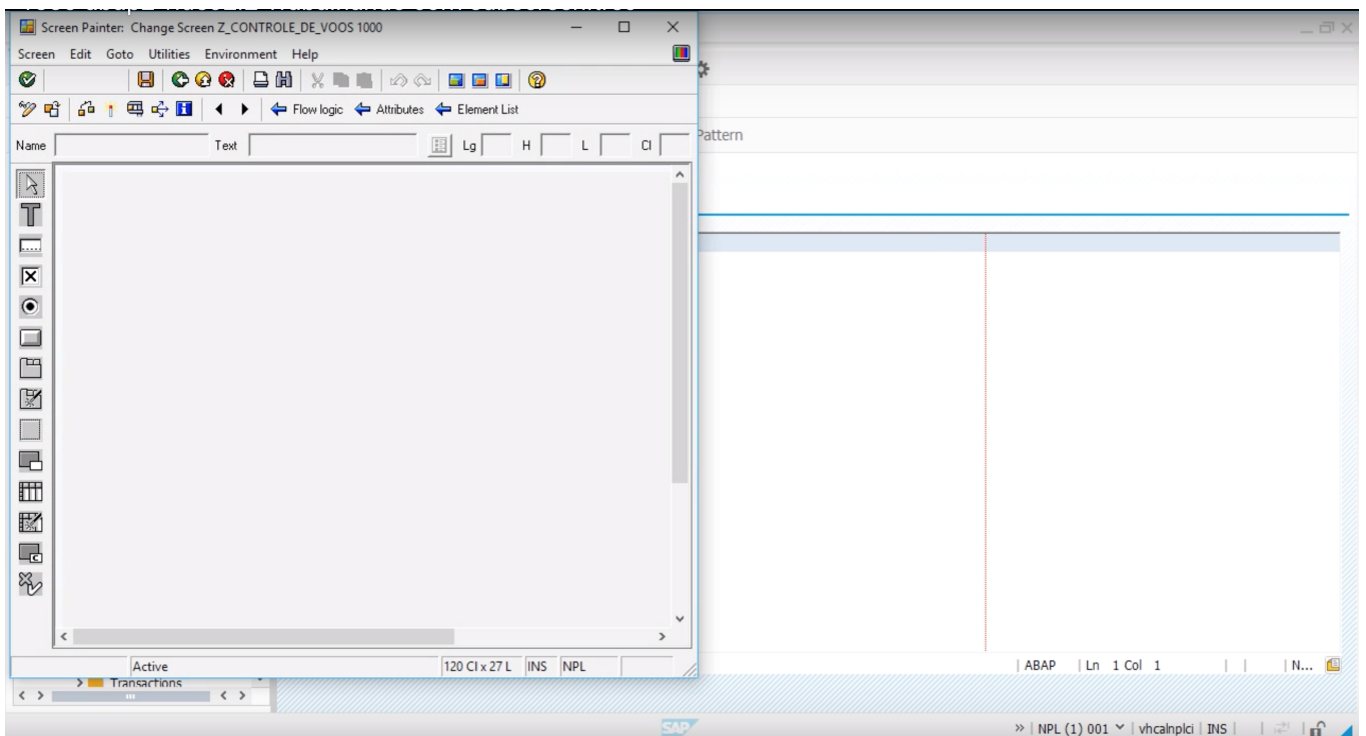
Vamos executar novamente a aplicação, mas agora utilizando a transação. Então, vamos clicar com o botão secundário em cima do nome da transação e escolher a opção: **Execute > Direct Processing**.



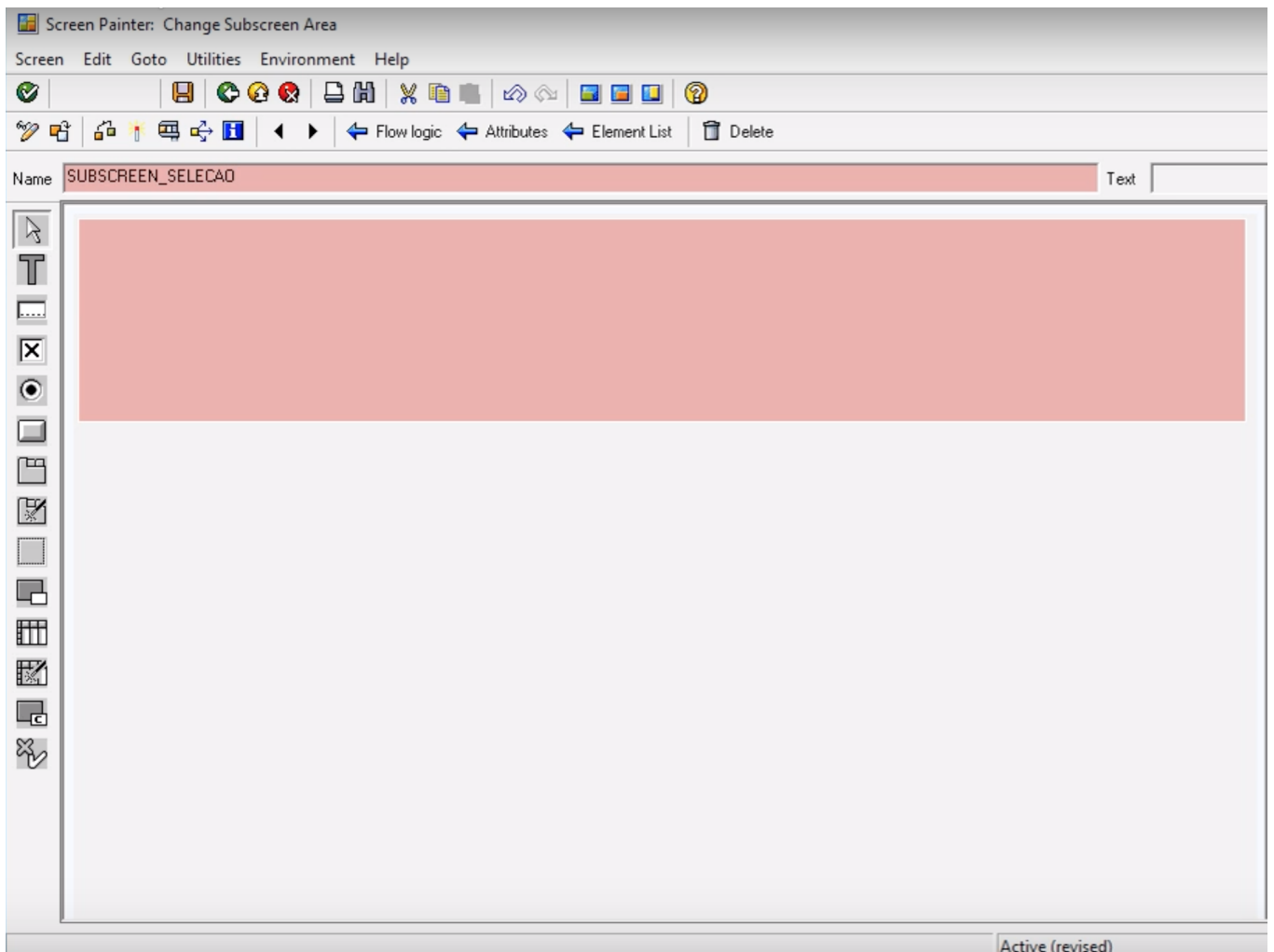
Conseguimos executar a nossa aplicação. Entretanto, nada é exibido.



Vamos verificar o layout da nossa janela 1000. Para isso damos um duplo clique no nome da janela e, então, seremos direcionados para a tela de fluxo lógico.

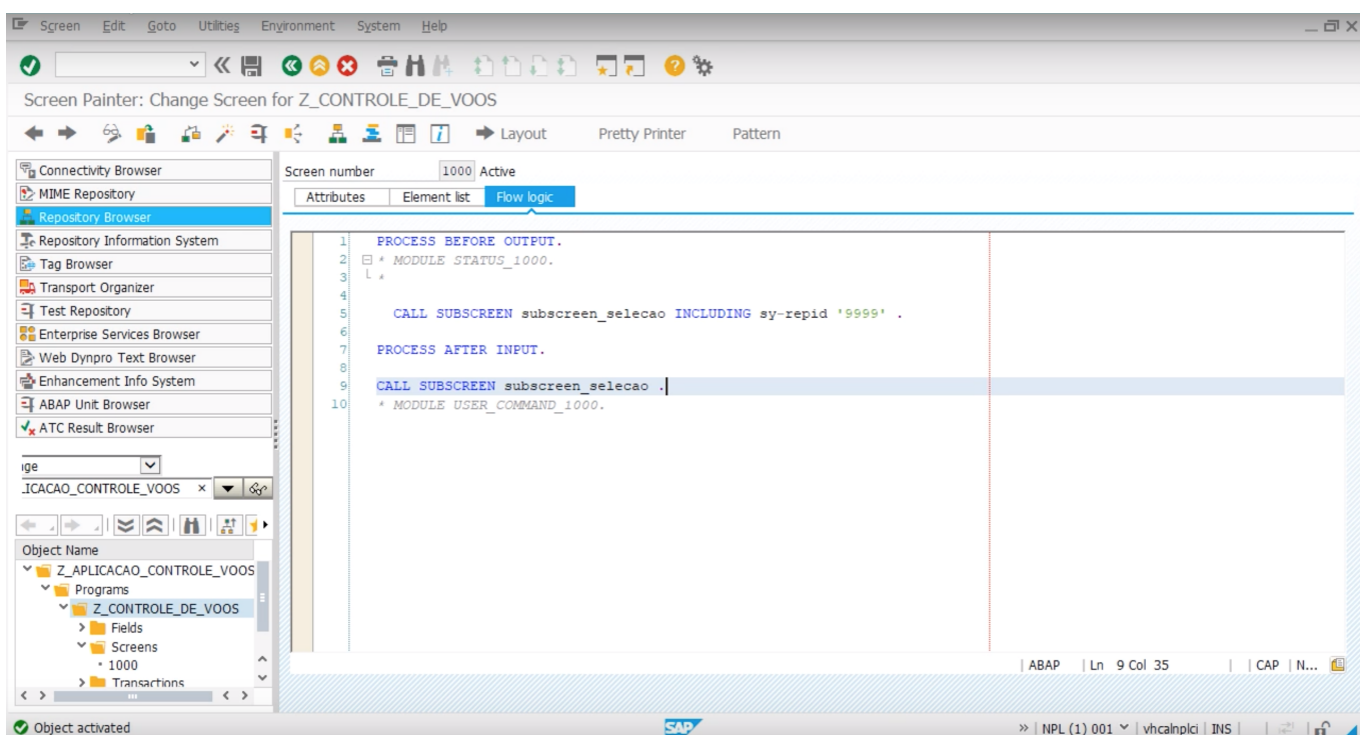


Vamos clicar na barra de tarefas na opção Layout.

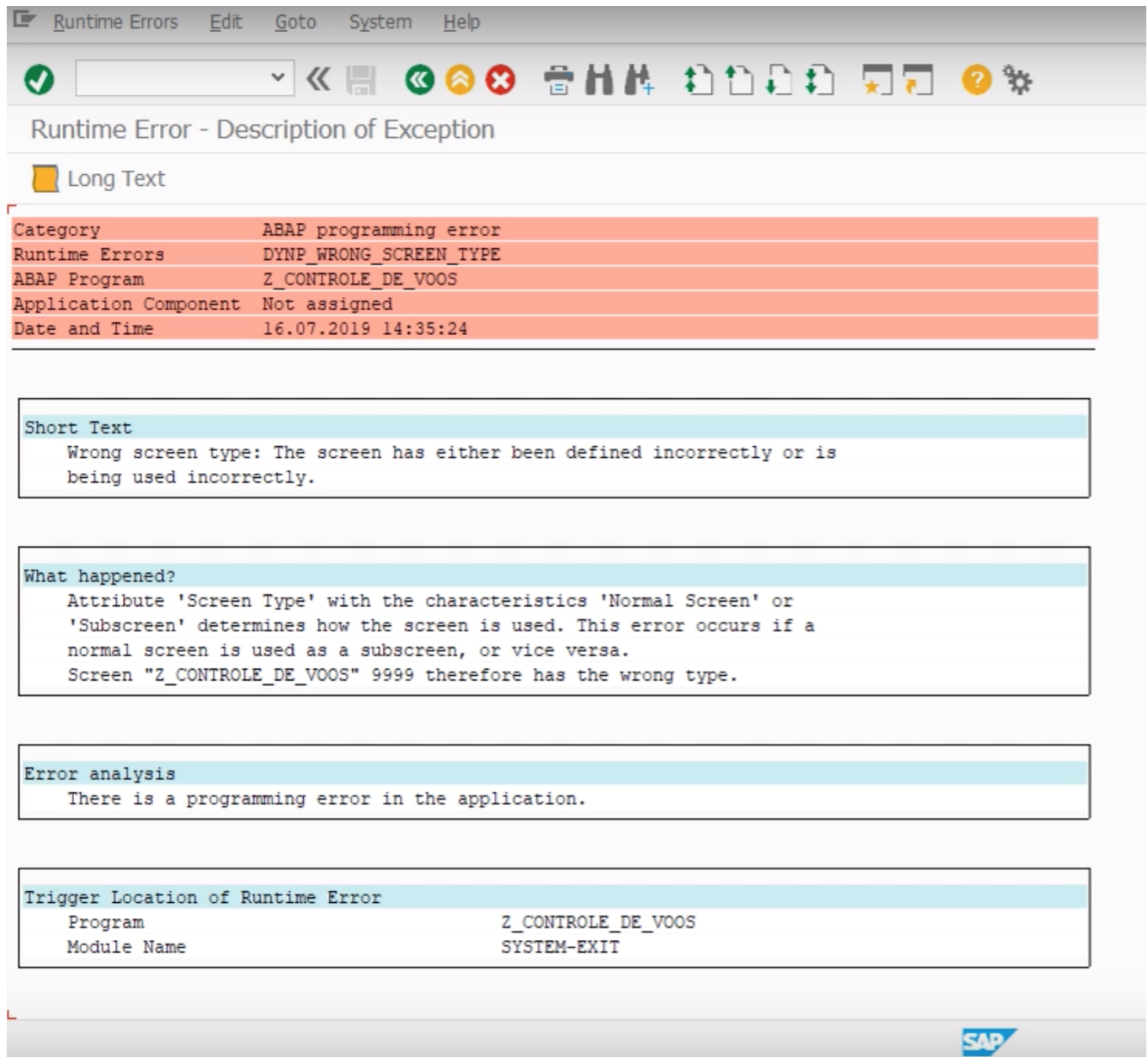


Aqui, percebemos que faz sentido que, ao executarmos a nossa aplicação, nada tenha sido exibido, porque a nossa tela principal realmente não possui nenhum elemento gráfico. Então, agora precisamos referenciar a nossa outra tela (select-options) dentro dessa tela.

Vamos, então, inserir uma subscreen na nossa tela.



Vamos ativar a nossa subscreen utilizando a **Flow Logic** e vamos iniciar a configuração nos eventos de PBO e PAI. Para isso basta utilizar a opção `CALL SUBSCREEN`, referenciando o elemento gráfico que acabamos de criar.



The screenshot shows the 'Runtime Error - Description of Exception' dialog box in SAP. The title bar includes 'Runtime Errors', 'Edit', 'Goto', 'System', and 'Help'. The main content area is titled 'Long Text' and contains a table with error details. Below the table are sections for 'Short Text', 'What happened?', 'Error analysis', and 'Trigger Location of Runtime Error'. The SAP logo is visible in the bottom right corner.

Category	ABAP programming error
Runtime Errors	DYNP_WRONG_SCREEN_TYPE
ABAP Program	Z_CONTROLE_DE_VOOS
Application Component	Not assigned
Date and Time	16.07.2019 14:35:24

Short Text
Wrong screen type: The screen has either been defined incorrectly or is being used incorrectly.

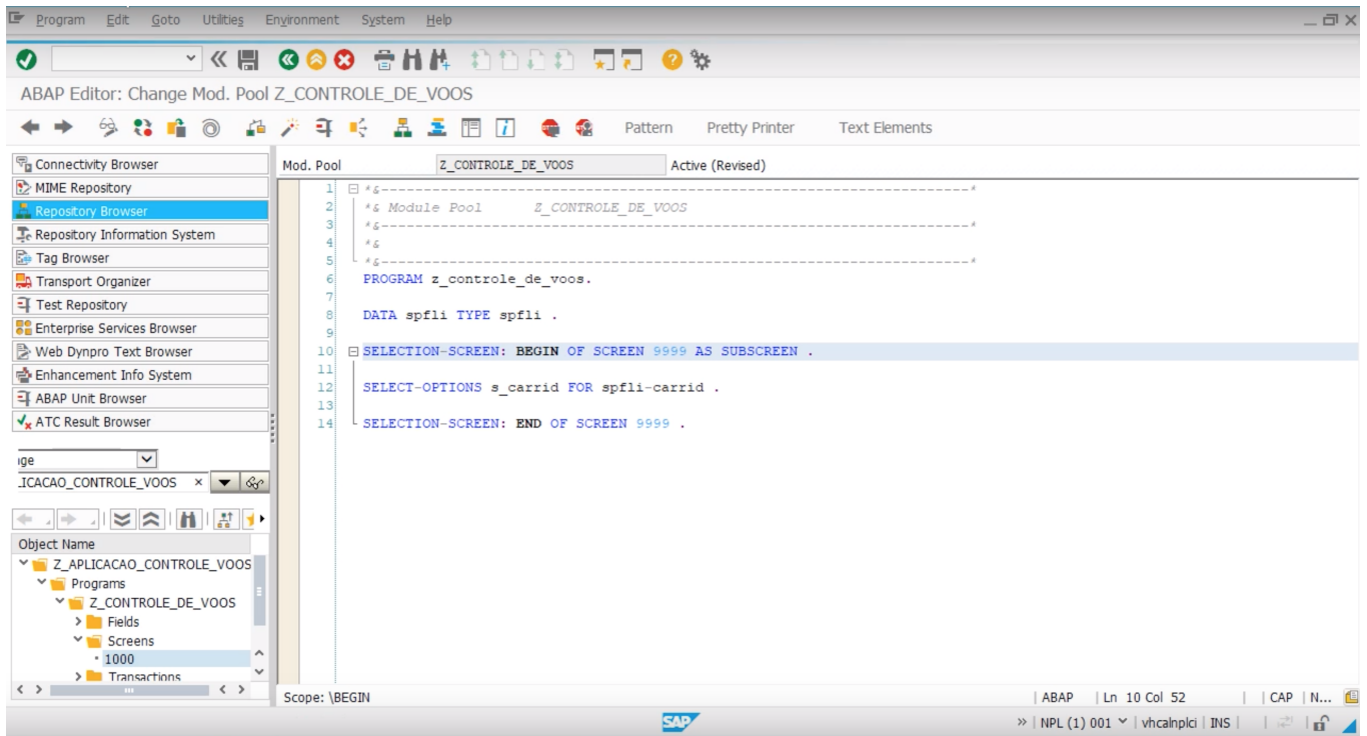
What happened?
Attribute 'Screen Type' with the characteristics 'Normal Screen' or 'Subscreen' determines how the screen is used. This error occurs if a normal screen is used as a subscreen, or vice versa.
Screen "Z_CONTROLE_DE_VOOS" 9999 therefore has the wrong type.

Error analysis
There is a programming error in the application.

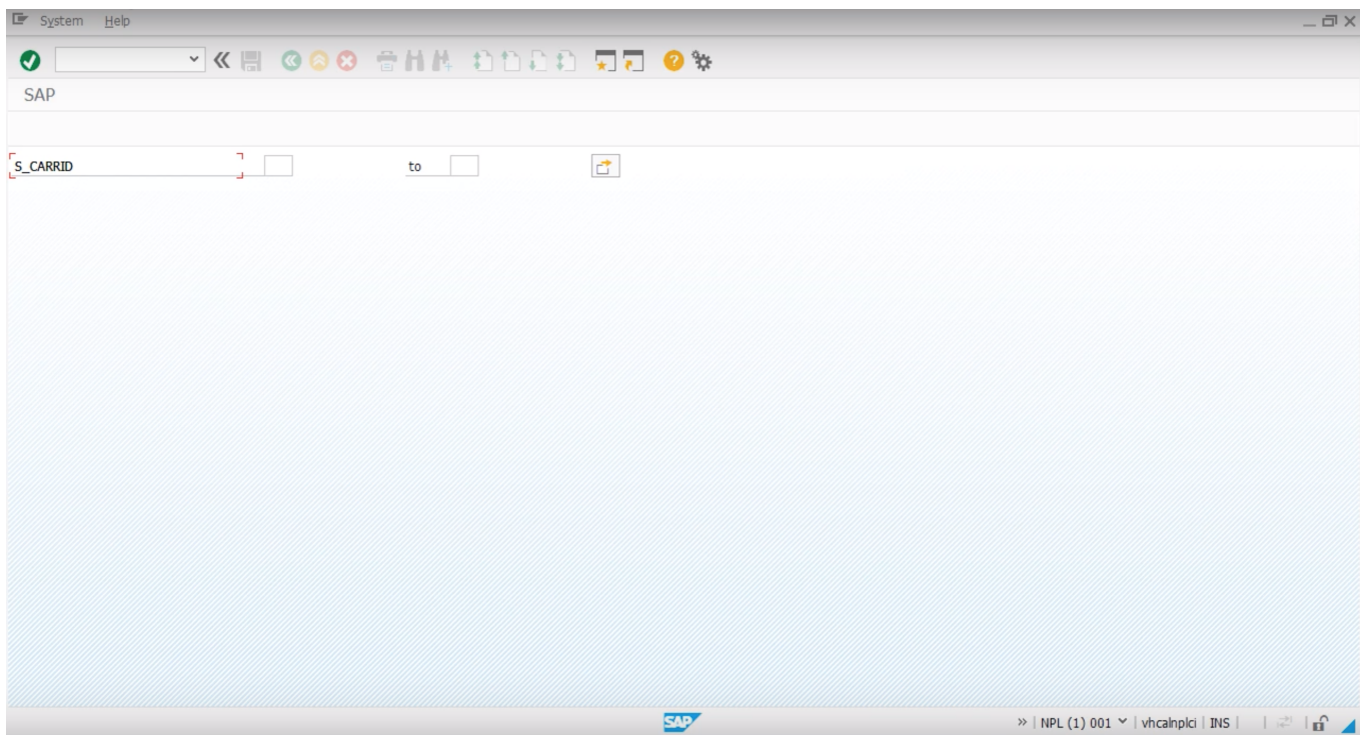
Trigger Location of Runtime Error

Program	Z_CONTROLE_DE_VOOS
Module Name	SYSTEM-EXIT

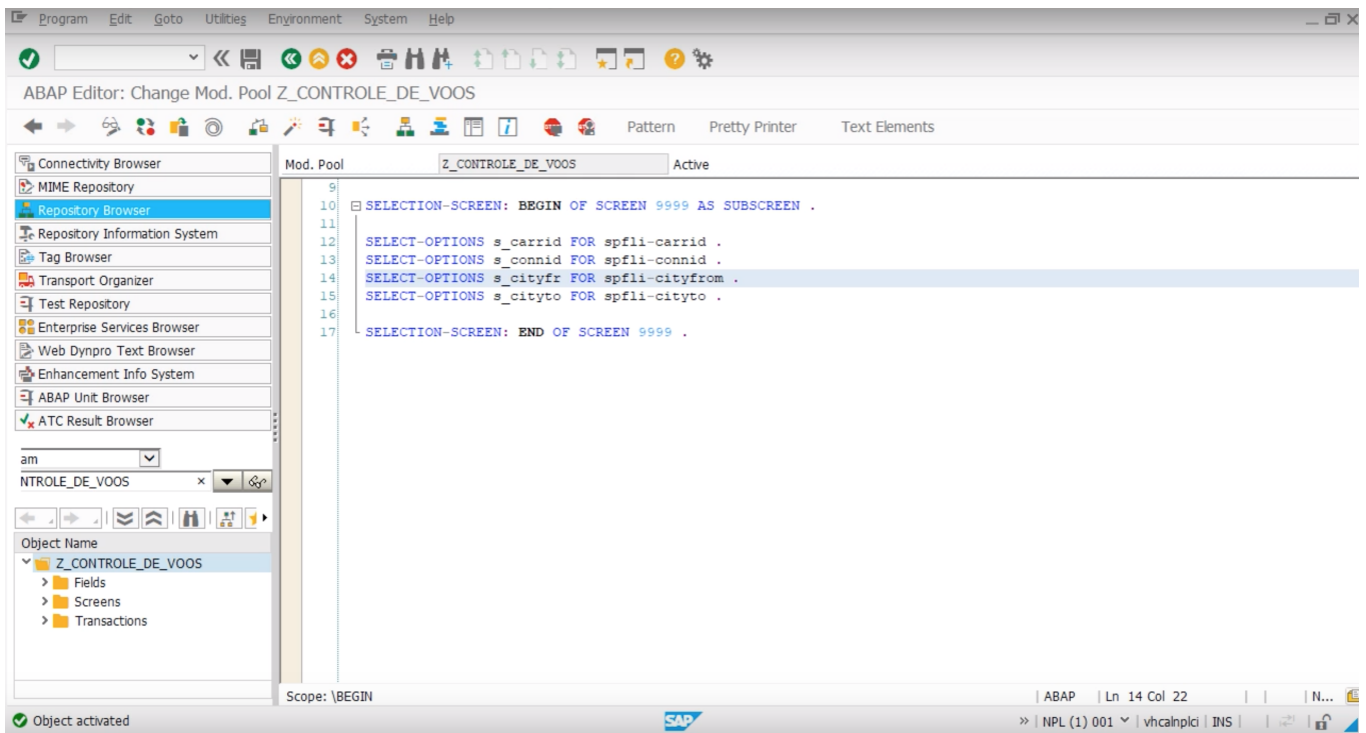
Feito isso, vamos executar a nossa aplicação.



Então, o que acontece é um dump. Esse dump ocorreu porque o compilador ABAP até encontrou a nossa Janela e o Subscreen, entretanto, a forma como declaramos está incorreta.

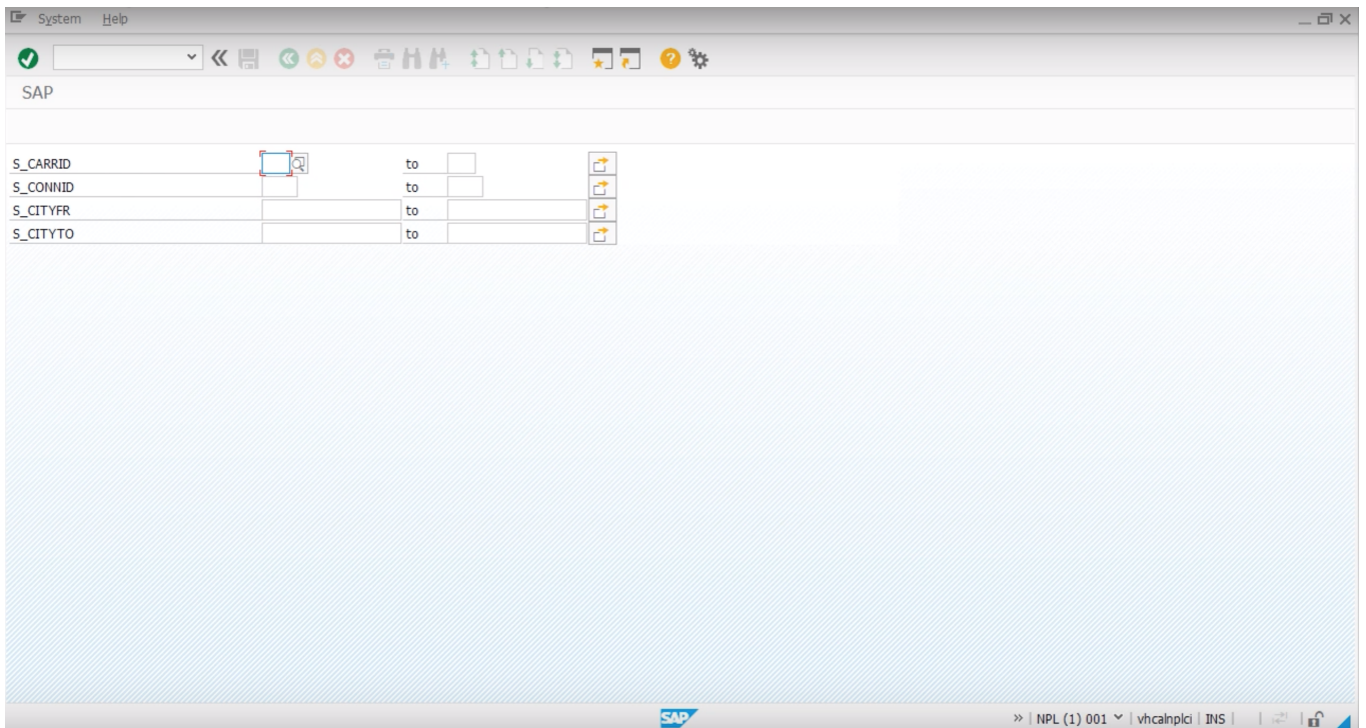


Após a alteração, vamos executar novamente a aplicação.

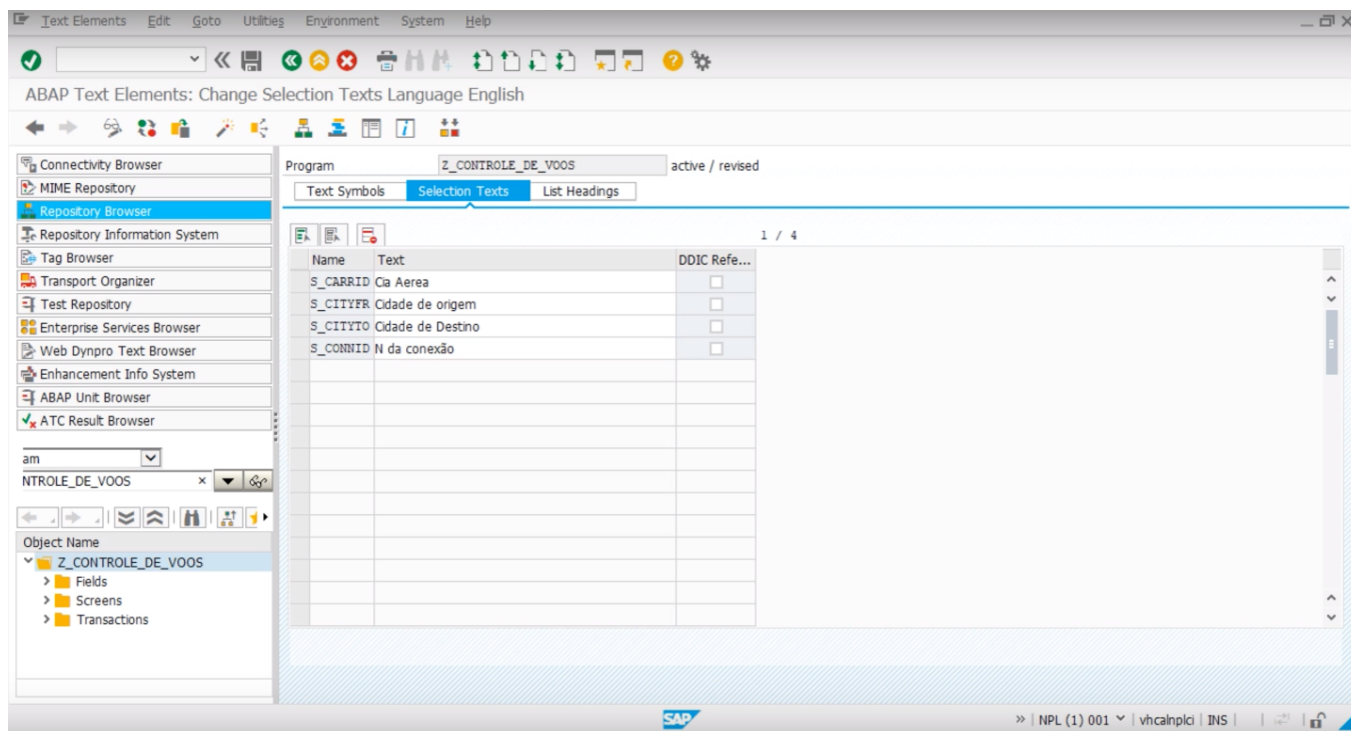


Com o nosso campo de seleção funcionando, vamos, então, inserir os outros campos e configurar os textos para que não apareça o nome técnico durante a execução da aplicação.

Campos declarados:

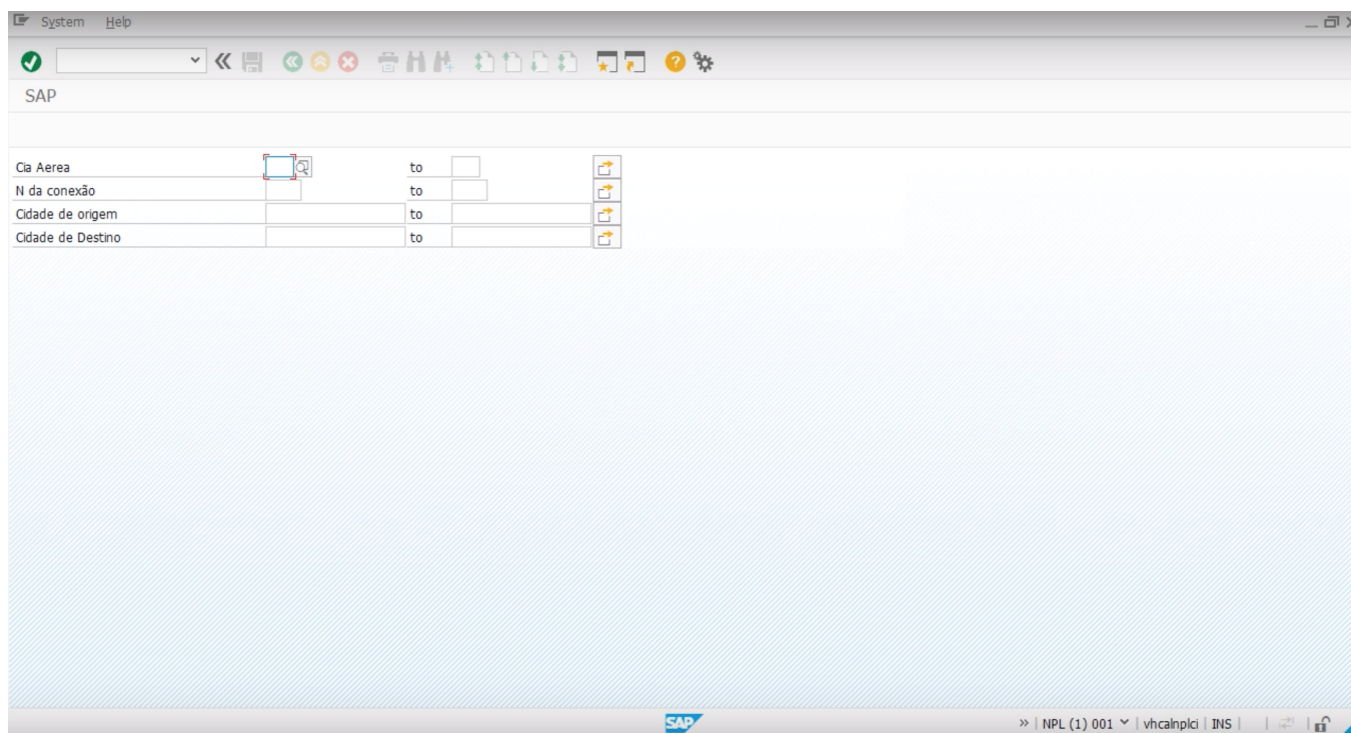


Vamos executar a aplicação novamente.

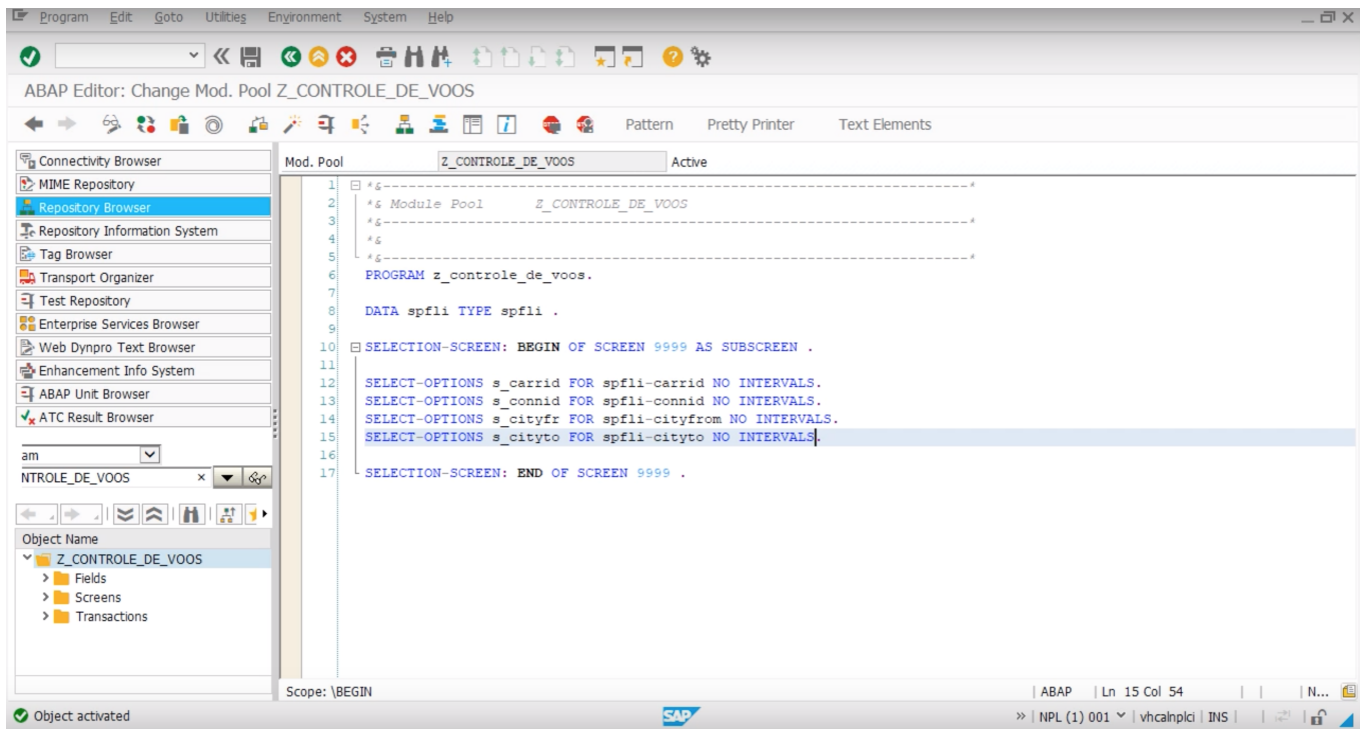


Todos os campos de seleção estão funcionando, então, vamos apenas ajustar os nomes dos campos.

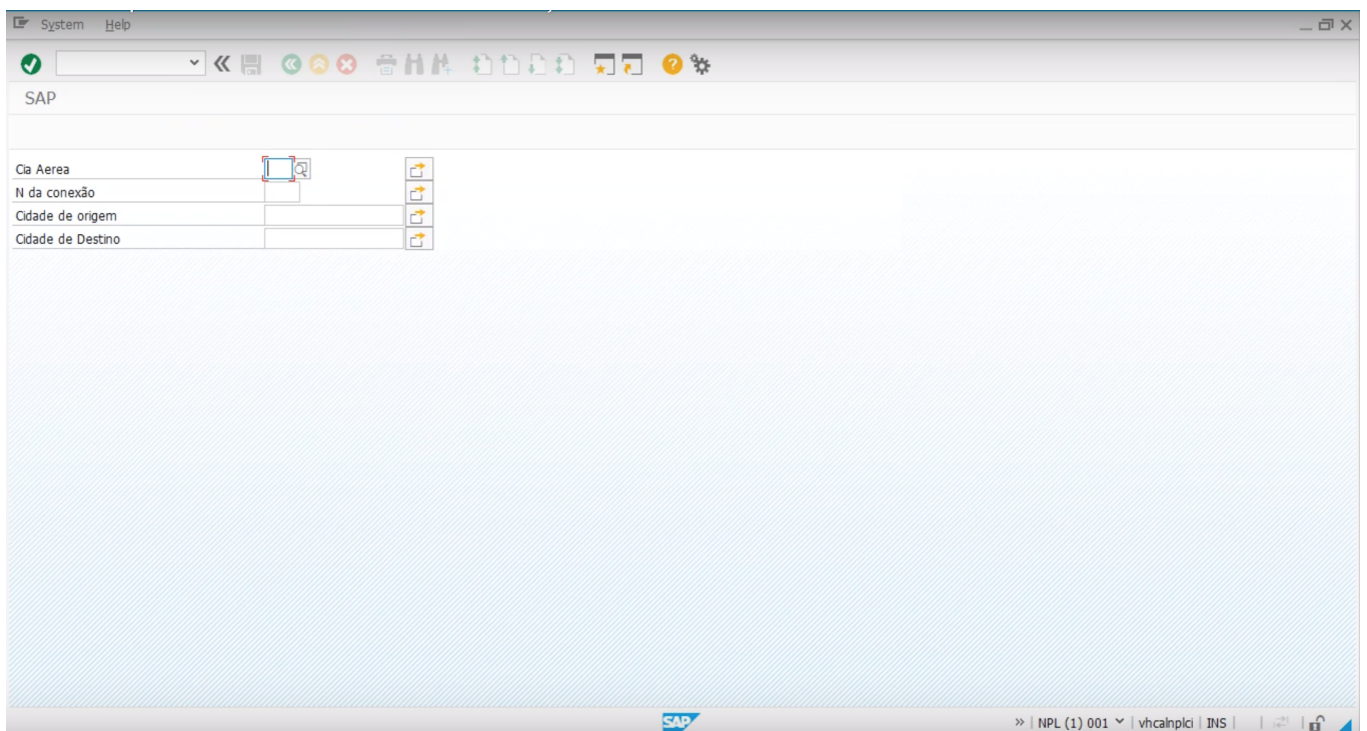
Para isso, clicamos em **Goto > Text Elements > Selection Texts**.



Executando a aplicação, temos a nossa tela com os nomes que informamos no passo anterior.



Uma melhoria que podemos implementar é referente à necessidade de campos DE...ATÉ da nossa tela de seleção. Então, após análise temos o seguinte:



E o resultado é o seguinte:

Código

```

*&-----*
*& Module Pool      Z_CONTROLE_DE_VOOS
*&
*&-----*
*&
*&
*&-----*
PROGRAM z_controle_de_voos

```

```
PROGRAM Z_CONTRATOS_DE_VOOS.
```

```
DATA spfli TYPE spfli .
```

```
SELECTION-SCREEN: BEGIN OF SCREEN 9999 AS SUBSCREEN.
```

```
SELECTION-SCREEN: BEGIN OF BLOCK blk1 WITH FRAME TITLE text-001 .
```

```
SELECT-OPTIONS: s_carrid FOR spfli-carrid NO INTERVALS.
```

```
SELECT-OPTIONS: s_connid FOR spfli-connid NO INTERVALS.
```

```
SELECT-OPTIONS: s_cityfr FOR spfli-cityfrom NO INTERVALS.
```

```
SELECT-OPTIONS: s_cityto FOR spfli-cityto NO INTERVALS.
```

```
SELECTION-SCREEN: END OF BLOCK blk1.
```

```
SELECTION-SCREEN: END OF SCREEN 9999.
```