

Workshop Abap – recursos adicionais

A lista abaixo contém alguns recursos do SAP que usamos em programas abap.

O conhecimento destes recursos economiza muito trabalho de codificação. Muitas vezes criamos programas e tabelas para fazer algo já existente no SAP.

Atenção! Este roteiro faz parte do Workshop de Abap, não é um material de auto-estudo. As descrições abaixo são somente uma visão geral de alguns recursos adicionais.

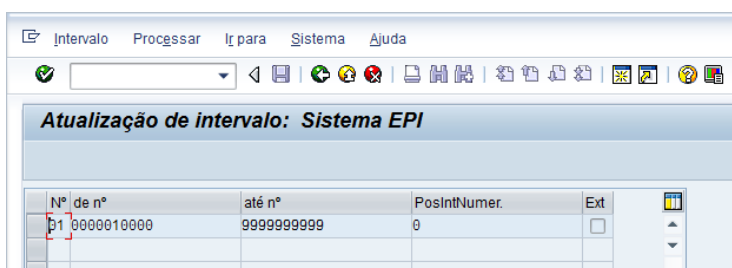
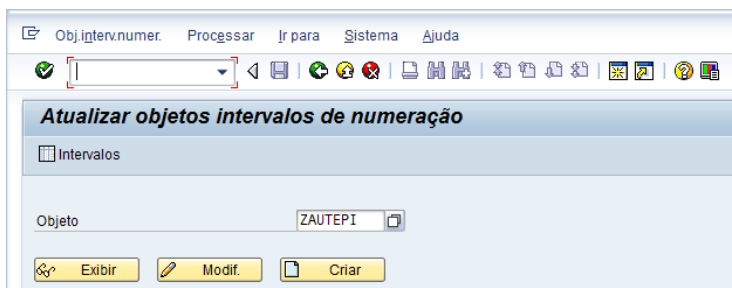
1 – Intervalo de numeração

Para que serve: criação de números únicos para qualquer tipo de documento no SAP, incluindo campos de tabelas Z.

Uso: na criação de documentos e dados mestres, temos a opção (através de configuração) de geração automática de numeração. Isto é feito através do “Intervalo de numeração”: o SAP garante que, a cada “chamada” do intervalo de numeração, tenhamos um número único. Por exemplo, numeração automática de Fornecedores ou Ordens de manutenção. Portanto, não crie programas e tabelas para “pegar o último número e somar 1”. O SAP tem um controle eficiente sobre os números gerados, incluindo a garantia de numeração única quando do acesso simultâneo de vários usuários ou processos.

Como fazer:

- criar o intervalo e subintervalo com a transação SNRO;
- chamar o módulo de função NUMBER_GET_NEXT para gerar o próximo número.



2 – Comparação de programas

Para que serve: comparar o código fonte de dois programas e apontar as diferenças.

Uso: comparação de programas fontes de nomes diferentes, para verificar as alterações que foram feitas ou a fazer. Os códigos do programa estão em arquivos textos e/ou tem nomes diferentes no ambiente.

Como fazer:

- Acionar a transação SE39;
- Informar o nome dos dois programas e exibi-los. Em seguida clicar o botão de comparação.

The screenshot shows the 'ABAP editor de tela dividida: tela' window with the title 'Compara  o via sistemas'. It contains two panels for comparing sources. The left panel, labeled '  esquerda', has 'Programa' selected with the value 'ZPROGRAMA1'. The right panel, labeled '  direita', has 'Programa' selected with the value 'ZPROGRAMA2'. Both panels have radio buttons for 'Default', 'Ativo', and 'Inativa', with 'Default' being selected in both. Below the selection options are several unchecked radio buttons: 'M todo', 'Defini  o de classe local/tipos', 'Makros', 'Implementa   es de classe locais', and 'Lokale Testklassen' (on the left) or 'Classes teste locs.' (on the right). At the bottom, there are two buttons: 'Exibir' (with a magnifying glass icon) and 'Modificar' (with a pencil icon).

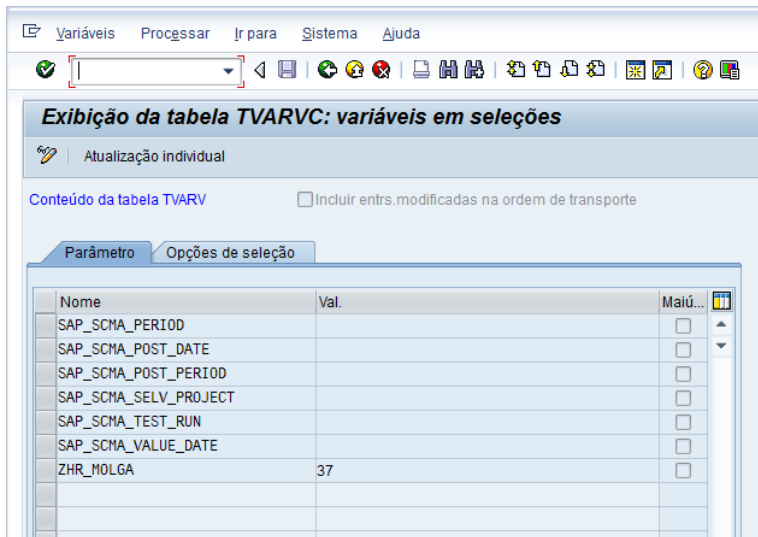
3 – Armazenamento de par metros

Para que serve: guardar valores de vari veis ou faixa de valores, sem necessitar criar tabela Z.

Uso: manter par metros diversos para programas Z ou enhancement-points, que n o sejam alterados com frequ ncia: datas, valores v lidos ou default de algum campo de tela. Estes valores podem ser mantidos em request e, desta forma, o SAP mant m registro de altera  es.

Como fazer:

- Criar a(s) vari vel(is) na transa   o STVARV;
- Ler a tabela TVARVC para recuperar o(s) valor(es).



4 – Armazenamento de documentos no SAP

Para que serve: armazenar qualquer tipo de arquivo (anexo) e ligá-lo a um valor, para recuperação posterior.

Uso: anexar qualquer arquivo (documento, planilha, pdf) e associá-lo a um valor no SAP.

Exemplos: armazenar o certificado de conclusão de curso a um número de matrícula, armazenar um laudo de manutenção de equipamento a uma ordem de manutenção.

Como fazer:

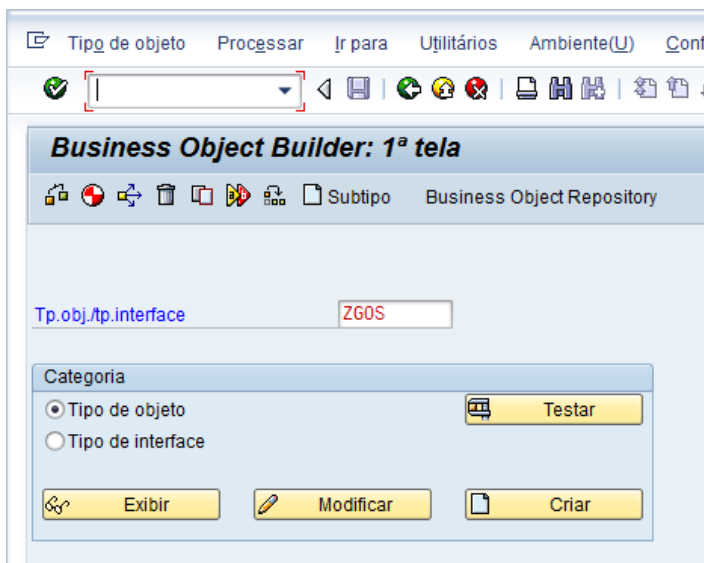
a) Criar um objeto para o valor – transação SWO1

b) Exemplo de codificação:

<http://www.sdn.sap.com/irj/scn/go/portal/prtroot/docs/library/uuid/0e6b0d95-0a01-0010-4696-ca0a48de5fb3?QuickLink=index&overridelayout=true&5003637734936>

c) Outras informações:

<http://wiki.scn.sap.com/wiki/display/SI/Frequently+asked+questions+about+Generic+Object+Services+GOS>



5 – Download de código fonte

Para que serve: guardar em arquivo texto o código abap.

Uso: manter backup local de código fonte abap, elementos de texto e telas.

Como fazer:

- a) Rodar o programa REPTRAN;
- b) Indicar o(s) nome(s) do(s) programa(s) e diretório de destino (desmarcar checkbox “Só código abap”).

Transportador de pool de módulos e de reports

Nome progr. até

Criado por até

último modificador até

Pacote até

Tipo do programa até

Status até

Data de criação até

Data de modificação até

Caminho do diretório

☐ Exibir
☒ Gravar em files
☐ Gravar em files e exibir

☐ Só código ABAP

6 – Debug em janela popup

Para que serve: disparar o debug quando uma janela popup está ativa.

Uso: quando uma janela popup está ativa, não temos acesso ao campo de comando para digitar /H.

Como fazer:

- a) Crie um arquivo texto com exatamente o conteúdo abaixo:

```
[Function]
Command=/H
Title=Title
Type=SystemCommand
```

- b) Deixe o Windows Explorer aberto. Quando a janela popup aparecer, arraste o arquivo para cima da janela e o debug será ligado.

7 – LSMW - Legacy System Migration Workbench (LTMC no S/4 Hana)

Para que serve: carga de dados.

Uso: ler arquivo texto, fazer pequenas mudanças em campos e importar no SAP, via SHDB, BAPI Ou IDoc.

Como fazer:

- Rodar a transação LSMW
- Informar projeto, subprojeto e objeto
- Montar estruturas de entrada, leitura de dados, conversão e importação



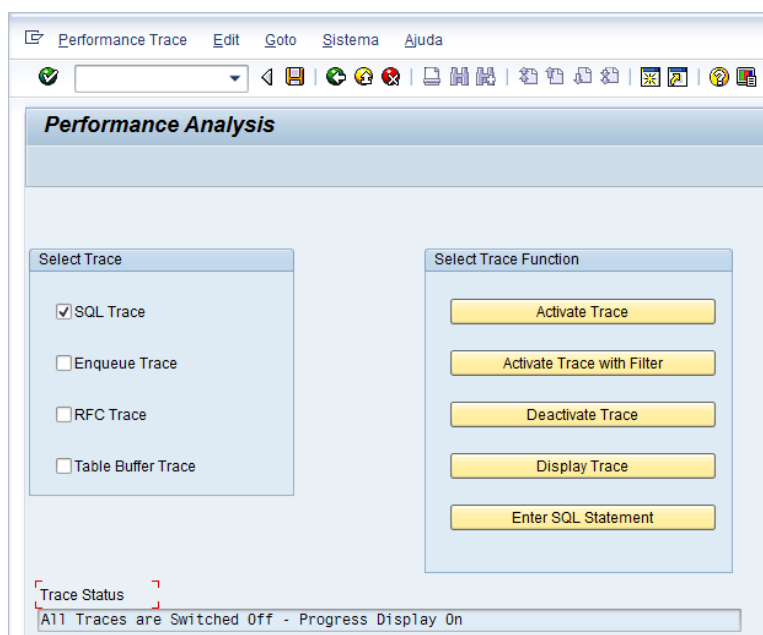
8 – Trace

Para que serve: rastrear operações de banco de dados e RFCs.

Uso: saber quais tabelas uma transação acessou. Muito útil para pesquisar onde as informações de uma transação standard são lidas ou gravadas.

Como fazer:

- Acionar a transação ST05 e clicar no botão “Activate Trace”
- Rodar a transação ou programa desejado em outra janela;
- Voltar a janela da transação ST05 e clicar no botão “Deactivate Trace”;
- Clicar no botão “Display Trace”.



9 – Queries

Para que serve: construir relatórios baseados em relacionamentos de tabelas.

Uso: relatórios simples cujas colunas são oriundas diretamente de campos de tabelas, sem processamento adicional.

Como fazer:

- a) Transação SQ03: criar grupo de usuário;
- b) Transação SQ02: criar infoset. O InfoSet é a área de associação de tabelas (join) e seleção dos campos;
- c) Transação SQ01: criar a query a partir de um grupo de usuários e infoset;
- d) Além das 3 transações acima, podemos usar:
 - a. Transação SQVI para geração rápida de queries;
 - b. Transação FGPR: report painter e report writer.

10 – Comparação do conteúdo de tabelas

Para que serve: comparar o conteúdo de tabelas entre instâncias.

Uso: exibir e/ou atualizar o conteúdo de tabelas.

Como fazer:

- a) Transação SCMP;
- b) Informar a tabela e a conexão SAP de destino.

11 – Caixa de texto

Para que serve: digitação de texto livre em telas abap.

Uso: permitir texto livre em telas de programa on-line.

Como fazer:

- a) Criar contêiner na tela desejada;
- b) Usar a classe `cl_gui_textedit`;
- c) Usar os módulos de função `READ_TEXT` e `SAVE_TEXT` para ler e armazenar os textos.

12 – Pesquisa de strings em programas

Para que serve: buscar qualquer string em código abap.

Uso: pesquisar uma cadeia de texto qualquer, principalmente comentários.

Como fazer:

- a) Transação EWK1;
- b) Informar o(s) programa(s) e a string a buscar.

Temos também o report `RS_ABAP_SOURCE_SCAN`, com mais opções.

13 – FTP no SAP

Para que serve: FTP de arquivos no servidor SAP.

Uso: copiar arquivos entre a estação e o servidor SAP.

Como fazer:

- a) Transação CG3Y: download de arquivos (servidor SAP → estação de trabalho);
- b) Transação CG3Z: upload de arquivos (estação de trabalho → servidor SAP).

14 – Acesso direto ao banco de dados

Para que serve: leitura e gravação direto no banco de dados, sem passar pelo Open SQL do abap.

Uso: execução de comandos nativos do banco de dados instalado (Oracle, MS-SQL, MaxDB, SYBASE), seja da base SAP ou outras tabelas fora do SAP. Normalmente usado para leitura/gravação em sistemas não-SAP.

Como fazer:

- a) Transação DBCO: criar a conexão SAP ↔ Banco de dados (atribuição de basis);
- b) No código abap, o comando SQL nativo do banco DEVE estar entre um EXEC SQL e ENDEXEC.

15 – Registro de log de operações

Para que serve: registrar logs diversos em programas abap.

Uso: gravar informações de status de qualquer procedimento em programas abap. Exemplos: erros em qualquer processamento, fases de processo e quaisquer outras mensagens, sem a necessidade criação de tabelas Z para armazenamento.

Como fazer:

- a) Transação SLG0: criar o objeto de log;
- b) Módulos de função: BAL_LOG_CREATE → criar o objeto no programa;
BAL_LOG_MSG_ADD → adicionar mensagem;
- c) Transação SLG1: consultar as mensagens armazenadas.

16 – Caracteres especiais em programas abap

Para que serve: pesquisa ou gravações em cadeia de caracteres.

Uso: verificar se existe caracter especial (tabulação, fim de linha, backspace, dentre outros) em arquivos texto, normalmente do Unix. Alguns sistemas legados definem que os arquivos lidos e/ou gravados devem conter estes caracteres.

Como fazer:

- a) Usar os atributos estáticos da classe CL_ABAP_CHAR_UTILITIES;
- b) Dentro do programa testar pelo atributo, por exemplo: IF V_LINHA+3(1) = CL_ABAP_CHAR_UTILITIES=> HORIZONTAL_TAB .

17 – Sites úteis, com muitas dicas

- a) <https://www.zapyard.com/>
- b) <http://www.abapzombie.com>
- c) <http://abap101.com>
- d) <http://zevolving.com/>
- e) <http://abapinho.com/>

18 – Tabelas interessantes

- a) DD02L - tabelas SAP, DD03L – campos de tabelas, DD01L – domínios, DD27M – visão elementos de dados/textos/domínios;
- b) DD03M – visão com campos de tabelas, seus elementos dados, textos e domínios;
- c) E070/E071/TADIR: requests e seus objetos;
- d) TFDIR/TFTIT: módulos de função;
- e) ICON: ícones;
- f) TSTC/TSTCP: transações e seus parâmetros;
- g) VARI/VARID: variantes de report;
- h) NRIV: intervalo de numeração;
- i) REPOSRC: Fontes de report.

19 – Módulos de função úteis

- 1) REUSE_ALV_POPUP_TO_SELECT e K_KKB_LIST_FOR_ITEM_SELECTION: lista conteúdo de tabela interna em janela popup;
- 2) OIL_MONTH_GET_FIRST_LAST / HR_HCP_GET_LAST_DAY_OF_MONTH / HR_JP_MONTH_BEGIN_END_DATE: retorna o primeiro e ultimo dia de um mês;
- 3) RFC_READ_TABLE: lê campos de uma tabela em outro mandante;
- 4) SWE_CD_TST_DELAY: gerar um atraso no processamento, em milissegundos;
- 5) OIL_DATE_SUBTRACT_MONTH: subtrair uma quantidade de meses em uma data;
- 6) TMP_GUI_BROWSE_FOR_FOLDER: busca diretório (SAPGUI);
- 7) RZL_READ_DIR_LOCAL: retorna lista de arquivos do servidor SAP;
- 8) KD_GET_FILENAME_ON_F4: popup para seleção de arquivo (usar no AT SELECTION-SCREEN ON VALUE-REQUEST FOR campo);
- 9) FIEB_PASSWORD_ENCRYPT / FIEB_PASSWORD_DECRYPT: encriptar / decriptar cadeias de caracteres;
- 10) F4_CONV_SELOPT_TO_WHERECLAUSE: converte RANGE / SELECT-OPTIONS em condição para a cláusula WHERE;
- 11) SF_SPECIALCHAR_DELETE / SCP_REPLACE_STRANGE_CHARS: elimina caracteres especiais de uma cadeia de caracteres;
- 12) COPO_POPUP_TO_DISPLAY_TEXTLIST: exibe uma tabela interna na tela;
- 13) THUSRINFO: lista os usuários logados;
- 14) SO_SPLIT_FILE_AND_PATH / SPLIT_FILENAME: separa o nome do arquivo do diretório;
- 15) RS_VARIANT_CONTENTS: lê o conteúdo de uma variante em um report;
- 16) JOB_OPEN / JOB_CLOSE: cria job através de programa abap;
- 17) BKK_GET_MONTH_LASTDAY: último dia do mês;
- 18) RS_REFRESH_FROM_SELECTOPTIONS: lê todos os parâmetros da tela atual;
- 19) IMPORT_DYNPRO: obtém todos os elementos de uma tela;
- 20) SYSTEM_CALLSTACK: busca a pilha de chamadas de programas, funções e rotinas;
- 21) HR_TIME_AND_DURATION / SD_CALC_DURATION_FROM_DATETIME / COPF_DETERMINE_DURATION: calcula diferença entre dois valores de horas;
- 22) POPUP_TO_SELECT_MONTH: popup para selecionar mês e ano;
- 23) SPELL_AMOUNT: retorna por extenso um valor monetário