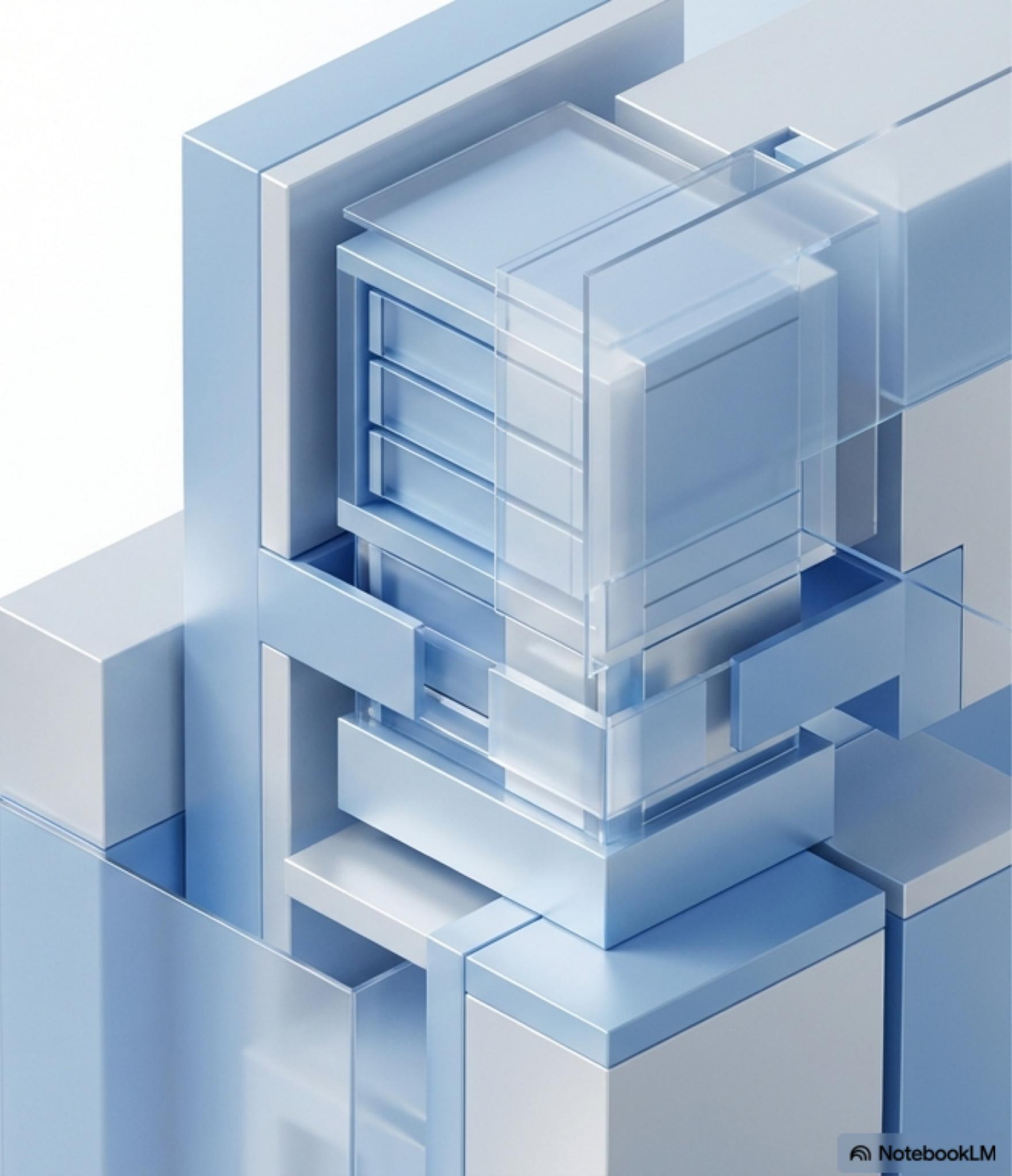


Lendo Dados do Banco de Dados (ABAP SQL)

Programação ABAP Básica: Sintaxe
Moderna e Melhores Práticas



Objetivos de Aprendizagem



Sintaxe Robusta: Escrever comandos SELECT utilizando a sintaxe moderna (Strict Mode).



Host Variables: Compreender o uso do caractere de escape @ para diferenciar variáveis ABAP de colunas SQL.



Declarações Inline: Utilizar @DATA para criação automática de tabelas e estruturas.



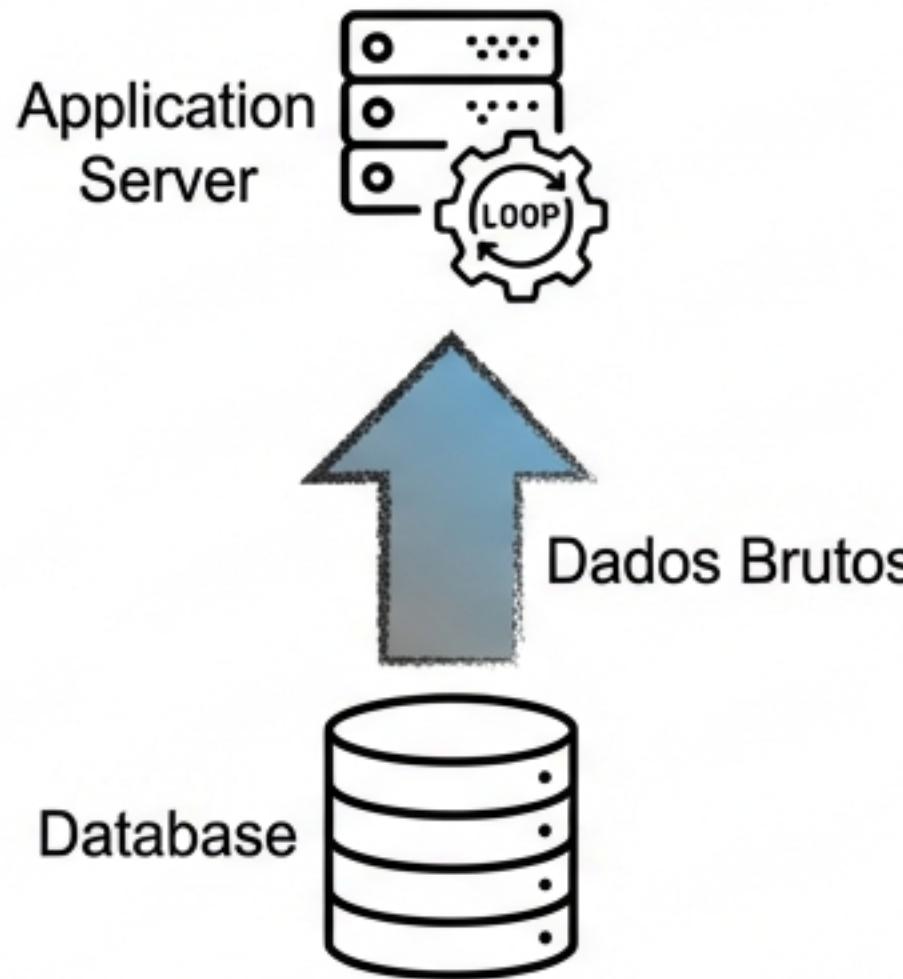
Filtros e Ordenação: Aplicar WHERE, ORDER BY e limitação de linhas (UP TO n ROWS) eficientemente.



Leitura de Dados: Diferenciar estratégias de SELECT SINGLE vs. UP TO 1 ROWS e monitorar sy-subrc.

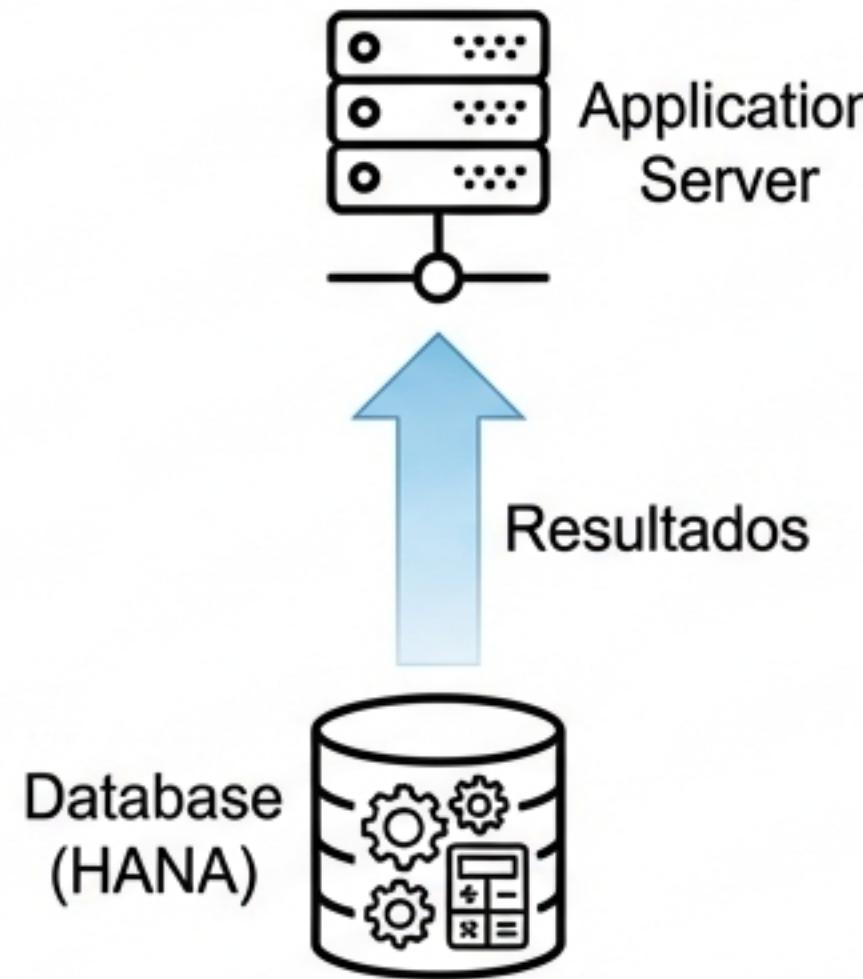
A Evolução: Do ‘Open SQL’ ao ABAP SQL Moderno

O Passado (Open SQL Clássico)



Dados ao Código: Tráfego pesado, processamento na aplicação.

O Futuro (ABAP SQL Moderno / HANA)



Code Pushdown: A lógica é ‘empurrada’ para o banco. Retorna apenas o necessário.

O Mudança de Paradigma: Com a chegada do SAP HANA e ABAP 7.40+, o foco mudou para Code Pushdown.

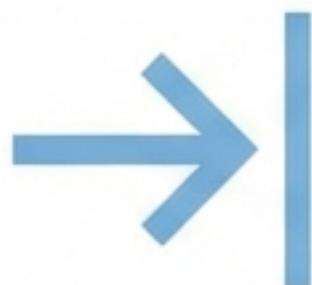
O 'Strict Mode' (Modo Estrito)

Regras ativadas automaticamente pelo compilador ao usar sintaxe moderna.



1. Vírgulas Obrigatórias

Ao contrário do estilo antigo (espaços), os campos devem ser separados por vírgula na lista de seleção.



2. INTO no Final

A cláusula de destino (INTO) deve vir após as cláusulas de seleção, filtro (WHERE) e ordenação (ORDER BY).



3. Escape (@) Obrigatório

Toda variável do programa ABAP usada na query deve ter o prefixo @ para diferenciar de colunas.

Anatomia do SELECT Moderno

```
1  SELECT FROM /dmo/connection  
2    FIELDS •                                Lista de Colunas (obrigatório vírgulas)  
3      carrier_id,  
4      connection_id,  
5      airport_from AS departure_airport, — ALIAS: Renomeação na projeção  
6      distance  
7  WHERE carrier_id = @lv_airline •          Host Variable (Filtro)  
8  ORDER BY carrier_id, connection_id DESCENDING  
9  INTO TABLE @DATA(lt_flights).
```

• • Destino no final do comando



Lendo uma Única Linha: Precisão vs. Amostragem

SELECT SINGLE

Leitura por Chave

- **Uso:** Quando se tem a chave primária completa.
- **Objetivo:** Validar existência ou ler atributos específicos.

```
WHERE carrier_id = 'AA'  
AND connection_id = '0017'
```

UP TO 1 ROWS

Amostragem

- **Uso:** Quando NÃO se tem a chave completa.
- **Objetivo:** Buscar 'o primeiro encontrado' ou o mais recente/antigo.

```
ORDER BY distance ASCENDING  
UP TO 1 ROWS
```

O Poder do '@' (Host Variables)



A Distinção: O parser SQL precisa diferenciar colunas da tabela de variáveis na memória.

Segurança & Performance: O uso do @ evita ambiguidades e permite que o banco reproveite o plano de execução.



WHERE carrier_id = @lv_id;



WHERE carrier_id = lv_id (Erro de Sintaxe no Strict Mode)

O Painel de Controle (Variáveis de Sistema)

sy-subrc (Return Code)



0 Sucesso (Dados encontrados)



4 Aviso (Nenhum dado atende ao filtro)



8 Erro

Dica: Verificar imediatamente após o SELECT.

sy-dbcnt (Database Count)



- Contém o número de linhas processadas.
- Fundamental para validação de volume e relatórios.
- Se sy-subrc = 0, mostra quantos registros retornaram.

Exemplo Prático: Code Walkthrough

```
1  " 1. Preparação
2    DATA(lv_carrier) = 'UA'.
3    DATA(lv_min_dist) = 1000.
4
5  " 2. Consulta
6  SELECT FROM /dmo/connection
7    FIELDS carrier_id, distance, distance_unit
8    WHERE carrier_id = @lv_carrier
9      AND distance > @lv_min_dist
10   ORDER BY distance DESCENDING
11   INTO TABLE @DATA(lt_results).
12
13 " 3. Validação
14 IF sy-subrc = 0.
15   out->write( |Encontrados: { sy-dbcnt }| ).
16 ELSE.
17   out->write( 'Nenhum voo encontrado.' ).
18 ENDIF.
```

Definição de variáveis de filtro

Filtro composto e ordenação decrescente

Verificação obrigatória do status

Glossário Técnico

ABAP SQL

Abstração independente de banco (antigo Open SQL). Converte ABAP para SQL nativo.

FIELDS

Palavra-chave para listar colunas explicitamente.

Host Variable (@)

Variável ABAP usada dentro do SQL. Obrigatória no Strict Mode.

Inline Declaration

Criação de dados (@DATA) em tempo de execução inferindo tipos.

Code Pushdown

Mover lógica de processamento para o banco de dados para ganho de performance.

Strict Mode

Regras de sintaxe ativadas automaticamente ao usar recursos modernos.

Pontos de Atenção e Melhores Práticas

Prática Ruim (Legado)	Prática Boa (Moderno)
 SELECT * Economia de memória e rede.	 SELECT field1, field2 Economia de memória e rede.
 SELECT ... UP TO 1 ROWS (para checar existência) Performance otimizada.	 SELECT SINGLE @abap_true Performance otimizada.
 SELECT dentro de LOOP Evita múltiplas idas ao banco.	 FOR ALL ENTRIES ou JOIN Evita múltiplas idas ao banco.
 WHERE id = lv_id Segurança e Clareza (Strict Mode).	 WHERE id = @lv_id Segurança e Clareza (Strict Mode).

Quiz de Fixação

1.

O que acontece se esquecer o símbolo @ antes de uma variável no Strict Mode?

2.

Por que devemos evitar o uso de SELECT * em aplicações de alta performance?

3.

Qual a diferença fundamental entre sy-subrc e sy-dbcnt?

4.

Quando é preferível usar SELECT SINGLE em vez de UP TO 1 ROWS?

Escreva código para o futuro.

Limpo, eficiente e otimizado para HANA.

**Revise seus programas antigos hoje mesmo:
substitua SELECT * por listas de campos e aplique o
Strict Mode para maior segurança e performance.**