

SQL

SELECT – parte 1



INFORMÁTICA
1º ANO – 2023/2024

ARTUR JORGE DA SILVA ROCHA
E-MAIL: D001457@UMAIA.PT

SELECT

2

- SELECT é a instrução usada para extrair informação da base de dados
- Cláusulas mais importantes:
 - SELECT
 - FROM
 - WHERE
- Comando SELECT constitui uma pergunta à base de dados.
- Também denominada ‘query’ ou ‘consulta’

SELECT

3

- SELECT é seguido de uma lista de expressões, separadas por vírgulas
- Resultado do SELECT é uma série de colunas, uma por cada expressão na sua lista
- Expressões normalmente são nomes de campos em tabelas.
- Mas também podem ser: valores simples, expressões matemáticas, funções, outros SELECTs
- Exemplo: `SELECT 2, getdate(), (SELECT ((4+6)*2))`
- SELECT sem FROM é uma forma degenerada de SELECT

SELECT - FROM

4

- FROM indica a que tabela o SELECT vai buscar os seus dados
- Forma básica: **SELECT <lista de colunas> FROM <tabela>**
- Devolve uma linha por cada linha da tabela especificada no FROM
- Em cada linha, aparecem apenas os atributos especificados no SELECT

SELECT - FROM

5

- **Argumentos do SELECT - FROM:**
 - todos os anteriores
 - nomes de colunas de <tabela>
 - funções podem incluir nomes de colunas como argumentos
 - expressões com operadores que operem sobre colunas
 - * significa 'todos os campos da tabela'

SELECT - Exemplos

6

- Obter os dados de todos os motores usados na história da F1:
 - `SELECT * FROM motor`
- Obter todas as nacionalidades de pilotos que correram na F1
 - `SELECT pais FROM piloto`
- Obter o nome de todos os construtores e o respectivo país, mas numa coluna só
 - `SELECT nome + ' - ' + pais FROM construtor`

SELECT DISTINCT

7

- Resultados de um SELECT podem ser repetidos
- Ao contrário de uma tabela, uma query não é um conjunto de registos, mas sim um multi-conjunto
- Usa-se DISTINCT para obter apenas linhas distintas
- Exemplos:
 - Seleccionar os países de origem de todos os pilotos
 - ✦ SELECT pais from piloto
 - Seleccionar os países de origem de todos os pilotos, sem repetição
 - ✦ SELECT DISTINCT pais from piloto
 - Seleccionar todas as combinações distintas de país / último nome, de todos os pilotos
 - ✦ SELECT DISTINCT pais, unome from piloto

SELECT TOP

8

- TOP n permite seleccionar as primeiras n linhas da query
- Deve seguir-se ao DISTINCT mas vir antes de qualquer especificação de coluna no SELECT
- Exemplo:
 - Mostrar os primeiros cinco pilotos registados
 - ✦ `SELECT TOP 5 pnome, unome FROM piloto`

SELECT - AS

9

- É possível mudar o nome de uma coluna com a declaração 'AS'
- Segue-se a expressão respectiva do SELECT com AS <Nome>.
- Se o argumento do AS tiver espaços, deve ser rodeado de apóstrofos.
- Também se pode mudar o nome da tabela dentro da query. Não tem utilidade para o utilizador final, mas é útil para o programador
- Em qualquer dos casos abaixo, pode-se omitir o AS e deixar apenas o novo nome
- Exemplos:
 - `SELECT pnome AS Nome, unome AS Apelido FROM piloto`
 - `SELECT P.pnome Nome, P.unome Apelido FROM piloto AS P`

SELECT - FROM - WHERE

10

- Cláusula WHERE especifica condições para filtrar as linhas a mostrar
- Para cada linha na tabela, a expressão no WHERE é avaliada
- A linha só é incluída no resultado final, e processada de acordo com a cláusula SELECT, se satisfizer a condição no WHERE
- Condição do WHERE tem de ter valor lógico verdadeiro ou falso e pode ser composta de várias condições unidas com conectores lógicos (AND, OR, NOT)
- Condição aceita operadores relacionais, operador LIKE, teste IS [NOT] NULL, construção BETWEEN .. AND ...
- Aceita ainda outras construções a ver mais tarde...

SELECT - FROM – WHERE: Exemplo

11

- Mostrar os dados de todos os construtores ingleses
 - `select * from construtor where pais = 'uk'`
- Mostrar o nome, a nacionalidade e a idade de todos os pilotos franceses ou italianos que tenham mais de 30 anos
 - `SELECT pnome, unome, pais, year(getdate()) - anoNasc AS idade FROM piloto WHERE (pais = 'Fra' OR pais = 'Ita') AND (year(getdate()) - anoNasc > 30)`



idade

Operador BETWEEN

12

- Operador BETWEEN ... AND ... permite especificar um intervalo
- Exemplo: Mostrar todos os pilotos que tenham nascido entre 1950 e 1960
 - `SELECT * FROM piloto WHERE anonasc BETWEEN 1950 AND 1960`

Operador LIKE

13

- Operador LIKE permite especificar formatos de strings
- Formato: <string> LIKE <máscara> - devolve verdadeiro se a string corresponder à máscara especificada
- Admite caracteres especiais, marcadores de espaços:
 - `_` indica um carácter
 - `%` indica uma sequência de caracteres
- Mostrar os dados de todos os pilotos cujo último nome começa por 'P' e tem um 'S' na penúltima posição
 - `SELECT * FROM piloto where unome like 'p%s_'`

ORDER BY

14

- Os resultados de uma query podem ser ordenados por qualquer campo
- Ordenação especifica-se com cláusula ORDER BY seguida da coluna que será ordenada
- Ordenação faz-se por ordem alfabética ou crescente (ascendente)
- Para especificar ordem inversa adiciona-se DESC após a ordenação
- Podem-se especificar várias colunas para ordenação: esta é feita segundo a primeira coluna especificada, para valores iguais usa-se a segunda coluna para desempatar e assim sucessivamente. Para cada um pode-se especificar uma ordem diferente
- Exemplo:
 - mostrar todos os pilotos, ordenando por país em ordem anti-alfabética e depois por último nome por ordem alfabética
 - ✦ `SELECT * FROM piloto order by pais desc, unome asc`

Funções

15

- Há várias funções agrupadas em diversas categorias
- Numa instrução SELECT é comum usar funções sobre nomes de colunas
- Para cada linha da tabela em causa, a função é avaliada sobre o valor respectivo da coluna
- Função tem de ser apropriada para o tipo de coluna
- Qualquer função avaliada sobre *null* devolve o valor *null*

Exemplos de Funções: Texto

16

- **LEN(texto)** - comprimento de texto
- **LOWER(texto)** - devolve texto em minúsculas
- **UPPER(texto)** - devolve texto em maiúsculas
- **LTRIM(texto)** - elimina espaços em branco no início de texto
- **RTRIM(texto)** - elimina espaços em branco no fim de texto
- **SUBSTRING(texto, pos_ini, tamanho)** - devolve a sub-cadeia de caracteres de texto que começa na posição indicada e tem o comprimento especificado

Exemplos de Funções: Data e Hora

17

- **DAY(data)** - devolve o dia de uma data
- **MONTH(data)** - devolve o mês de uma data
- **YEAR(data)** - devolve o ano de uma data
- **GETDATE()** - devolve a data actual
- **DATEADD(parte, número, data)** - adiciona um número a parte de uma data
- **DATEDIFF(parte, data inicial, data final)** - devolve a diferença entre duas datas, tomando como unidade a parte especificada
 - Formatos de datas: usar formato ISO 8601 ou Unseparated String Format

Exemplos de Funções: Matemáticas

18

- **ABS(número)** - devolve o valor absoluto de número
- **CEILING(número)** - devolve o menor inteiro que é maior ou igual que número
- **FLOOR(número)** - devolve o maior inteiro que é menor ou igual que número
- **PI()** - devolve o valor de π
- **RAND()** - devolve um número aleatório entre 0 e 1. Tem um argumento opcional que serve de semente
- **ROUND(número, precisão, truncar)** - arredonda número para o número de casas decimais indicadas por precisão indicada. Se o terceiro argumento for 1, o resultado é truncado (aproximado por defeito), caso contrário é arredondado para o valor mais próximo
- **SIGN(número)** - retorna o sinal de um número (positivo, negativo ou zero)

Exemplos de Funções: Genéricas

19

- **CASE expressão WHEN condição THEN resultado [...]** - pode ter uma série de condições sucessivas. Apresenta o resultado da primeira condição que se verificar.
- **CAST(expressão as tipo)** - converte a expressão para o tipo indicado
- **ISNULL(expressão, valor)** - retorna a expressão se esta não for null ou valor, no caso contrário
- **ISNUMERIC(expressão)** - retorna 1 se a expressão for numérica, 0 se não for

Exemplos

20

- Listar o nome dos pilotos numa única coluna, indicando apenas as três primeiras letras do primeiro nome seguidas por um ponto e o segundo nome todo em maiúsculas
 - `SELECT substring(pnome,1,3) +'. ' + UPPER(unome) from piloto`
- Obter a idade dos pilotos registados
 - `SELECT year(getdate()) - anoNasc FROM piloto`

Exemplo com Condições múltiplas

21

```
select pnome + ' ' + unome, 'nacionalidade' =  
  case  
    when pais = 'BRA' THEN 'Brasileiro'  
    when pais = 'ARG' THEN 'Argentino'  
    else 'Outro'  
  end  
, 'idade' =  
  case  
    when anoNasc < 1950 then 'Veterano'  
    when anoNasc >= 1950 and anoNasc < 1980 then  
      cast(year(getdate()) - anoNasc as nvarchar)  
    else '...'  
  end  
from piloto
```

Exercícios

22

- Criar as tabelas necessárias para representar na base de dados o conceito de patrocinador. Indicar o seu nome do patrocinador, as equipas que patrocina em cada ano, e o orçamento dado em cada ano para cada equipa. Uma equipa pode ter vários patrocinadores, mas por vezes o mesmo patrocinador pode financiar várias equipas (por exemplo, a Marlboro já patrocinou Ferrari e McLaren no mesmo ano).
- Criar as tabelas necessárias para representar vários recordes. Deve ser possível indicar para cada pista o recorde da melhor volta em corrida, da melhor volta em qualificação, do número de voltas na liderança, do número total de vitórias nesse circuito e do número total de pole-positions. O registo do recorde deve incluir o nome do piloto, a data em que o recorde foi alcançado, e o valor desse recorde. Quando um recorde for batido por um novo piloto, o novo registo deve ser adicionado à base de dados mas sem apagar os anteriores.

Exercícios

23

- obter a idade e o último nome de todos os pilotos ingleses nascidos na primeira metade do século XX
- obter as nacionalidades (sem repetições) de todas as mulheres que já correram na F1
- para cada construtor inglês, apresentar o seu nome e uma coluna do tipo '2 décadas', indicando há quantos anos foi a fundação. Este valor deve ser indicado em 'décadas' e arredondado por defeito.

Exercicios

24

- Mostrar todos os apelidos diferentes dos pilotos de F1, ordenando de forma crescente segundo o número de letras de cada nome
- Mostrar o nome e a idade de todos os construtores que não sejam ingleses
- Mostrar o nome de todas as pilotas de qualquer país ou todos os pilotos não italianos com o nome formatado da seguinte forma: “<ultimo nome>, <primeiro nome>”. Ordenar por ordem alfabética do último nome