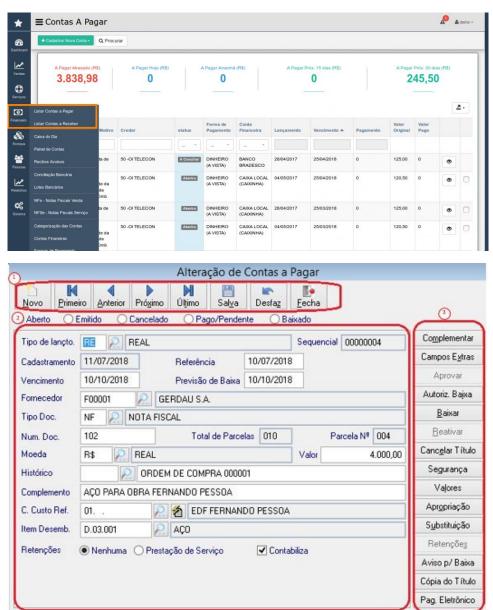
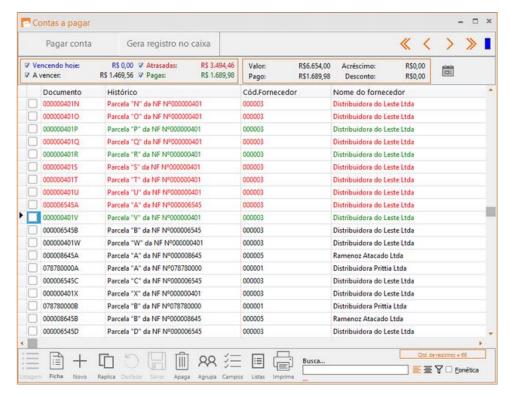
Lista 3 de banco de dados – GROUP BY, ORDER BY, HAVING

1) Projete um modelo lógico para um banco de dados de um Sistema de controle financeiro, que contenha MINIMAMENTE as tabelas: **IMPOSTOS** (dica: usem herança ou composição, para as tabelas: impostos estaduais, municipais e federais, se julgarem necessário), **PREÇOS**, **CONTAS** (dica: usem herança ou composição, para as tabelas: contas a receber, contas a pagar e contas pagas, se julgarem necessário). Sistema de controle financeiro se aplica a praticamente a qualquer domínio de aplicação. E logo em seguida, insira 4 valores em cada dado (será importante para prática).

Exemplos na web:





A Sintaxe para **AGRUPAMENTO** é:

SELECT nome_coluna
FROM nome_tabela
WHERE condição
GROUP BY nome_coluna;

Obs: Pode usar sem a condicional também (porém a consulta se torna menos especifica, quando usamos filtros e condições, manipulamos subconsultas)

As funções (funções de agregação) para estabelecer critérios são:

- somatória dos valores de uma coluna: **SUM()**;
- quantidade de registros que atenda a um determinado critério: **COUNT**();
- cálculo da média ponderada: AVG();
- identificar os menores valores: MIN();
- identificar os maiores valores: **MAX()**.

2) Agrupem os resultados a partir de cada subitem abaixo:

- a) Menor imposto em cada tabela (de imposto estadual, mun e fed)
- b) Maior imposto em cada tabela (de imposto estadual, mun e fed)
- c) Menor conta em cada tabela (conta a pagar, receber e pagas)
- d) Some os impostos da tabela IMPOSTOS Estaduais
- e) Some os IMPOSTOS de todas as tabelas
- f) Tire a média ponderada dos preços

É possível manipular a estrutura de um BD para ORDENAR uma *query*? Sim, utilizando a clausula **ORDER BY.** Inclusive ela é recomendada para otimizar uma subconsulta aliada a outras clausulas.

Podendo ser utilizado desta forma:

SELECT nome_coluna
FROM nome_tabela
WHERE condição
GROUP BY nome_coluna

ORDER BY nome_coluna;

A Sua forma mais simples:

SELECT nome_coluna FROM nome_tabela

ORDER BY nome_coluna ASC ou DESC

3) Ordene os resultados:

- a) Ordene de forma crescente preço da tabela preço
- b) Ordene de forma decrescente preço da tabela preço
- c) Ordene o maior imposto dentro a partir do agrupamento das 3 tabelas de impostos. Faça isso para o menor imposto também.

O **HAVING** se comporta de forma antônima ao **WHERE**. O WHERE delimita os resultados extraídos após o uso do FROM, já a cláusula HAVING filtra o retorno do agrupamento.

Especificando uma subconsulta temos:

SELECT

FROM

WHERE

GROUP

BY

nome_coluna

condição

nome_coluna

condição

condição

ORDER BY nome_coluna

Podemos usar o HAVING sem o ORDER BY? Sim. Sem o WHERE? Sim. Porém teremos uma subconsulta menos especifica.

DESAFIO Ø

Aplique o HAVING junto com o GROUP BY e ORDER BY nas tabelas IMPOSTOS e observem o resultado.