

Exercício 06 - Álgebra Relacional

1- Descreva as principais restrições de Integridade do modelo Relacional.

Restrições de domínios: Especificam que o valor de cada atributo deve ser um valor atômico do domínio, ou seja, os valores válidos para um determinado atributo devem conter o mesmo tipo de dados do domínio definido para o atributo.

Restrição de obrigatoriedade: Indica se deve ou não ser permitida a existência de nulos numa coluna.

Restrição de integridade existencial: Refere-se às chaves primárias e procura garantir que toda e qualquer linha de uma tabela deve poder ser acessada com base apenas no conteúdo de sua chave primária.

Restrição de integridade referencial: É especificada entre duas relações e é utilizada para manter consistência entre tuplas de duas relações. Informalmente, a restrição de integridade referencial declara que uma tupla em uma relação que se refere a uma outra relação deve se referir a uma tupla existente naquela relação.

Fontes:

<https://sites.google.com/site/fkbancodedados1/modelo-logico>

https://en.wikipedia.org/wiki/Relational_database

2- Qual a diferença entre chaves e índices?

Uma chave identifica a linha armazenada no banco de dados.

Um índice é uma estrutura que contém chave e suas localizações. Ajuda você a encontrar a localização das linhas. No caso de um livro, o índice diz em qual página você pode encontrar a palavra. O índice do banco de dados tem a mesma função.

Fonte: <https://stackoverflow.com/questions/769407/in-a-database-what-is-the-difference-between-a-key-and-an-index>

3- Porque na abordagem relacional a ordem das tuplas é irrelevante assim como das linhas?

"NO modelo relacional as estruturas lógicas de dados – tabelas de dados, visualizações e índices – estão separadas das estruturas de armazenamento físico. Essa separação significa que os administradores de banco de dados podem gerenciar o armazenamento físico de dados sem afetar o acesso a esses dados como uma estrutura lógica. Por exemplo, renomear um arquivo de banco de dados não renomeia as tabelas armazenadas dentro dele.

A distinção entre lógica e física também se aplica às operações de banco de dados, que são ações claramente definidas que permitem aos aplicativos manipular os

dados e estruturas do banco de dados. As operações lógicas permitem que um aplicativo especifique o conteúdo de que precisa e as operações físicas determinam como esses dados devem ser acessados e, em seguida, realiza a tarefa.

Para garantir que os dados sejam sempre precisos e acessíveis, os bancos de dados relacionais seguem certas regras de integridade. Por exemplo, uma regra de integridade pode especificar que linhas duplicadas não são permitidas em uma tabela, a fim de eliminar o potencial de informações errôneas que entram no banco de dados."

Fonte:

<https://www.oracle.com/database/what-is-a-relational-database/>

"A Sequência de Colunas é insignificante

Esta propriedade afirma que o ordenamento das colunas na tabela relacional não tem sentido. As colunas podem ser recuperadas em qualquer ordem e em várias sequências. A vantagem desta propriedade é que permite que muitos usuários compartilhem a mesma tabela sem se preocupar em como a tabela é organizada. Também permite que a estrutura física do banco de dados mude sem afetar as tabelas relacionais.

A Sequência de Linhas é insignificante

Esta propriedade é análoga à anterior, mas se aplica a linhas em vez de colunas. O principal benefício é que as linhas de uma tabela relacional podem ser recuperadas em diferentes ordem e sequências. Adicionar informações a uma tabela relacional é simplificado e não afeta consultas existentes."

Fonte:

https://www.cs.wcupa.edu/~zjiang/RDB_table.htm

- 4- Represente os comandos SQL correspondentes às expressões em álgebra relacional:

a- $\pi_{\text{descr, créditos}} (\text{Disciplina})$

```
Select descr, créditos from Disciplina;
```

b- $\pi_{\text{IdAluno, NomeAluno}} (\sigma_{\text{idCurso} = 003} (\text{ALUNO}))$

```
select idaluno, nomealuno from aluno
Where idcurso = 003;
```

c- $\pi_{\text{NomeAluno, anoVestibular}} (\sigma_{\text{conceito} = 'A'} (\text{Matricula} \bowtie_{\text{idAluno} = \text{idAluno}} \text{Aluno}))$

```
select nomealuno, anovestibular
from matricula inner join aluno
on matricula.idaluno = aluno.idaluno
Where conceito = 'A';
```

- 5- Considere as relações F e D que representam respectivamente as entidades Funcionário e Dependentes:

F(numf, nomef)
D(numf, nomed, par) ,

Onde *numf* é o número de matrícula do funcionário, *nomef* é o nome do funcionário, **nomed** é o nome do dependente e *par* é o grau de parentesco (esposa(o), filho, filha, etc.)

Escreva em álgebra relacional as seguintes consultas:

1- Quais os nomes e parentescos de todos os dependentes?

$\pi_{\text{nome, parentesco}} (\text{Dependente})$

2- Liste o número de matrícula dos funcionários que possuem dependentes filhas.

$\pi_{\text{matricula}} (\sigma_{\text{parentesco} = \text{'filha'}} (\text{Dependente}))$

3- Liste o nome dos funcionários que possuem algum dependente.

$\pi_{\text{Funcionario.nome}} (\bowtie_{\text{Funcionario.idFuncionario=Dependente.idFuncionario}})$

4. Liste o nome de cada funcionário que tem uma dependente chamada ALICE.

$\pi_{\text{Funcionario.matricula}} (\sigma_{\text{nome} = \text{'ALICE'}} (\bowtie_{\text{Funcionario.idFuncionario=Dependente.idFuncionario}}))$

4- Liste o número de matrícula dos funcionários que não possuem dependentes.

$\pi_{\text{Funcionario.matricula}} (\sigma_{\text{iddependente} = \text{'NULL'}} (\bowtie_{\text{Funcionario.idFuncionario=Dependente.idFuncionario}}))$

)

5- Liste o nome dos funcionários que não possuem dependentes.

```
 $\pi$  Funcionario.Nome (  $\sigma$  iddependente = 'NULL'
(  $\bowtie$  Funcionario.idFuncionario = Dependente.idFuncionario )
)
```

6- Nome dos funcionários que não tem Alice como dependente? (isto é nenhuma dependente chamada Alice, mas tem algum dependente).

```
 $\pi$  Funcionario.Nome (  $\sigma$  iddependente != 'Alice'
(  $\bowtie$  Funcionario.idFuncionario = Dependente.idFuncionario )
)
```

7- Supondo

a **Relação PROJ (PNO, Orçam)**, com chave primária PNO,
a **Relação EMP (ENO, ENome, Cargo)** com chave primária ENO,
a **Relação TRB (ENO, PNO, Dur, Resp)**, com chave primária {ENO, PNO},
chave estrangeira PNO em relação a PROJ e chave estrangeira ENO em
relação a EMP.

Responda: Qual das expressões da álgebra relacional abaixo NÃO
corresponde à seguinte consulta SQL:

```
SELECT ENome
FROM EMP, TRB, PROJ
WHERE EMP.ENO = TRB.ENO
AND PROJ.PNO = TRB.PNO
AND Dur > 36
```

- a) π ENome (PROJ \bowtie PNO=PNO (EMP \bowtie ENO=ENO (σ Dur > 36 TRB)))
- b) π ENome (PROJ \bowtie PNO=PNO ((π ENome, ENO (EMP)) \bowtie ENO=ENO (σ Dur > 36 TRB)))
- c) π ENome (PROJ \bowtie PNO=PNO (EMP \bowtie (σ Dur > 36 (π Dur (TRB)))))
- d) π ENome (PROJ \bowtie PNO=PNO (EMP \bowtie ENO=ENO (π ENO, PNO (σ Dur > 36 TRB))))

(exercício extraído do exame Pós_Comp da Unicamp – 2003)

7. Sendo as tabelas:

R

3
6
7
8
11

S

2
6
7
8

Indique qual o resultado de:

a- $R \cup S$

3
6
7
8
11
2
6
7
8

b- $R \cap S$

6
7
8

c- $R \bowtie S$

R	S
6	6
7	7
8	8