

## Desvantagens

- Complexidade;
- Tamanho;
- Custos;
- Desemprego por ser generalista

## Esquemas

### Esquema Externo

- Corresponde às diferentes formas de ver os dados.

### Esquema Conceptual

- Corresponde à descrição de todas as entidades, atributos e relações.

### Esquema Interno

- Contém os registos guardados, os campos de dados e os métodos de índice e armazenamento usados.

## Modelo Relacional

- Composto por Relações, Atributos e Tuplos.

Grav - Quantidade de atributos de uma relação.

Cardinalidade - Quantidade de tuplos de uma relação.

## Notação Backus Naur Form (BNF)

- Palavras reservadas em maiúsculas;
- |, indica opção entre as alternativas,  $a|b|c$ ;
- {}, indica um elemento necessário,  $\{a\}$ ;
- [], indica um elemento opcional;

## Data Definition Language (DDL)

- Cria a estrutura de base de dados e os mecanismos de acesso.

## Data Manipulation Language (DML)

- Consultas as tabelas e manipula os dados.

## Linguagem SQL

### SELECT

- Consulta dados, tem o objetivo de obter e mostrar os dados de uma ou mais tabelas de base de dados.

## SELECT

Especifica quais colunas vão aparecer.

Asterisco (\*) → representa todos os elementos.

SELECT salario/12 → pega em cada valor da coluna salario e divide por 12.

FROM → especifica a(s) tabela(s) usada(s)

## WHERE

Filtra as linhas com certa condição.

BETWEEN → WHERE salario BETWEEN 100 AND 200  
verifica se o salario está entre 100 e 200.

IN → WHERE posicao IN ('Professor', 'Tecnico')  
verifica se posicao esta numa condição com valores que contenha "Professor" e/ou "Tecnico".

LIKE → WHERE endereco LIKE '%Glasgow%'  
verifica se o endereco em questão tem a condição de ter 0 ou mais caracteres seguido da sequência Glasgow e seguido depois de 0 ou mais caracteres.

% → 0 ou mais caracteres

\_ → 1 só caractere

IS NULL → WHERE comentario IS NULL

verifica se o comentario tem algum valor nulo.

GROUP BY → Agrupa registros com a mesma coluna.

HAVING → Filtra o grupo a certa condição.

## ORDER BY

Especifica a ordem de saída de dados.

DESC → ORDER BY salario DESC

ordena os dados de forma decrescente

ASC → ORDER BY salario ASC

ordena de forma Ascendente

ASC é a ordem padrão.

COUNT → retorna a quantidade de valores de uma coluna específica.

SUM → retorna a soma de valores de uma coluna.

AVG → retorna a média dos valores de uma coluna.

MIN → retorna o valor mínimo.

MAX → retorna o valor máximo.

WHERE salario > ANY (SELECT...)

o salario tem que ser maior que pelo menos 1 dos salários

WHERE salario > ALL

tem de ser maior que todos.

FROM Tabela1 JOIN Tabela2 USING ColunaComum

consulta múltiplas tabelas.

FROM Tab1 { LEFT JOIN Tab2 ON Condicao  
RIGHT  
FULL

Junta o que satisfaz a condição.

EXISTS

Retorna verdadeiro se e somente se existir pelo menos 1 registro na tabela de subconsulta.

Retorna falso se a tabela de subconsulta estiver vazia.

UNION → a união de duas tabelas, resulta numa tabela com todos os registros que estão tanto numa como noutra tabela.

INTERSECT → a interseção de duas tabelas resulta numa tabela que contém os registros comuns a ambas.

EXCEPT → a diferença entre duas tabelas, A e B, resulta numa tabela que contém todos os registros que estão em A mas não em B.

INSERT → Insere os dados numa tabela

UPDATE → atualiza os dados

DELETE → apaga dados

## Sistema de Gestão de Base de dados Relacional

O SGBD aceita instruções SQL, produz a consulta e mostra os resultados.

### Otimização de Consultas (query optimizer)

- Usa a informação como os dados estão guardados para produzir um plano de execução.

### Plano de Execução (execution plan)

- Avalia a consulta e geralmente apresenta-se como uma árvore de operações relacionais.

### Gestor de Memórias (buffer manager)

- Sendo um SGBD uma coleção de páginas/ficheiros, é o gestor de memórias que traz as páginas do disco para a memória principal, conforme necessário, em resposta às solicitações de leitura.

### Gestor de Espaço em Disco (disk space manager)

- É a camada mais baixa do SGBD, as camadas superiores alocam e desalocam, leem e escrevem páginas através de funções fornecidas por esta camada.

### Gestor de Transações (transaction manager)

- É um gestor que garante que cada transação solicita e liberta os trancos segundo um protocolo de trancos apropriado e agenda a execução das transações.

### Gestor de Trancos (lock manager)

- Guarda toda a informação sobre os registos de pedidos e de concessões de trancos sobre os objetos da base de dados e quando estes ficam disponíveis.

### Gestor de Recuperações (recovery manager)

- Mantém um registo e permite restaurar para um estado consistente após uma falha.



## 2.3.2 Processamento de Consultas

Transformam a consulta numa estratégia de execução correta e eficiente, sendo o foco obter a melhor maneira de avaliar essa consulta.

A descrição de dados, ou metadados, guarda em tabelas especiais designadas por catálogos do sistema, é usada para encontrar a melhor forma de avaliar uma consulta.

• Índice: é uma estrutura de acesso rápido aos dados e está associado a uma ou mais colunas de uma tabela.

## 2.3.3 Catálogo

Nomeado também de tabelas do catálogo, dicionário de dados, Catálogo do sistema, possui dados globais ao sistema, tais como o tamanho do conjunto dos registros e tamanho das páginas, assim como dados sobre as tabelas, índices e vistas:

Para cada tabela:

- Nome da tabela;
- Nome e estrutura do ficheiro onde está armazenada;
- Nome e tipo de cada atributo;
- Nome de cada índice da tabela;
- Restrições de integridade.

Para cada índice

- Nome e estrutura do índice;
- Atributos da chave de pesquisa.

Para cada vista

- Nome e definições

Adicionalmente, contém informação acerca dos utilizadores e privilégios como ainda estatísticas sobre tabelas e índices atualizadas periodicamente, contendo:

- cardinalidade da cada tabela;
- Quantidade de páginas de cada tabela;
- cardinalidade dos índices;
- quantidade de páginas de cada índice;
- quantidade de níveis não-folha de cada índice;
- Grau do índice.

## 2.3.4 Fases do processamento de consultas

### Decomposição (análise e validação)

- Transforma consulta de alto nível em álgebra relacional;
- Verifica se esta conexão sintática e semanticamente.

#### Estágios de decomposição:

##### Análise

- Analisa sintaticamente com técnicas dos compiladores de linguagens de programação.

##### Normalização

- Converte a consulta para algo facilmente manipulável.

##### Análise semântica

- Rejeita consultas inconsequentemente formuladas ou contraditórias.

##### Simplificação

- Deteta e remove redundâncias, eliminando subexpressões comuns para a consulta, tornando-a mais fácil e eficiente. É visto ainda as restrições de acesso, definições de vistas e restrições de integridade.

##### Reestruturação de Consulta

- ~~Para~~ Para fornecer uma implementação mais eficiente.

##### Otimização

Objetivo de encontrar um bom plano de execução, normalmente representados em álgebra relacional. Para otimizar envolve 2 passos:

- Enumera planos alternativos;
- Estima o custo de cada plano enumerado e escolhe o ~~menor~~ de menor custo estimado.

##### Geração de Código

##### Execução de Código

## 2.3.6 As 12 Regras de Codd

Conjunto de regras que um SGBD tem que obedecer para considerar-se como relacional.

1. Todos os dados, incluindo o próprio catálogo de dados, são representados numa só forma, em tabelas bidimensionais;

2. Cada elemento de dados fica bem determinado pela combinação do nome da tabela onde está guardado, valor da chave primária e respetiva coluna (atributo);

Valores nulos (NULL) são suportados para alguns tipos de dados, não disponíveis ou não aplicável, independentemente do formato das propriedades estatísticas.

As metadados são representados e criados de maneira física que os usuários duvidam.

É Term de criação pelo menos uma linguagem com as seguintes características:

- Manipulação física, com uma biblioteca de utilização independente ou em programas de aplicação;
- Definição de dados;
- Definição de vistas;
- Definição de restrições de integridade;
- Definição de acessos (autorizações);
- Manipulação de dados.

1 - Nunca vistas, todos os dados atualizados modificados devem vir essas modificações introduzidas nas tabelas base;

2 - Capaz de se tratar uma tabela como se fosse uma simples operação, sendo em operações como de atualização;

3 - Alterações na organização física dos ficheiros de base de dados e os métodos de acesso a esses não devem afetar o nível conceitual;

4 - Alterações no nível conceitual, que não envolvam remoção de elementos, não devem afetar o nível externo;

5 - Restrições de integridade devem poder se especificadas numa linguagem notacional independentemente dos programas de aplicação e armazenadas no catálogo;

6 - A física de uma base de dados centralizam-se numa máquina ou distribuem-se por várias, não deve repensar-se no nível da manipulação dos dados;

7 - Se existir no sistema uma linguagem de nível baixo-nível (tipo orientada a ~~dados~~ registros, record-oriented), esta não deve permitir ultrapassar as restrições de integridade e segurança.

50

select C.Nome

from Ciclista C, (

select top 1 P.IDC, count(P.IDC) AS TotalParticipacoes

from Participacao P

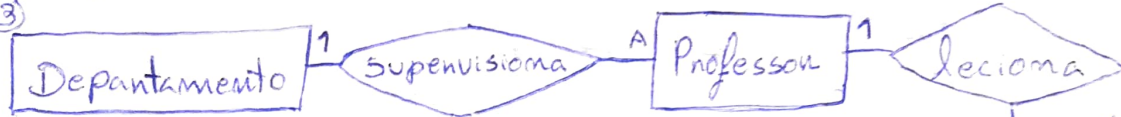
Group By P.IDC

Order By TotalParticipacoes DESC) AS ParticipacaoCliente

Where C.IDC = ParticipacaoCliente.IDC

Folha 5

③



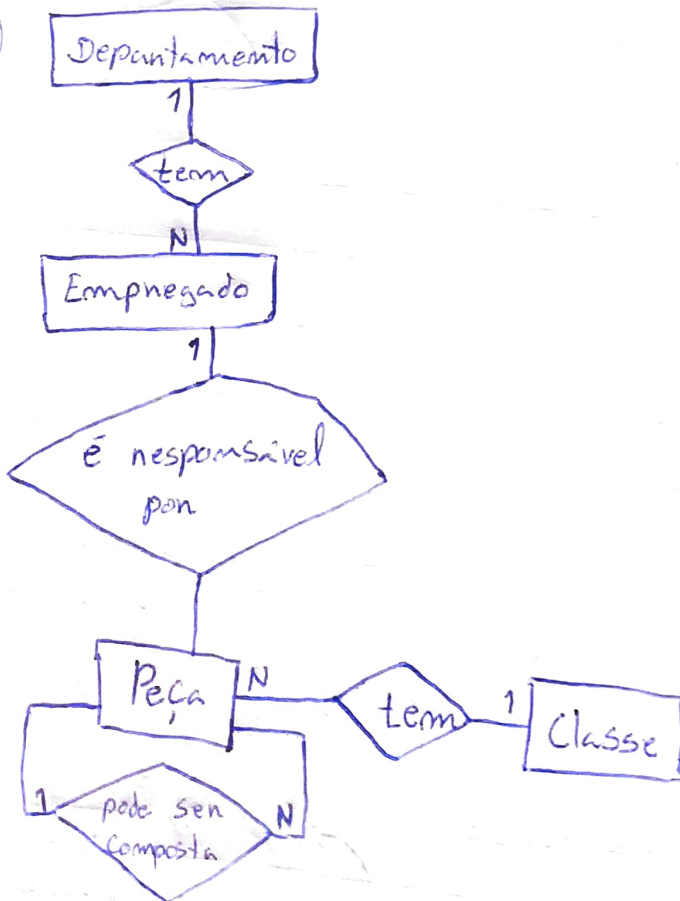
Departamento (IDD, IDP, Nome-Dep);

Professor (IDP, Nome-Prof, Salario, Telefone, Gabinete);

Cursos (IDC, IDD, Nome-Curso, IDP, Data-inicio, Data-fim, num-sal);

Estudante (IDE, Nome-Estudiante, Endereco, IDC, Classificacao);

④



Classe (IDC, Nome-classe, descrição);

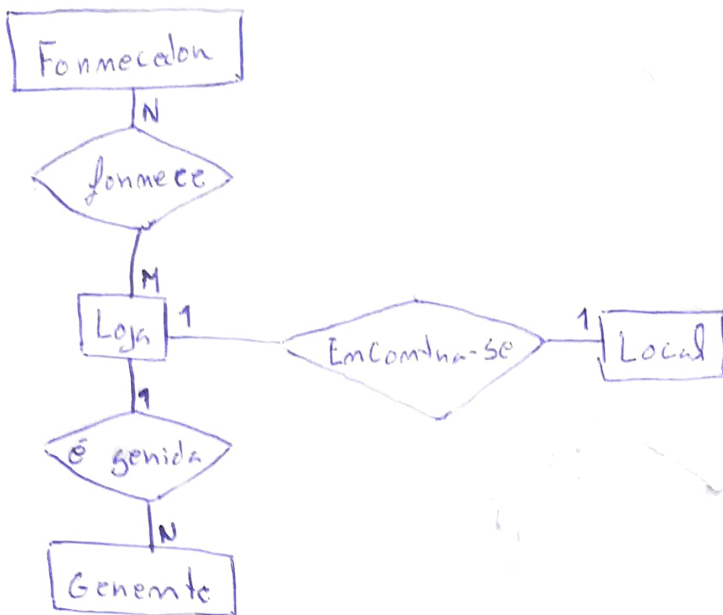
Peça (IDP, IDC, IDE, Nome-peça, descrição, Estado);

Departamento (IDP, Nome-Departamento);

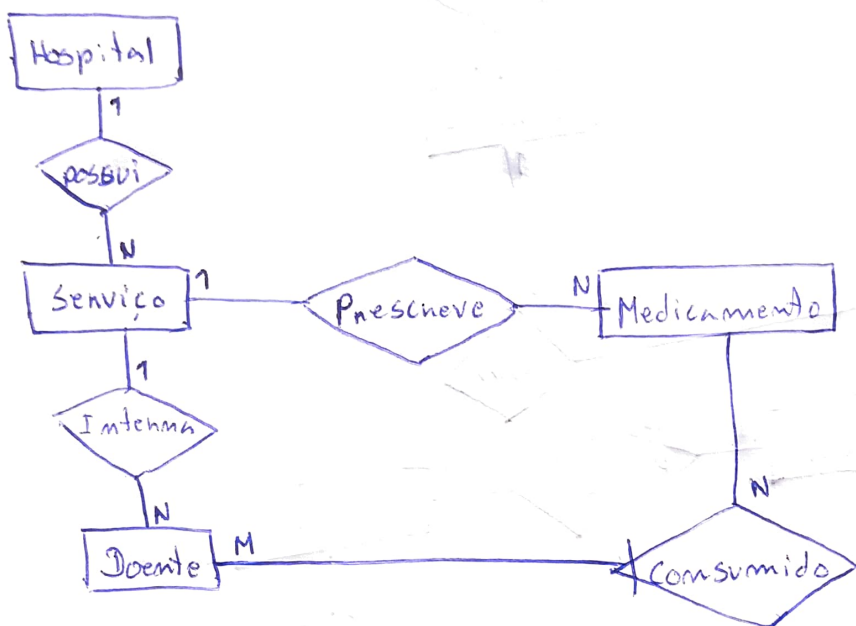
Empregado (IDE, IDP, Nome-Empregado, Telefone, local-trabalho);



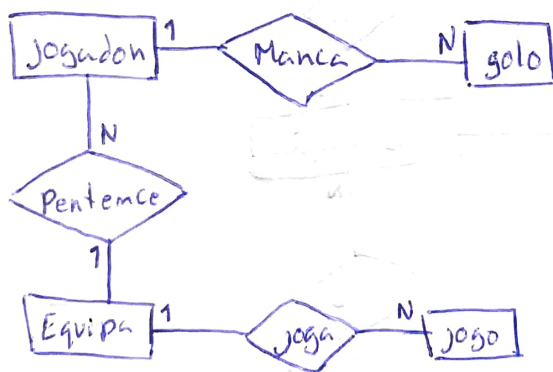
5) a)



b)



c)



d)

