

# BANCO DE DADOS

## AULA 1

+55 (21) 99461-8818

@explicadoresnet

[www.explicadores.net.br](http://www.explicadores.net.br)



## MODELO NAVEGACIONAL

No modelo, os dados são organizados em registros, que são coleções de itens de dados, e podem ser armazenados ou recuperados de um banco de dados de forma conjunta.

## MODELO SEMI-ESTRUTURADOS

Neste modelo onde os dados são guardados e manipulados na forma de **XML** (ao contrário da forma de tabelas).

XML é a sigla para “*Extensible Markup Language*”, uma linguagem de marcação com regras para formatar documentos de forma que eles sejam facilmente lidos tanto por humanos quanto por máquinas.

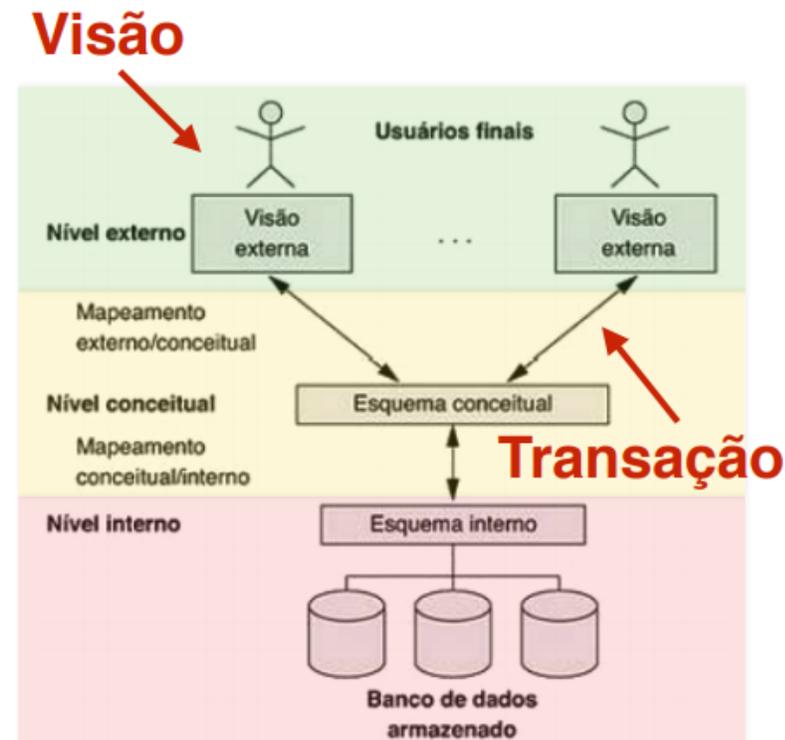


# Arquitetura ANSI/SPARC

Modelo em **três níveis**, com o objetivo de separar as aplicações e banco de dados físico, garantindo **independência** de dados.

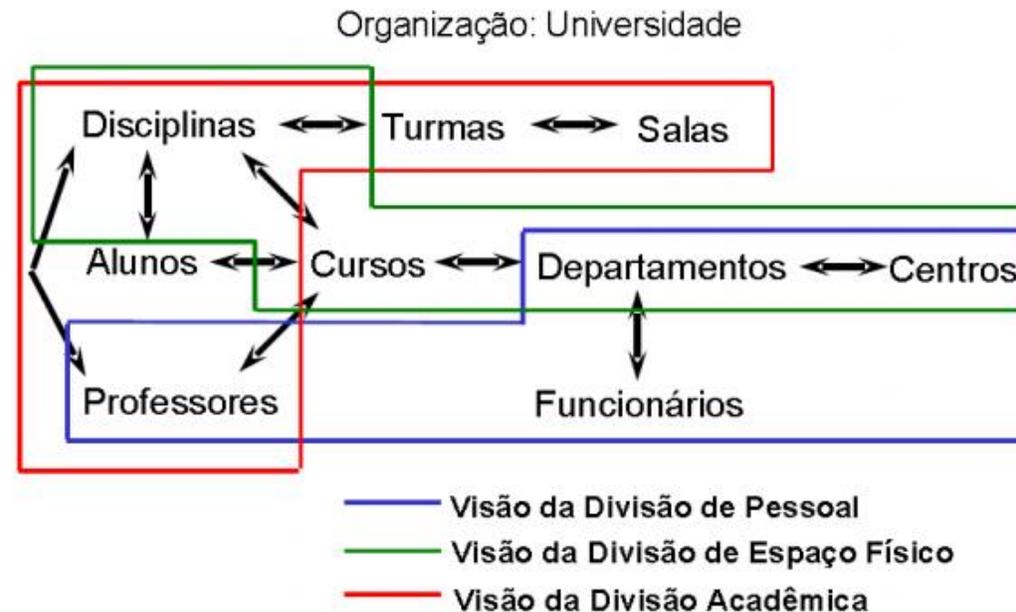
**Independência lógica**, permite alterar o esquema conceitual sem alterar o esquema externo nem as aplicações.

**Independência física**, permite alterar o esquema físico sem alterar o esquema conceitual.



# Visão

Uma **visão** não precisa estar necessariamente armazenada no banco de dados. Ela pode ser uma interpretação processada dos dados para facilitar a visualização de um grupo de usuários.



# Transações

São **ações** que podem ser solicitadas por multi-usuários e devem ser tratadas para que não haja conflito. Possui as seguintes características:

**A**tomicidade

**C**onsistência

**I**solamento

**D**urabilidade

**ACID**

**CIDA**

**DICA**

# Transações

## ACID

**Atomicidade:** Uma transação não pode ser dividida em pedaços menores. Não existe meio-termo, ou todo o trabalho é feito (*commit*), ou nada é feito (*abort/rollback*).

**Consistência:** A execução de uma transação deve levar o banco de um estado consistente para outro estado igualmente consistente.

**Isolamento:** Evita que transações paralelas interfiram umas nas outras. O resultado entre solicitações simultâneas deve ser o mesmo obtido se elas fossem sequenciais.

**Durabilidade:** Os efeitos de uma transação de sucesso (*commit*) deve persistir no banco de dados mesmo em situações adversas.



# Linguagens SGBD

**Data Definition Language - DDL**, usada pelo DBA e projetistas para criar o esquema conceitual.

**Data Manipulation Language - DML**, usada após a criação do banco para manipular os dados.

**Data Control Language - DCL**, autorizar (*grant*) ou revogar (*revoke*) acesso ao banco ou parte dele.

# Linguagens SGBD

Uma das linguagens mais usadas e mais completas para SGBD é a **Structured Query Language**.

A SQL possui características de **DDL + DML**.

Algumas versões da SQL implementam características de **DCL**.

# SQL

Structured Query Language

*Linguagem de Consulta Estruturada*

## Subconjuntos

**DDL - *Data Definition Language***: criação, modificação e remoção estrutural.

CREATE

ALTER

DROP

**DML - *Data Manipulation Language***: recuperação, inclusão, remoção de dados.

INSERT

UPDATE

DELETE

SELECT

**DCL - *Data Control Language***: controle de acessos

GRANT

REVOKE



# DDL-SQL

Data Definition Language

```
mysql -u root -p
```

Conectar ao servidor MySQL com o seguinte comando

## CREATE DATABASE

```
CREATE DATABASE banco;
```

```
CREATE DATABASE explicadores;
```

```
CREATE DATABASE nome-do-banco character set utf8 collate utf8-general-ci;
```

Cria o banco de dados usando codificação UTF-8

**USE DATABASE;** Antes de criar uma tabela ou realizar qualquer operação, é necessário selecionar o banco de dados que vai ser usando

```
USE EXPLICADORES;
```

## CREATE TABLE

```
CREATE TABLE tabela  
( coluna1 tipo(tam),  
  coluna2 tipo(tam),  
  ...);
```

```
CREATE TABLE alunos  
(Mat int,  
  Nome varchar(35),  
  Telefone varchar(12));
```

## Tipos Primitivos de Dados

**VARCHAR**( $n$ ) - string com tamanho máximo  $n$ .

**CHAR**( $n$ ) - string de tamanho fixo  $n$ .

**TEXT** - texto longo

**SMALLINT** - inteiro com precisão de 16 bits

**INT** - inteiro com precisão de 32 bits

**FLOAT** - número real

**DECIMAL**( $n,d$ ) - número com  $n$  casas e  $d$  decimais

**DATE** - data

**TIME** - hora

**TIMESTAMP** - data e hora

# CREATE TABLE + constraints

```
CREATE TABLE tabela  
( coluna1 tipo(tam) restrição,  
  coluna2 tipo(tam) restrição,  
  ...);
```

```
CREATE TABLE alunos  
(mat int,  
  nome varchar(35) NOT NULL,  
  tel varchar(12),  
  codCurso int,  
  UNIQUE (nome),  
  PRIMARY KEY(mat),  
  FOREIGN KEY(codCurso) REFERENCES cursos(cod));
```

**NOT NULL** - não aceita valor opcional

**UNIQUE** - valor único nas tuplas

**PRIMARY KEY** - chave primária

**AUTO\_INCREMENT** - incremento automático

**FOREIGN KEY** - chave estrangeira

**DEFAULT** - especifica um valor padrão

**Enum('F','M');**

**DEFAULT CHARSET=utf8  
AUTO\_INCREMENT=1000;**

# ALTER TABLE

```
ALTER TABLE alunos  
ADD cidade VARCHAR(20) NOT NULL;
```

```
ALTER TABLE alunos  
ADD UNIQUE (tel);
```

```
ALTER TABLE alunos  
DROP PRIMARY KEY;
```

```
ALTER TABLE alunos  
DROP COLUMN tel;
```

```
ALTER TABLE alunos  
MODIFY cidade VARCHAR(20) DEFAULT 'Rio de Janeiro'
```

```
ALTER TABLE alunos CHANGE sex sexo varchar(35);
```

```
ALTER TABLE alunos T4 RENAME T5;
```

# CREATE INDEX

```
CREATE INDEX indice  
ON tabela (coluna)
```

```
CREATE INDEX ordemNome  
ON alunos (nome)
```

# DROP

DROP DATABASE educandus;

DROP TABLE alunos;

DROP INDEX ordemNome;

ALTER TABLE alunos DROP INDEX ordemNome;

TRUNCATE TABLE alunos;

**DROP DATABASE IF EXISTS explicadores;**

**Apague o banco explicadores, caso ele exista.**

# SHOW DATABASES

```
SHOW DATABASES;
```

```
+-----+
| Database          |
+-----+
| information_schema |
| mysql              |
| performance_schema |
| test               |
+-----+
04 rows in set (0.00 sec)
```

# SHOW TABLES

```
SHOW TABLES;
```

```
+-----+
| Tables_in_mysql |
+-----+
| columns_priv    |
| db              |
| event           |
| func            |
| time_zone_transition |
| time_zone_transition_type |
| user            |
+-----+
07 rows in set (0.00 sec)
```

# DESC

DESC users;

Field	Type	Null	Key	Default	Extra
Host	char(60)	NO	PRI		
User	char(16)	NO	PRI		
Password	char(41)	NO			
max_questions	int(11) unsigned	NO		0	
max_updates	int(11) unsigned	NO		0	
authentication_string	text	YES		NULL	
password_expired	enum('N','Y')	NO		N	

07 rows in set (0.01 sec)