

Banco de Dados – C07

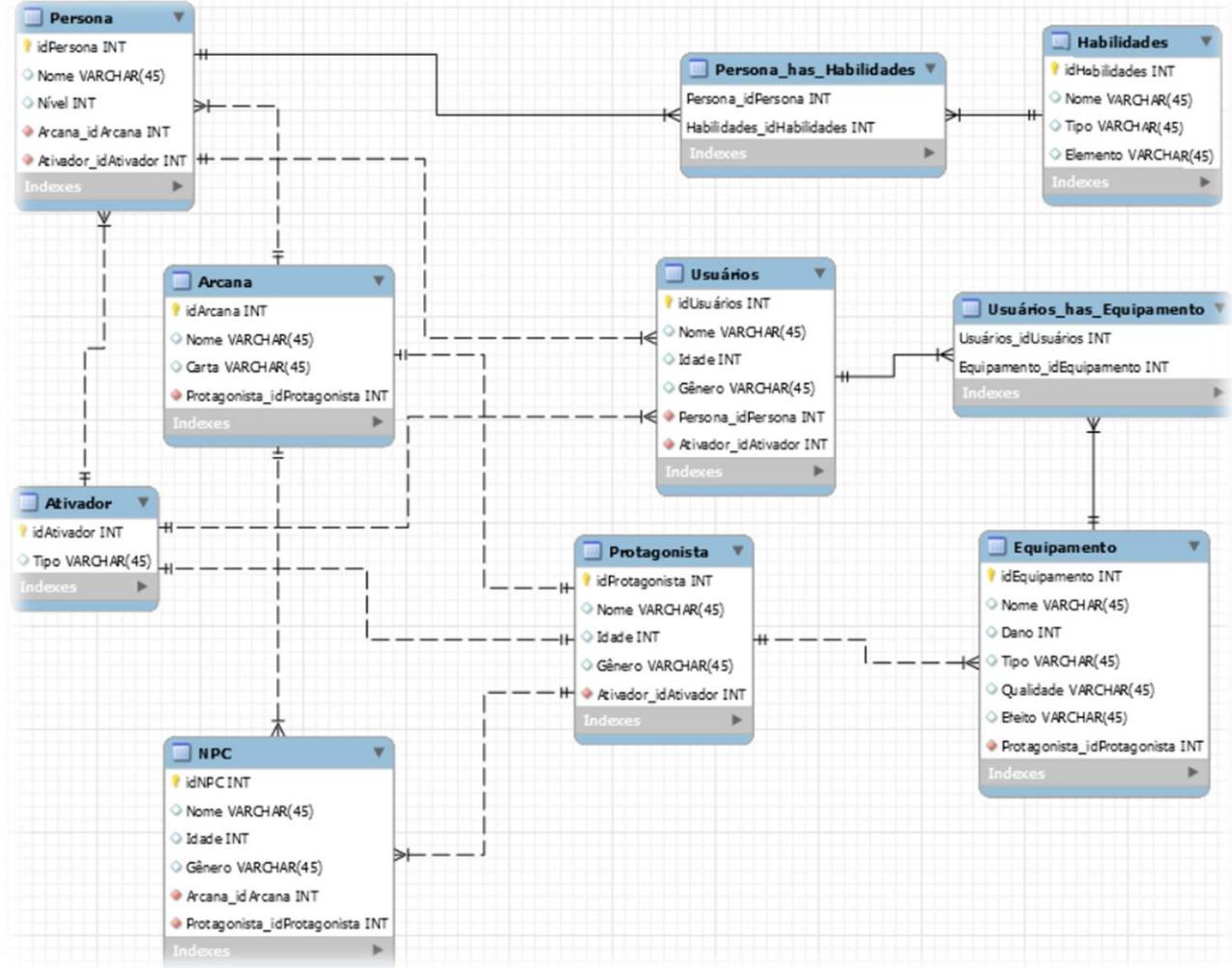
Felipe Tagawa Reis

Introdução a Banco  
de Dados Relacionais

**Inatel**

# Bancos de Dados Relacionais

São sistemas de gerenciamento e armazenamento de dados em tabelas estruturadas com linhas e colunas, onde as informações são distribuídas e organizadas de forma lógica e inter-relacionada.



# Prós e Contras

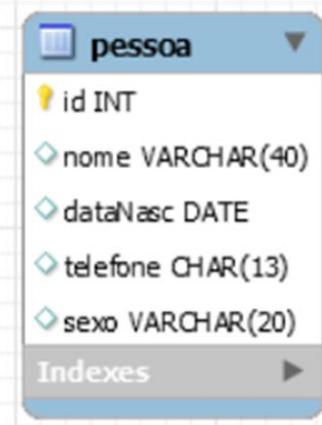
## Prós

- Garantia **ACID** – Segurança e Consistência;
- Sistema Robusto de consultas;
- Padronização – Estabilidade.

## Contras

- Escala Vertical (Baixa Escalabilidade);
- Esquema Rígido (Pouca Flexibilidade);
- Menor Desempenho em cenários Big Data – Exemplo do Post do Instagram.

# Tabelas



Modelo do  
Workbench

id(PK)	nome	dataNasc	telefone	sexo

Tabela Simples

# Tipos de Dados mais Utilizados

## Numéricos

- INTEGER(INT);
- DECIMAL(i,j);
- FLOAT(j);
- DOUBLE.

i: dígitos totais;  
j: dígitos decimais.

## Data e Hora

- DATE;
- TIME;

DATE (YYYY-MM-DD);  
TIME (HH:MM:SS).

## String

- CHAR(n);
- VARCHAR(n);
- TEXT.

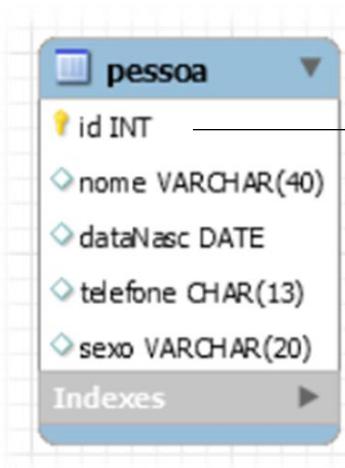
CHAR é FIXO em n caracteres!  
VARCHAR aceita até n caracteres!  
TEXT não tem quantidade fixa, é bastante usado para textos e descrições extensos.

## Lógico

- BOOLEAN.

# Chaves Primárias (Primary Key/PK)

- É um ou mais campos (atributos) que identificam **unicamente** cada registro em uma tabela;
- **NÃO pode conter valores nulos (NULL) nem valores duplicados;**
- **Cada tabela pode ter apenas uma chave primária**, mas essa chave pode envolver **mais de um campo**
  - nesse caso, é chamada de **chave primária composta**;
- Em relacionamentos entre tabelas, uma chave primária pode ser combinada com a chave estrangeira de outra tabela para compor uma chave — esse conceito será abordado posteriormente.



Chave Primária  
da tabela pessoa

# Chaves estrangeiras

```
CREATE TABLE Bruxo (
    id INT PRIMARY KEY AUTO_INCREMENT,
    nome VARCHAR(255) NOT NULL,
    Casa_idCasa INT NOT NULL,
    Varinha_idVarinha INT NOT NULL,
    CONSTRAINT fk_Casa
    FOREIGN KEY (Casa_idCasa)
    REFERENCES Casa(idCasa),
    CONSTRAINT fk_Varinha
    FOREIGN KEY (Varinha_idVarinha)
    REFERENCES Varinha(idVarinha)
    ON UPDATE CASCADE
    ON DELETE CASCADE
);
```

- CONSTRAINT - Bloco de comandos para configurar a chave estrangeira. Para cada chave estrangeira da tabela um bloco CONSTRAINT deve ser feito
- ON UPDATE e ON DELETE - Definem o que fazer estrangeira. ao se excluir um registro em outra tabela relacionado pela chave estrangeira
- CASCADE - A alteração na tabela inicial se estende ao registro da tabela com a chave estrangeira
- FOREIGN KEY - Comando que mostra qual coluna é uma chave estrangeira
- REFERENCES - Referencia a qual tabela a chave estrangeira está relacionada

# Relacionamentos

**Conexões lógicas entre entidades** que representam como os dados em diferentes tabelas estão associados uns aos outros.

No modelo relacional, **relacionamentos** ajudam a manter a **integridade referencial**, evitando dados duplicados e inconsistentes.

Classificação:

- 1:1
- 1:N
- N:M

# Relacionamentos

## ❖ Relacionamento 1:1

Esse tipo de relacionamento é menos comum, mas útil quando há a necessidade de **separar dados sensíveis** ou dividir lógicas distintas. Por exemplo, em um sistema de RH, cada funcionário pode ter um único registro de pagamento.

## ❖ Relacionamento 1:N

É o tipo mais frequente de relacionamento em bancos de dados. Imagine um cliente que faz vários pedidos: temos uma relação **1 cliente → N pedidos**. Isso permite organizar o sistema de forma escalável, mantendo os dados centralizados e fáceis de acessar.

## ❖ Relacionamento N:M

Esse relacionamento ocorre quando várias instâncias de uma entidade se relacionam com várias instâncias de outra. Como não é possível representar isso diretamente em bancos relacionais, utilizamos **tabelas associativas** para intermediar a conexão.

# Instalação do MySQL Workbench

- Vídeo com os detalhes da instalação no canal do Teams.
- Arquivo com todos os detalhes no TEAMS ou no meu repositório da disciplina no Github.
- IMPORTANTE: para facilitar entradas posteriores (principalmente no projeto), recomendo colocar o nome de usuário e senha como “root”.

AULA  
CONCLUÍDA!