



Flickr, shindotv/3835365695/

# BASES DE DADOS

# Armazenamento de Informação

- Aplicações necessitam de armazenar informação
  - ▣ Configurações
  - ▣ Dados de operação
    - Emails, imagens, etc...
- Necessitam de lhe aceder de forma rápida

# Armazenamento de Informação

- Podem ser utilizados ficheiros “normais”
- Útil para elementos discretos
  - Imagens
  - Emails
  - Músicas
  - Documentos

# Armazenamento de Informação

- Como armazenar informação tabular?
  - ▣ listagens de contactos, registos de compras, etc...

<b>name</b>	<b>address</b>	<b>phone</b>
Pedro Miguel Mota	Aveiro	918376376
Alexandra Maria Silva	Lisboa	965786735

# Armazenamento de Informação

- “solução”: usar CSV, JSON, etc...
- Problema:
  - ▣ Pesquisar informação rapidamente
  - ▣ Alterar um elemento
  - ▣ Apagar informação
  - ▣ Correlacionar informação:
    - Pedro trabalha na empresa XPTO

# Bases de dados relacionais

- Organizam informação tabular
- Permitem acesso rápido CRUD
  - ▣ Create
  - ▣ Read
  - ▣ Update
  - ▣ Delete
- Permitem relacionar linhas entre tabelas
  - ▣ A funcionalidade mais útil!

# Estrutura

---

- Tabelas

- ▣ Possuem um nome

- Colunas

- ▣ Possuem um tipo
  - ▣ Possuem um nome

- Linhas ou Registos

# Estrutura

## □ Tabela “Contactos”

Nome da coluna e Tipo

<b>name</b> <b>TEXT</b>	<b>address</b> <b>TEXT</b>	<b>phone</b> <b>INTEGER</b>	
Pedro Miguel Mota	Aveiro	918376376	
Alexandra Maria Silva	Lisboa	965786735	

Linha



# Relações entre tabelas

---

**Tendo contactos e empresas, como dizer que um contacto pertence a uma empresa?**

# Relações entre tabelas

- Cada tabela lista um tipo de dados
- Linhas podem possuir identificadores
  - ▣ Chamados: Chaves Primárias
  - ▣ Que são utilizadas noutras tabelas
    - Chaves Estrangeiras
- Evita-se replicação de informação
  - ▣ Ex: escrever o nome da empresa em cada contacto

# Relações entre tabelas

## Contactos

<b>name</b> <b>TEXT</b>	<b>address</b> <b>TEXT</b>	<b>phone</b> <b>INTEGER</b>
Pedro Miguel Mota	Aveiro	918376376
Alexandra Maria Silva	Lisboa	965786735

## Empresas

<b>name</b> <b>TEXT</b>	<b>address</b> <b>TEXT</b>	<b>vat</b> <b>INTEGER</b>
ProArte	Porto	50123344354
MaxiPortas	Lisboa	50134325673

# Relações entre tabelas

## Chaves Primárias

### Contactos

<b>id</b> <b>INTEGER KEY</b>	<b>name</b> <b>TEXT</b>	<b>address</b> <b>TEXT</b>	<b>phone</b> <b>INTEGER</b>
1	Pedro Miguel Mota	Aveiro	918376376
2	Alexandra Maria Silva	Lisboa	965786735

### Empresas

<b>id</b> <b>INTEGER KEY</b>	<b>name</b> <b>TEXT</b>	<b>address</b> <b>TEXT</b>	<b>vat</b> <b>INTEGER</b>
1	ProArte	Porto	50123344354
2	MaxiPortas	Lisboa	50134325673

# Relações entre tabelas

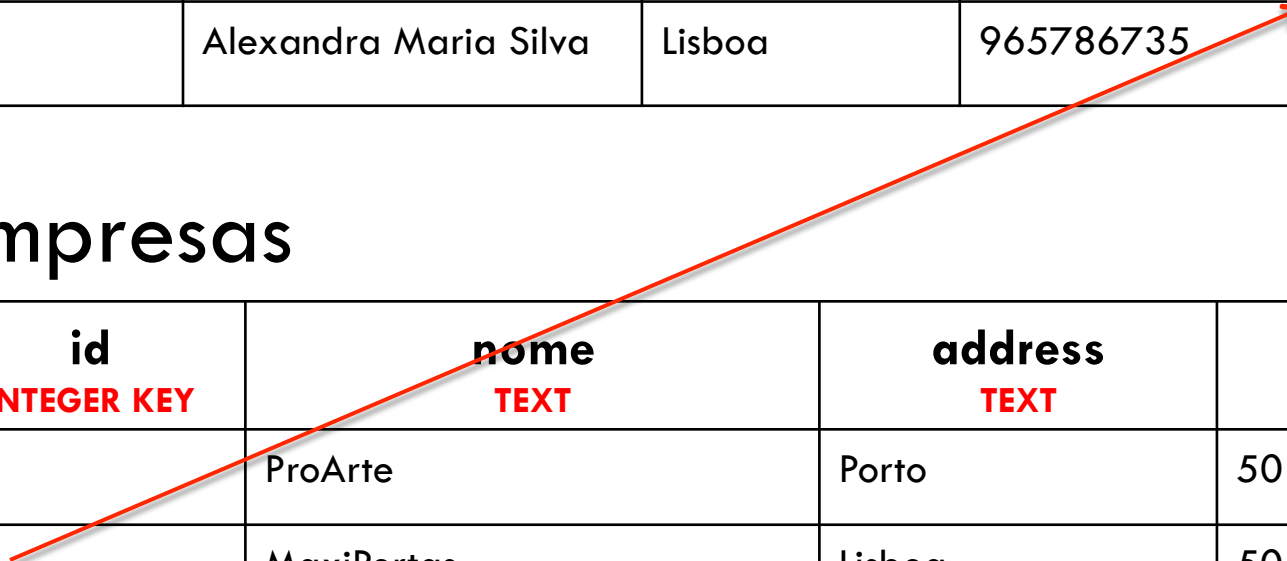
## Chaves Estrangeiras

### Contactos

<b>id</b> INTEGER KEY	<b>name</b> TEXT	<b>address</b> TEXT	<b>phone</b> INTEGER	<b>comp_id</b> INTEGER
1	Pedro Miguel Mota	Aveiro	918376376	2
2	Alexandra Maria Silva	Lisboa	965786735	2

### Empresas

<b>id</b> INTEGER KEY	<b>nome</b> TEXT	<b>address</b> TEXT	<b>vat</b> INTEGER
1	ProArte	Porto	50123344354
2	MaxiPortas	Lisboa	50134325673



# Relações entre tabelas

## Chaves Estrangeiras

### Contactos

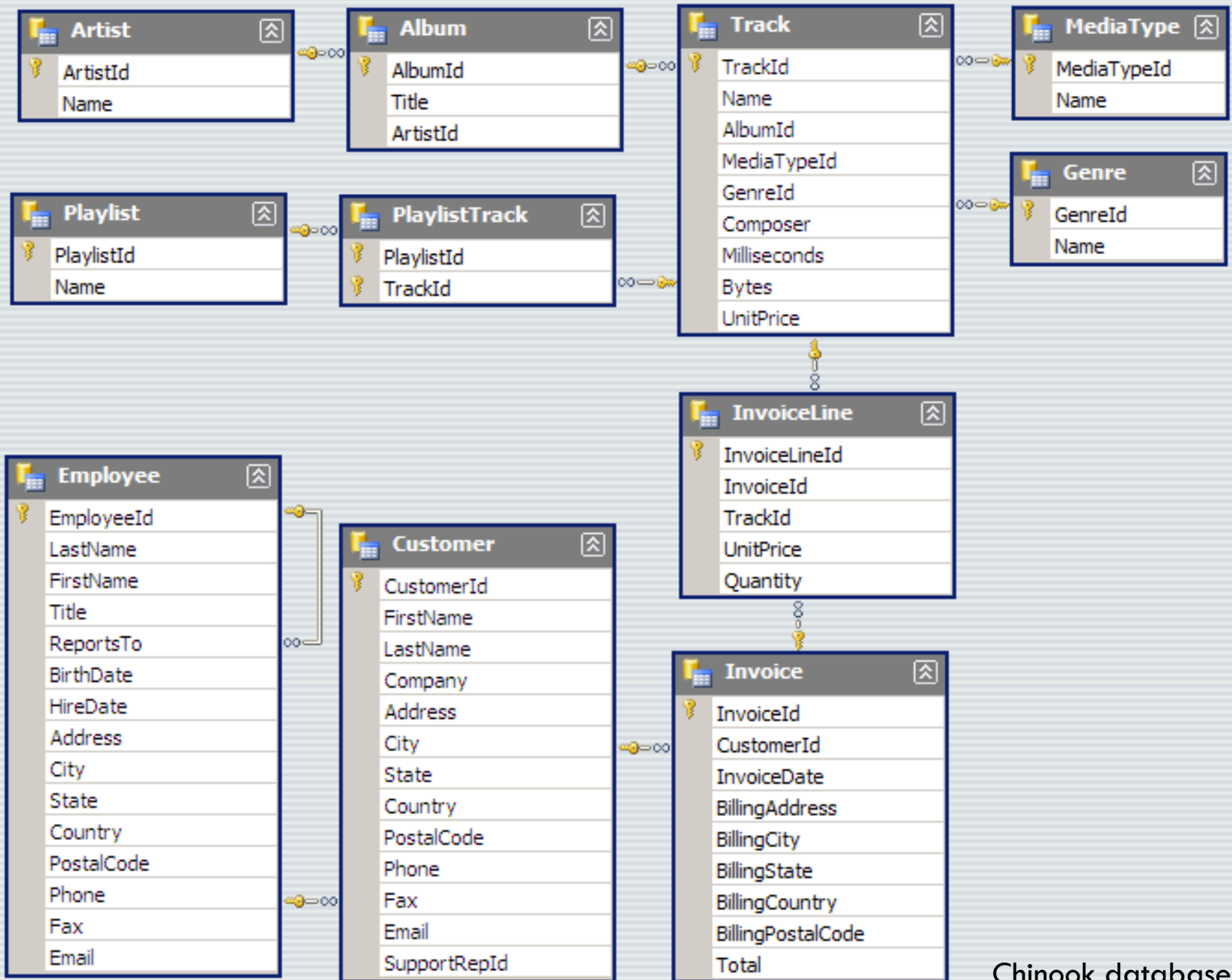
<b>id</b> <b>INTEGER KEY</b>	<b>name</b> <b>TEXT</b>	<b>address</b> <b>TEXT</b>	<b>phone</b> <b>INTEGER</b>	<b>comp_id</b> <b>INTEGER</b>
1	Pedro Miguel Mota	Aveiro	918376376	2
2	Alexandra Maria Silva	Lisboa	965786735	2

**Ambos os contactos pertencem à mesma empresa.**

***comp\_id* é uma chave estrangeira**

***id* é uma chave primária**

<b>id</b> <b>INTEGER KEY</b>	<b>name</b> <b>TEXT</b>	<b>address</b> <b>TEXT</b>	<b>vat</b> <b>INTEGER</b>
1	ProArte	Porto	50123344354
2	MaxiPortas	Lisboa	50134325673



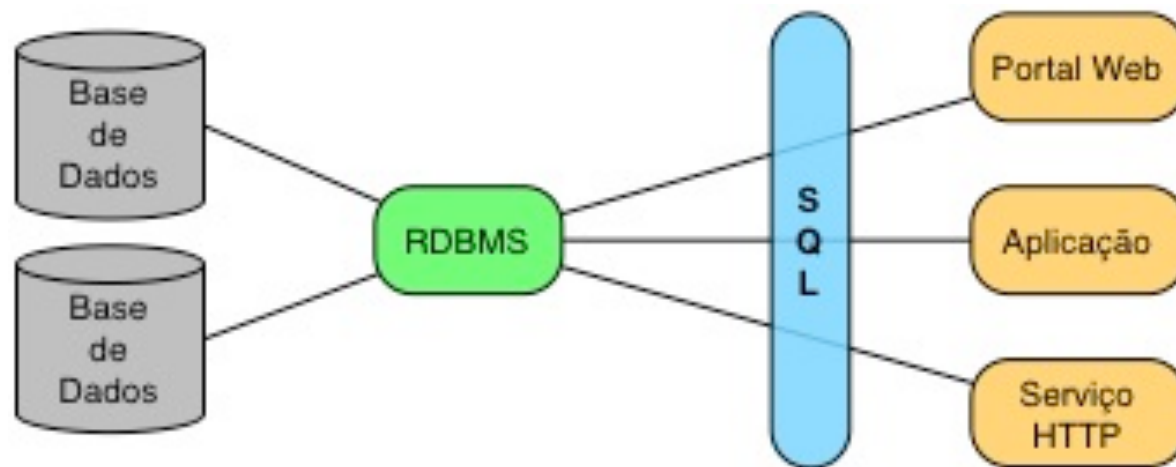
Chinook database

# RDBMS

- Relational DataBase Management System
- Permite acesso a uma base de dados
  - ▣ 1: estabelecimento de uma ligação
  - ▣ 2: execução de comandos
  - ▣ 3: obtenção de resultados (linhas)
  - ▣ 4: término da ligação
- Usa linguagem SQL



# RDBMS



# RDBMS

- Sistemas baseados em Servidores
  - ▣ MySQL, Postgres, Microsoft SQL Server
  - ▣ Acesso via Socket
- Sistemas baseados em Ficheiros
  - ▣ SQLite
  - ▣ Acesso via API
- Todos usam SQL
  - ▣ Com pequenas variações entre eles

# SQL: Structured Query Language

- Comandos para operar sobre uma base de dados relacional
  - ▣ Textual, terminados com ;
- Criada em 1970
  - ▣ É o standard atual
- Diferentes dialectos para cada RDBMS

# CREATE

- ❑ Cria tabelas ou bases de dados
  - ▣ Permite especificar estrutura de uma tabela

```
CREATE DATABASE labi;
```

```
CREATE TABLE contacts(  
    id INTEGER PRIMARY KEY AUTOINCREMENT,  
    name TEXT,  
    address TEXT,  
    phone INTEGER,  
    comp_id INTEGER  
);
```

# INSERT INTO

## □ Insere dados numa tabela

```
INSERT INTO contacts(name, address, phone, comp_id)
VALUES(
    "João Ribeiro Fonseca",
    "Porto",
    934875923,
    1
);
```

# SELECT

- Obtém todos os dados

```
SELECT * FROM contacts;
```

- Apenas alguns campos

```
SELECT name,phone FROM contacts;
```

- Apenas algumas linhas (de uma empresa)

```
SELECT name,phone FROM contacts WHERE comp_id = 1;
```

# SELECT

- Ordena resultados

```
SELECT name,phone FROM contacts  
WHERE comp_id = 1 ORDER BY name ASC;
```

- Resultados com telefones iniciados a 96\*

```
SELECT name,phone FROM contacts  
WHERE phone LIKE "96%";
```

- De múltiplas tabelas (contactos de todas as empresas em aveiro)

```
SELECT contacts.name FROM contacts,companies  
WHERE contacts.comp_id = companies.id  
AND companies.address = "Aveiro";
```

# UPDATE

- Actualiza dados em tabelas
- Todos os contactos estão em Aveiro

```
UPDATE contacts SET address = "Aveiro";
```

- Possível atualizar apenas alguns

```
UPDATE contacts SET address = "Aveiro"  
WHERE comp_id = 1;
```



# DELETE

- Apaga todos os registos

```
DELETE FROM contacts;
```

- Apaga registos de forma selectiva

```
DELETE FROM contacts  
WHERE contacts.comp_id = 1;
```

# DROP

- Apaga tabelas

```
DROP TABLE contacts;
```

- Apaga bases de dados

```
DROP DATABASE labi;
```