

# SQL (*Structured Query Language*)

- Linguagem comercial para BD relacional
  - padrão ISO desde a década de 80
    - SQL (86); SQL-2 (92); SQL-3 (99) e SQL:2003
  - não é apenas uma linguagem de consulta!
    - como o nome sugere...
- Base Formal
  - álgebra relacional e cálculo relacional
- Funcionalidades principais
  - definição (DDL) e manipulação (DML) de dados
  - definição de visões e autorizações de acesso
  - definição de restrições de integridade
  - definição de transações
  - comandos para embutimento em LPs

# SQL - DDL

- Criação de um BD
  - SQL padrão não oferece tal comando
    - BDs são criados via ferramentas do SGBD
  - alguns SGBDs (SQL Server, DB2, MySQL) oferecem este comando
    - `create database nome_BD`
    - `drop database nome_BD`

# SQL - DDL

- Comandos para definição de esquemas

- `create table`

- define a estrutura da tabela, suas restrições de integridade e cria uma tabela vazia

- `alter table`

- modifica a definição de uma tabela (I / E / A atributos; I / E RIs)
    - RIs básicas:
      - atributos chave não podem ser removidos de uma tabela
      - atributos NOT NULL não podem ser inseridos em uma tabela

- `drop table`

- remove uma tabela com todas as suas tuplas

# SQL – Create Table

```
CREATE TABLE nome_tabela (  
  nome_atributo_1 tipo_1 [[NOT]NULL] [UNIQUE]  
  [{, nome_atributo_n      tipo_n}]  
  [, PRIMARY KEY (nome(s)_atributo(s))]  
  [{, FOREIGN KEY (nome_atributo)  
    REFERENCES nome_tabela}]  
)
```

- Principais tipos de dados

- *int*, *smallint*, *tinyint*,  
*numeric(tamanho[,nro\_casas\_decimais])*,  
*char(tamanho)*, *varchar(tamanho)*, *date*, *time*,  
*datetime*, ...
- formato para data e hora
  - “YYYY-MM-DD HH:MM:SS”

# Exemplos de Criação de Tabela

```
CREATE TABLE Ambulatorios (  
    nroa                int,  
    andar               numeric(3) NOT NULL,  
    capacidade          smallint,  
    PRIMARY KEY(nroa)  
)
```

```
CREATE TABLE Medicos (  
    codm               int,  
    nome               varchar(40) NOT NULL,  
    idade              smallint NOT NULL,  
    especialidade     char(20),  
    CPF                numeric(11) UNIQUE,  
    cidade             varchar(30),  
    nroa               int,  
    PRIMARY KEY(codm),  
    FOREIGN KEY(nroa) REFERENCES Ambulatorios  
)
```

# SQL – *Alter Table*

```
ALTER TABLE nome_tabela  
ADD [COLUMN] nome_atributo_1 tipo_1 [{RIs}]  
    [{, nome_atributo_n tipo_n [{RIs}]]  
|  
MODIFY [COLUMN] nome_atributo_1 tipo_1 [{RIs}]  
    [{, nome_atributo_n tipo_n [{RIs}]]  
|  
DROP COLUMN nome_atributo_1  
    [{, nome_atributo_n }]  
|  
ADD CONSTRAINT nome_RI_1 def_RI_1  
    [{, nome_RI_n def_RI_n}]  
|  
DROP CONSTRAINT nome_RI_1  
    [{, nome_RI_n}]  
|  
[ADD | DROP] [PRIMARY KEY ... | FOREIGN KEY ...]
```

# Exemplos de Alteração de Tabelas

```
ALTER TABLE Ambulatórios  
    ADD nome VARCHAR(30)
```

```
ALTER TABLE Médicos DROP PRIMARY KEY
```

```
ALTER TABLE Pacientes DROP COLUMN doença,  
    DROP COLUMN cidade
```

```
ALTER TABLE Funcionários  
    ADD FOREIGN KEY(nroa) REFERENCES Ambulatórios
```

```
ALTER TABLE Funcionarios  
    ADD constraint fk_nroa  
    FOREIGN KEY(nroa) REFERENCES Ambulatorios
```

# SQL – Índices

- Definidos sobre atributos para acelerar consultas a dados
- Índices são definidos automaticamente para chaves primárias
- Operações

```
CREATE [UNIQUE] INDEX nome_índice ON  
nome_tabela (nome_atributo_1[{, nome_atributo_n }])
```

```
DROP INDEX nome_índice ON nome_tabela
```

- Exemplos

```
CREATE UNIQUE INDEX indPac_CPF ON Pacientes (CPF)
```

```
DROP INDEX indPac_CPF ON Pacientes
```



# SQL – DML

- Define operações de manipulação de dados
  - I (INSERT)
  - A (UPDATE)
  - E (DELETE)
  - C (SELECT)
- Instruções declarativas
  - manipulação de conjuntos
  - especifica-se o *que fazer* e não *como fazer*

# SQL – DML

- Inserção de dados

```
INSERT INTO nome_tabela [(lista_atributos)]  
VALUES (lista_valores_atributos)
```

- Exemplos

```
INSERT INTO Ambulatorios VALUES (1, 1, 30)
```

```
INSERT INTO Medicos
```

```
(codm, nome, idade, especialidade, CPF, cidade)
```

```
VALUES (4, 'Carlos', 28, 'ortopedia',  
        11000110000, 'Joinville');
```

# SQL – DML

- Alteração de dados

```
UPDATE nome_tabela  
SET nome_atributo_1 = Valor  
    [{, nome_atributo_n = Valor}]  
[WHERE condição]
```

- Exemplos

```
UPDATE Medicos  
SET cidade = 'Florianopolis'
```

```
UPDATE Ambulatorios  
SET capacidade = capacidade + 5, andar = 3  
WHERE nroa = 2
```

# SQL – DML

- Exclusão de dados

```
DELETE FROM nome_tabela  
[WHERE condição]
```

- Exemplos

```
DELETE FROM Ambulatorios
```

```
DELETE FROM Medicos  
WHERE especialidade = 'cardiologia'  
or cidade < > 'Florianopolis'
```

# Exercícios

1. Crie um BD com nome **Clinica**
2. Crie as seguintes tabelas neste BD, considerando que os atributos sublinhados são chaves primárias e os em itálico são chaves estrangeiras:
  - **Ambulatorios:** nroa (int), andar (numeric(3)) (não nulo), capacidade (smallint)
  - **Medicos:** codm (int), nome (varchar(40)) (não nulo), idade (smallint) (não nulo), especialidade (char(20)), CPF (numeric(11)) (único), cidade (varchar(30)), *nroa* (int)
  - **Pacientes:** codp (int), nome (varchar(40)) (não nulo), idade (smallint) (não nulo), cidade (char(30)), CPF (numeric(11)) (único), doença (varchar(40)) (não nulo)
  - **Funcionarios:** codf (int), nome (varchar(40)) (não nulo), idade (smallint), CPF (numeric(11)) (único), cidade (varchar(30)), salario (numeric(10)), cargo (varchar(20))
  - **Consultas:** codm (int), codp (int), data (date), hora (time)
1. Crie a coluna **nroa (int)** na tabela **Funcionarios**
2. Crie os seguintes índices:
  - Medicos: CPF (único)
  - Pacientes: doença
1. Remover o índice **doenca** em Pacientes
2. Remover as colunas **cargo** e **nroa** da tabela de **Funcionarios**

# Exercícios

## Medicos

Popular as tabelas:

### Ambulatorios

nroa	andar	capacidade
1	1	30
2	1	50
3	2	40
4	2	25
5	2	55

codm	nome	idade	especialidade	CPF	cidade	nroa
1	Joao	40	ortopedia	10000100000	Florianopolis	1
2	Maria	42	traumatologia	10000110000	Blumenau	2
3	Pedro	51	pediatria	11000100000	São José	2
4	Carlos	28	ortopedia	11000110000	Joinville	
5	Marcia	33	neurologia	11000111000	Biguacu	3

### Pacientes

codp	nome	idade	cidade	CPF	doenca
1	Ana	20	Florianopolis	20000200000	gripe
2	Paulo	24	Palhoca	20000220000	fratura
3	Lucia	30	Biguacu	22000200000	tendinite
4	Carlos	28	Joinville	11000110000	sarampo

### Consultas

codm	codp	data	hora
1	1	2010/10/12	14:00
1	4	2010/10/13	10:00
2	1	2010/10/13	9:00
2	2	2010/10/13	11:00
2	3	2010/10/14	14:00
2	4	2010/10/14	17:00
3	1	2010/10/19	18:00
3	3	2010/10/12	10:00
3	4	2010/10/19	13:00
4	4	2010/10/20	13:00
4	4	2010/10/22	19:30

### Funcionarios

codf	nome	idade	cidade	salario	CPF
1	Rita	32	Sao Jose	1200	20000100000
2	Maria	55	Palhoca	1220	30000110000
3	Caio	45	Florianopolis	1100	41000100000
4	Carlos	44	Florianopolis	1200	51000110000
5	Paula	33	Florianopolis	2500	61000111000

# Exercícios (MySQL)

Realizar as seguintes atualizações no BD:

- 1) O paciente Paulo mudou-se para Ilhota
- 2) A consulta do médico 1 com o paciente 4 passou para às 12:00 horas do dia 4 de Novembro de 2010
- 3) A paciente Ana fez aniversário e sua doença agora é cancer
- 4) A consulta do médico Pedro (codf = 3) com o paciente Carlos (codf = 4) passou para uma hora e meia depois
- 5) O funcionário Carlos (codf = 4) deixou a clínica
- 6) As consultas marcadas após as 19 horas foram canceladas
- 7) Os pacientes com câncer ou idade inferior a 10 anos deixaram a clínica
- 8) Os médicos que residem em Biguacu e Palhoca deixaram a clínica