

# Modelo Relacional

Bancos de Dados I

Altigran Soares da Silva

IComp/UFAM - 2018/02

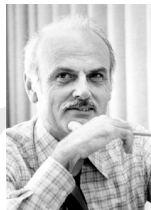
Adaptado dos Slides do Professor Jeffrey Ullman

## Modelo Relacional

- Modelo de dados mais utilizado em BDs
- Serve de base para a grande maioria dos SGBDs existentes hoje no mercado.
- Simples, estrutura uniforme
- Baseado em conceitos matemáticos da teoria de conjuntos.

## Origem

- Criado por Edgar F. Codd



- Mantido e desenvolvido por Chris Date e Hugh Darwen dentre outros



## Modelo Relacional

- *Álgebra relacional*: Operações sobre os BDs
- SQL *Structured Query Language*
  - LDD/LMD padrão
  - Simples e expressiva
  - Padrões
    - SQL2 - 1992
    - SQL:1999
    - SQL:2003
  - Fabricantes implementam dialetos

## Modelo Relacional e Projeto de BD

- Alvo da fase de **Projeto Lógico**
- Usaremos uma variação “neutra” da SQL
- Revisão:
  - ❑ Conceitos Básicos
  - ❑ Restrições de integridade básicas
  - ❑ Restrições de integridade avançadas
  - ❑ SQL LDD e definição de restrições

## Conceitos Básicos

Esquema : Nome, Atributos

Tabelas ou Relações

Instância

Linhas ou  
Tuplas

Produto	
Código	Nome
BAN1	Bancada Gr.
BAN2	Bancada Pq.
SUP1	Buporte Gr.
SUP2	Buporte Pq.

Material		
Código	Descrição	CodSub
123	Parafuso	-
124	Parafuso	-
345	Tampa de madeira	-
346	Tampa de madeira	-
347	Tampa de madeira	-
789	Haste metálica	-
790	Haste metálica	791
791	Haste metálica	-

Insumo		
CodPro	CodMat	Quant
BAN1	123	6
BAN1	124	6
BAN1	345	1
BAN1	789	4
BAN2	123	3
BAN2	124	3
BAN2	346	1
BAN2	789	4
SUP1	123	8
SUP1	347	1
SUP1	790	4
SUP2	124	8
SUP2	347	1
SUP2	790	4

Atributos: Nome, Domínio, Nulos

## Conceitos Básicos

- **Domínios de Atributos:** Conjunto de valores que o atributo pode assumir
- Os domínios comuns:
  - ❑ integer (números inteiros)
  - ❑ char (lista de caracteres)
  - ❑ float (números reais)
  - ❑ date (data)
- Valor indefinido: null
- Valores são atômicos !!

## Conceitos Básicos

- **Chaves** : conjunto de atributos de uma relação que deve ter valor único dentre todas as tuplas.
- SGBD é responsável por garantir a unicidade dos seus valores entre as tuplas.
- Chaves Candidatas: Primária X Alternativas
- Um atributo que faz parte de uma chave é chamado de atributo primo.

## Conceitos Básicos

- Esquema de uma relação:
  - Nome
  - Atributos com os respectivos domínios
  - Chaves
- LDD: Usada para descrever o esquema
- SQL é em parte uma LDD
- Armazenado como *meta-dados* no catálogo
- Consistência:
  - Instâncias “respeitam” o esquema

## SQL LDD

```
CREATE TABLE Projeto
( Nome          VARCHAR(15) NOT NULL,
  Numero        INT          NOT NULL,
  Local         VARCHAR(15),
  PRIMARY KEY   (Numero),
  UNIQUE        (Nome));
```

```
CREATE TABLE Projeto
( Nome          VARCHAR(15) NOT NULL,
  Numero        INT          NOT NULL,
  Local         VARCHAR(15),
  CONSTRAINT ProjPK PRIMARY KEY (Numero),
  CONSTRAINT ProjAK UNIQUE (Nome));
```

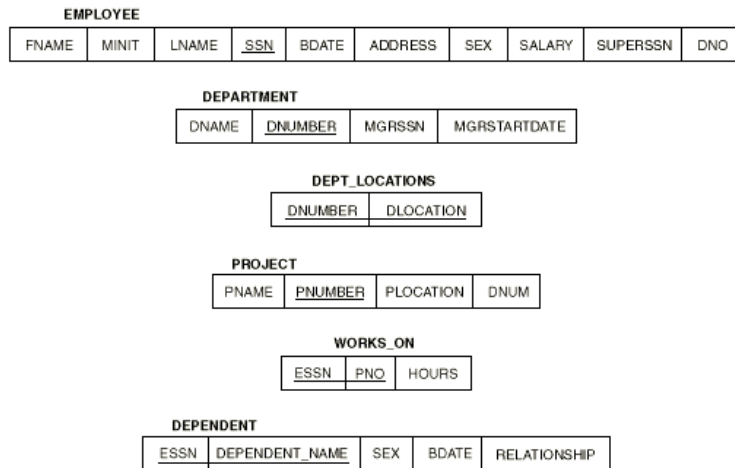
## SQL LDD

```
CREATE TABLE Projeto
( Nome          VARCHAR(15) NOT NULL,
  Numero        INT          NOT NULL,
  Local         VARCHAR(15));
...
ALTER TABLE Projeto
ADD CONSTRAINT ProjPK PRIMARY KEY (Numero);
ALTER TABLE Projeto
ADD CONSTRAINT ProjAK UNIQUE (Nome);
```

## Restrições de Integridade - Nativas

- Restrição de Domínio
  - Todo atributo só assume valores de seu domínio
- Restrição de Chave
  - Não existem duas tuplas com o mesmo valor para uma chave
- Restrição de Entidade
  - Chaves primárias não pode assumir NULL
- Restrição de Valores Nulos
  - Garantia do “NOT NULL”

# Restrições de Integridade Referencial



# Instância de um BD Relacional

EMPLOYEE	FNAME	MINIT	LNAME	SSN	BDATE	ADDRESS	SEX	SALARY	SUPERSSN	DNO
John			Smith	123456789	1965-01-09	731 Fondren, Houston, TX	M	30000	333445555	5
Franklin			Wong	333445555	1955-12-08	638 Voss, Houston, TX	M	40000	888665555	5
Alicia			Zelaya	999887777	1968-01-19	3321 Castle, Spring, TX	F	25000	987654321	4
Jennifer			Wallace	987654321	1941-06-20	291 Berry, Bellaire, TX	F	43000	888665555	4
Ramesh			Narayan	666884444	1962-09-15	975 Fire Oak, Humble, TX	M	38000	333445555	5
Joyce			English	453453453	1972-07-31	5631 Rice, Houston, TX	F	25000	333445555	5
Ahmad			Jabbar	987987987	1969-03-29	980 Dallas, Houston, TX	M	25000	987654321	4
James			Borg	888665555	1937-11-10	450 Stone, Houston, TX	M	55000	null	1

DEPARTMENT	DNAME	DNUMBER	MGRSSN	MGRSTARTDATE
Research		5	333445555	1988-05-22
Administration		4	987654321	1995-01-01
Headquarters		1	888665555	1981-06-19

DEPT_LOCATIONS	DNUMBER	DLOCATION
		Houston
		Stafford
		Bellaire
		Sugarland

WORKS_ON	ESSN	PNO	HOURS
	123456789	1	32.5
	123456789	2	7.5
	666884444	3	40.0
	453453453	1	20.0
	453453453	2	20.0
	333445555	2	10.0
	333445555	3	10.0
	333445555	10	10.0
	333445555	20	10.0
	999887777	30	30.0
	999887777	10	10.0
	987987987	10	35.0
	987987987	30	5.0
	987654321	30	20.0
	987654321	20	15.0
	888665555	20	null

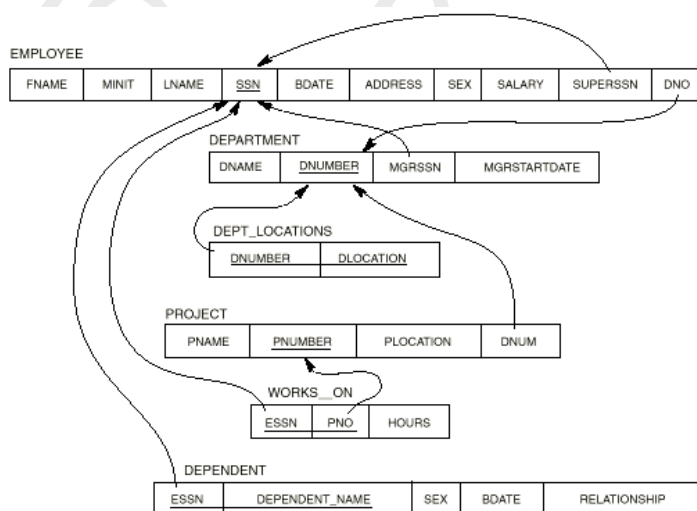
  

PROJECT	PNAME	PNUMBER	PLOCATION	DNUM
	ProductX	1	Bellaire	5
	ProductY	2	Sugarland	5
	ProductZ	3	Houston	5
	Computerization	10	Stafford	4
	Reorganization	20	Houston	1
	Newbenefits	30	Stafford	4

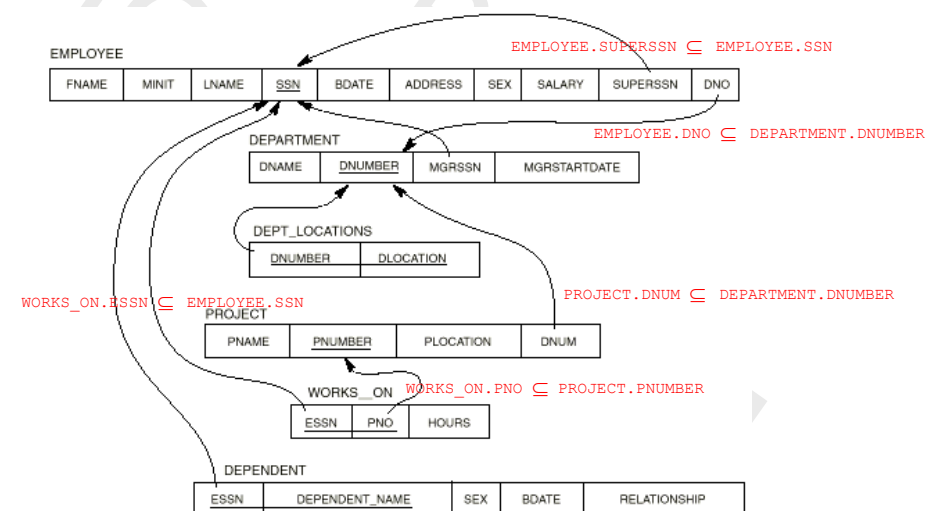
  

DEPENDENT	ESSN	DEPENDENT_NAME	SEX	BDATE	RELATIONSHIP
	333445555	Alice	F	1986-04-05	DAUGHTER
	333445555	Theodore	M	1983-10-25	SON
	333445555	Joy	F	1958-05-03	SPOUSE
	987654321	Abner	M	1942-02-28	SPOUSE
	123456789	Michael	M	1988-01-04	SON
	123456789	Alice	F	1988-12-30	DAUGHTER
	123456789	Elizabeth	F	1967-05-05	SPOUSE

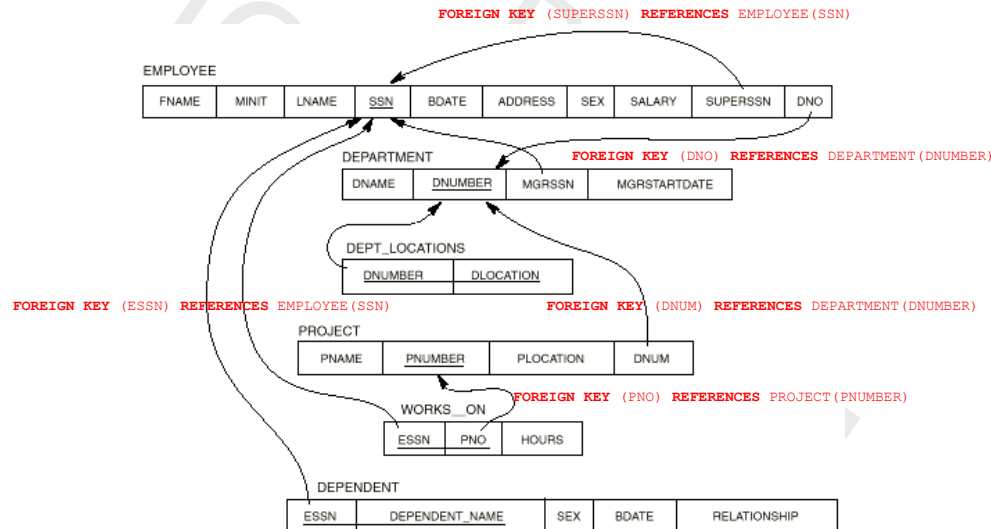
# Restrições de Integridade Referencial



# Restrições de Integridade Referências



# Integridade Referencial em SQL



# Dependências de Inclusão em SQL

EMPLOYEE.DNO  $\subseteq$  DEPARTMENT.DNUMBER

EMPLOYEE.SUPERSSN  $\subseteq$  EMPLOYEE.SSN

PROJECT.DNUM  $\subseteq$  DEPARTMENT.DNUMBER

WORKS\_ON.PNO  $\subseteq$  PROJECT.PNUMBER

WORKS\_ON.PSSN  $\subseteq$  EMPLOYEE.SSN

FOREIGN KEY (DNO) REFERENCES DEPARTMENT (DNUMBER)  
 FOREIGN KEY (SUPERSSN) REFERENCES EMPLOYEE (SSN)  
 FOREIGN KEY (DNUM) REFERENCES DEPARTMENT (DNUMBER)  
 FOREIGN KEY (PNO) REFERENCES PROJECT (PNUMBER)  
 FOREIGN KEY (ESSN) REFERENCES EMPLOYEE (SSN)

# SQL LDD com Integridade Referencial

```
CREATE TABLE EMPLOYEE
( FNAME      VARCHAR(15) NOT NULL ,
  MINIT      CHAR       NOT NULL ,
  LNAME      VARCHAR(15) NOT NULL ,
  SSN        CHAR(9)    NOT NULL ,
  BDATE      DATE       NOT NULL ,
  ADDRESS    VARCHAR(30) ,
  SEX        CHAR       ,
  SALARY     DECIMAL(10,2) ,
  SUPERSSN   CHAR(9)    ,
  DNO        INT        NOT NULL ,
  PRIMARY KEY (SSN) ,
  FOREIGN KEY (SUPERSSN) REFERENCES EMPLOYEE(SSN) ,
  FOREIGN KEY (DNO) REFERENCES DEPARTMENT(DNUMBER) );

CREATE TABLE DEPARTMENT
( DNAME      VARCHAR(15) NOT NULL ,
  DNUMBER    INT         NOT NULL ,
  MGRSSN     CHAR(9)    NOT NULL ,
  MGRSTARTDATE DATE ,
  PRIMARY KEY (DNUMBER) ,
  UNIQUE (DNAME) ,
  FOREIGN KEY (MGRSSN) REFERENCES EMPLOYEE(SSN) );

CREATE TABLE PROJECT
( PNAME      VARCHAR(15) NOT NULL ,
  PNUMBER    INT         NOT NULL ,
  PLOCATION    VARCHAR(15) ,
  DNUM       INT         NOT NULL ,
  PRIMARY KEY (PNUMBER) ,
  UNIQUE (PNAME) ,
  FOREIGN KEY (DNUM) REFERENCES DEPARTMENT(DNUMBER) );

CREATE TABLE WORKS_ON
( ESSN       CHAR(9)    NOT NULL ,
  PNO        INT         NOT NULL ,
  HOURS      DECIMAL(3,1) NOT NULL ,
  PRIMARY KEY (ESSN, PNO) ,
  FOREIGN KEY (ESSN) REFERENCES EMPLOYEE(SSN) ,
  FOREIGN KEY (PNO) REFERENCES PROJECT(PNUMBER) );
```

# Dominios

- Cadeia de caracteres (string)
  - ❑ CHAR(n): string de n caracteres
  - ❑ VARCHAR (n): string de até n caracteres
- Vetor de Bits
  - ❑ BIT(n): bit strings de tamanho n
  - ❑ BIT VARYING(n): bit strings de tamanho até n
- BOOLEAN valores lógicos
  - ❑ Os valores possíveis: TRUE, FALSE, UNKNOWN
- INTEGER, FLOAT, DATE, TIME, ...

## Modificando Esquema

- ALTER TABLE MovieStar ADD phone CHAR(16);
- ALTER TABLE MovieStar DROP birthdate;

21

## Valores padrões (default)

- gender CHAR(1) DEFAULT '?',  
birthdate DATE DEFAULT DATE '0000-00-00'
- ALTER TABLE MovieStar ADD phone CHAR(16) DEFAULT 'unlisted';

22

## Adicionando e Removendo Tuplas

- Inserindo uma tupla simples:  
INSERT INTO StarsIn  
VALUES ('The Maltese Falcon', 1942, 'Sydney Greenstreet');
- Removendo todas as tuplas que satisfazem uma condição:  
DELETE FROM StarsIn  
WHERE movieTitle = 'The Maltese Falcon' AND  
movie Year = 1942 AND  
starName = 'Sydney Greenstreet';

23