

Modelo Relacional

Mestrado Integrado em Engenharia Informática, 3º ano

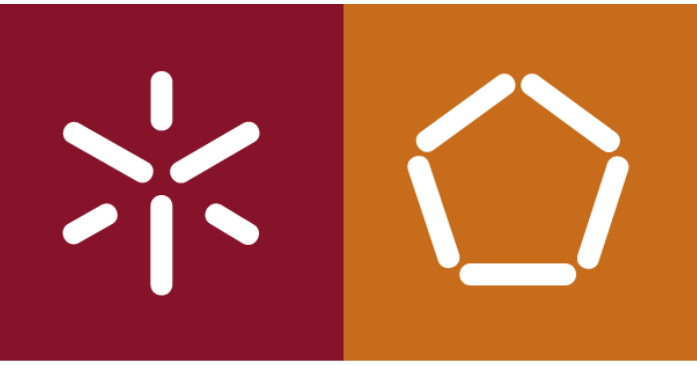
Unidade Curricular de ... Bases de Dados

Aula 3

António Abelha

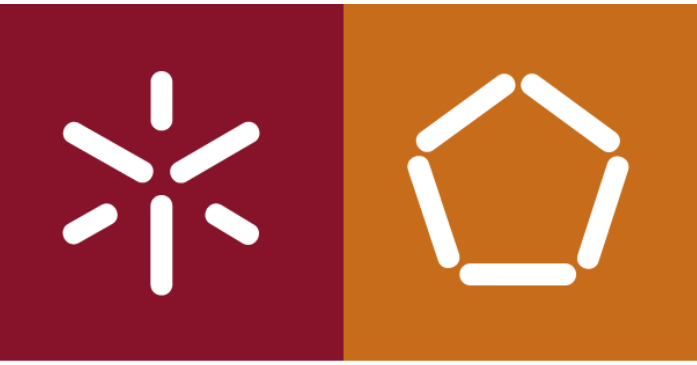
Departamento de Informática

Escola de Engenharia



Modelo Relacional

- A origem do Modelo Relacional.
- A terminologia do Modelo Relacional
- Tabelas para representar dados
- Propriedades das Relações nas Bases de Dados
- Como identificar Chaves (candidatas, primarias, alternativas, e estrangeiras)
- Integridade nas entidades e nas relações
- As *views* no Modelo Relacional



Modelo Relacional

Os Sistema de Gestão de Bases de Dados Relacional (SGDB) têm dominado o mercado de software, com estimativas de vendas de novas licenças de

US \$ 6 milhares de milhões e US \$ 10 milhares de milhões por ano (US \$ 25 milhares de milhões, com ferramentas incluídas). Estas versões de software representam a segunda geração de SGBD e baseiam-se no modelo relacional de dados proposto por E. F. Codd (1970).

No modelo relacional, todos os dados são logicamente estruturados dentro das relações (tabelas). Cada relação tem um nome e é composta por atributos (colunas) de dados, também identificados por um nome. Cada tuplo (linha) contém um valor por atributo.

A principal vantagem do modelo relacional é a sua estrutura lógica simples.



Modelo Relacional

O modelo relacional foi proposto pela primeira vez por EF Codd num artigo '*A relational model of data for large shared data banks*' (Codd, 1970). Este artigo é reconhecido como um marco nos sistemas de bases de dados, apesar de um modelo orientado a conjuntos já ter sido proposto por (Childs, 1968).

Os objetivos do modelo relacional foram especificados da seguinte forma:

- ◆ Permitir um elevado grau de independência de dados.
- ◆ Fornecer motivos para lidar com a semântica dos dados, problemas de consistência e redundância. Em particular, o artigo de Codd introduziu o conceito de relações normalizadas.
- ◆ Permitir a expansão de linguagens de manipulação de dados orientadas para conjuntos.



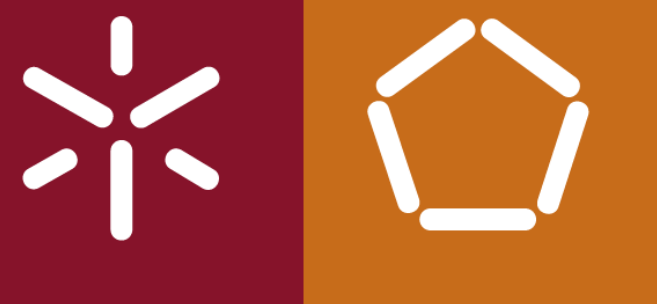
Modelo Relacional

A primeira implementação pelo ***IBM's San José Research Laboratory in California***, foi o protótipo DBMS System R, desenvolvido durante a década de 1970. este projeto foi concebido para provar a viabilidade do modelo relacional.

o projeto do System R conduziu a dois desenvolvimentos importantes:

O desenvolvimento de uma linguagem de consulta estruturada chamada **SQL**, que desde se tornou para a *International Organization for Standardization (ISO)* a linguagem padrão para SGBDs relacionais;

A produção de vários produtos de DBMS relacional comerciais durante o final da década de 1970 e 1980: por exemplo, DB2 e SQL / DS da IBM e Oracle da Oracle Corporation.



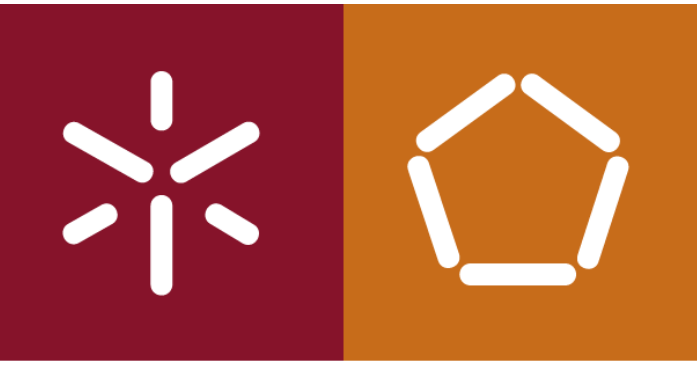
Modelo Relacional

- 1)** Numa base de dados relacional, todos os dados, incluindo o próprio dicionário de dados, são representados de uma só forma, em tabelas bidimensionais.
- 2)** Cada elemento de dados fica bem determinado pela combinação do nome da tabela onde está armazenado, valor da chave primária e respectiva coluna (atributo).
- 3)** Os valores nulos são suportados para representar informação não disponível ou não aplicável, independentemente do domínio dos respectivos atributos.
- 4)** Os metadados são representados e acedidos da mesma forma que os próprios dados.
- 5)** Apesar de um sistema relacional poder suportar várias linguagens, deverá existir pelo menos uma linguagem com as seguintes características:
 - Manipulação de dados, com possibilidade de utilização interactiva ou em programas de aplicação.
 - Definição de dados.
 - Definição de views.
 - Definição de restrições de integridade.
 - Definição de acessos (autorizações).
 - Manipulação de transacções (commit, rollback, etc.).
- 6)** Numa view, todos os dados actualizáveis que forem modificados, devem ver essas modificações traduzidas nas tabelas base.



Modelo Relacional

- 7) Há a capacidade de tratar uma tabela (base ou virtual) como se fosse um simples operando (ou seja, utilização de uma linguagem set-oriented), tanto em operações de consulta como de actualização.
- 8) Alterações na organização física dos ficheiros da base de dados ou nos métodos de acesso a esses ficheiros (nível interno) não devem afectar o nível conceptual – **independência física**.
- 9) Alterações no esquema da base de dados (nível conceptual), que não envolvam remoções de elementos, não devem afectar o nível externo – **independência lógica**.
- 10) As restrições de integridade devem poder ser especificadas numa linguagem relacional, independentemente dos programas de aplicação, e armazenadas no dicionário de dados.
- 11) O facto de uma base de dados estar centralizada numa máquina, ou distribuída por várias máquinas, não deve repercutir-se ao nível da manipulação de dados.
- 12) Se existir no sistema uma linguagem de mais baixo nível (tipo record-oriented), ela não deverá permitir ultrapassar as restrições de integridade e segurança.



Modelo Relacional

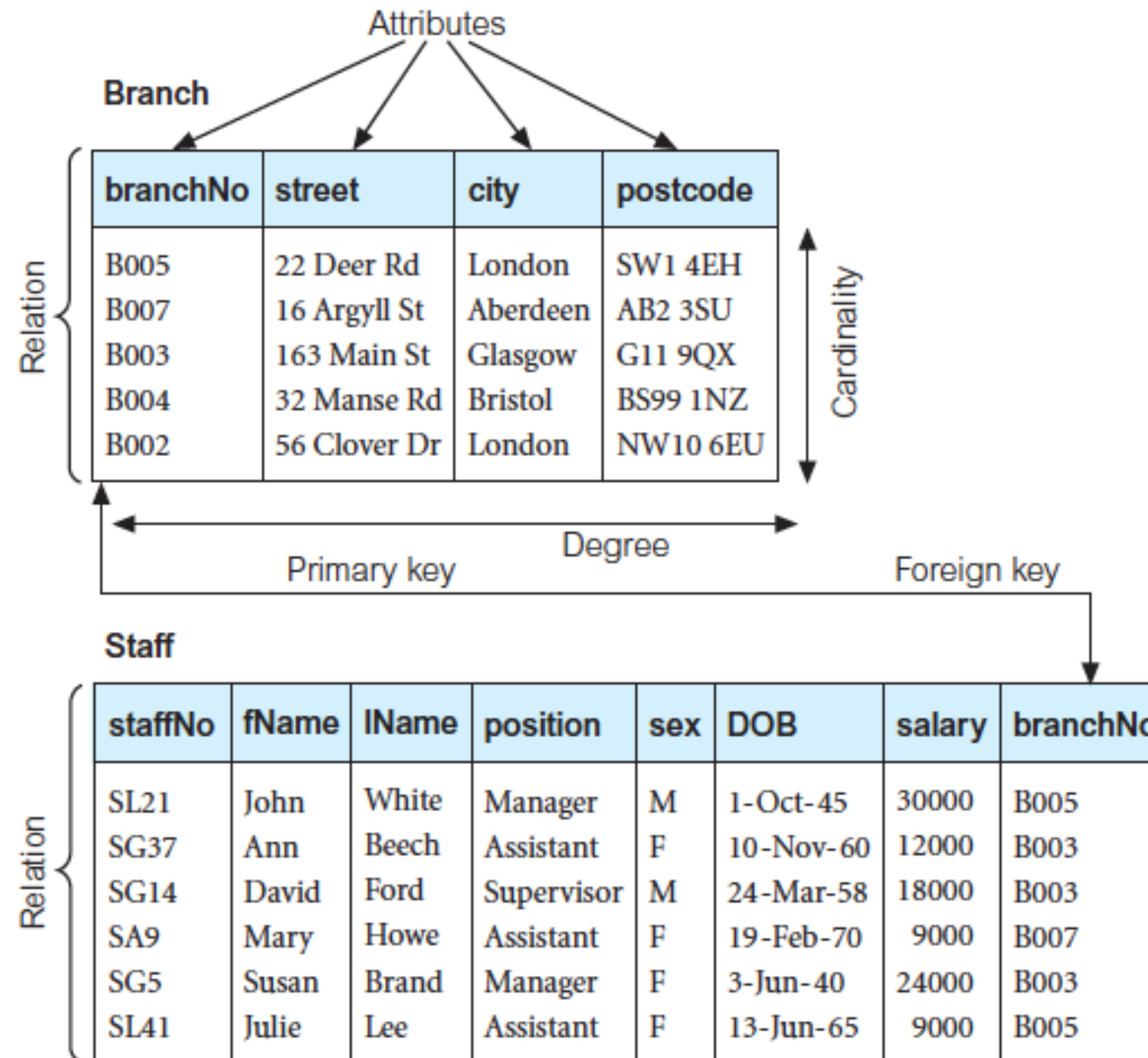
Terminologia associada:

Relação - Uma relação é uma tabela com colunas e linhas

Atributo - Um atributo é uma coluna com nome e que pertence a uma relação.

Domínio - Um domínio é o conjunto de valores permitidos para um ou mais atributos.

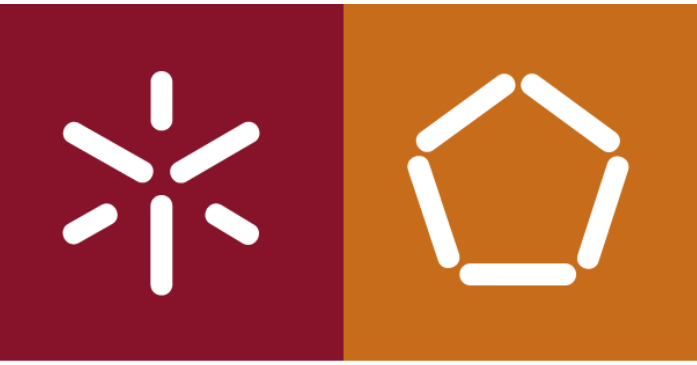
Modelo Relacional





Modelo Relacional

Attribute	Domain Name	Meaning	Domain Definition
branchNo	BranchNumbers	The set of all possible branch numbers	character: size 4, range B001–B999
street	StreetNames	The set of all street names in Britain	character: size 25
city	CityNames	The set of all city names in Britain	character: size 15
postcode	Postcodes	The set of all postcodes in Britain	character: size 8
sex	Sex	The sex of a person	character: size 1, value M or F
DOB	DatesOfBirth	Possible values of staff birth dates	date, range from 1-Jan-20, format dd-mmm-yy
salary	Salaries	Possible values of staff salaries	monetary: 7 digits, range 6000.00–40000.00



Modelo Relacional

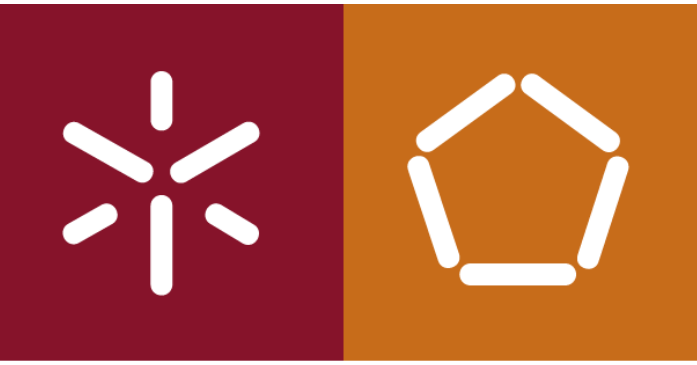
Terminologia associada:

Tuplo - Um tuplo é uma linha de uma relação.

Grau - O grau de uma relação é o número de atributos que ela contém.

Cardinalidade - A cardinalidade de uma relação é o número de tuplos que ela contém.

Base de Dados Relacional - Uma coleção de relações normalizadas com diferentes nomes para as relações.



Modelo Relacional

Esquema da relação - Uma relação é definida por pares de conjuntos de atributos e respectivos domínios.

Esquema de um base de dados relacional - Um conjunto de esquemas da relação, cada um com um nome diferente.

Uma relação tem as seguintes propriedades:

- a relação tem um nome que é diferente de todos os outros nomes de relações do esquema relacional;
- cada célula da relação contém exatamente um valor atômico;
- cada atributo tem um nome diferente;
- os valores de um atributo são todos do mesmo domínio;
- cada tuplo é distinto; não há tuplos duplicadas;
- a ordem dos atributos não tem nenhum significado;
- a ordem de tuplos não tem nenhum significado, teoricamente. (No entanto, na prática, da ordem pode afetar a eficiência)

•



Modelo Relacional

Super Chave - Um atributo ou conjunto de atributos, que identifica unicamente um tuplo dentro de uma relação.

Chave Candidata- É uma super chave tal que nenhum subconjunto, é uma super-chave dentro da relação.

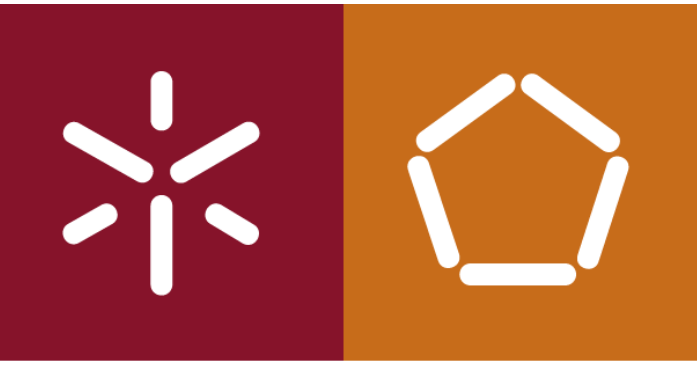
Exemplo: A chave candidata, K , para uma relação R tem duas propriedades:

singularidade - em cada tuplo de R , os valores de K identificam exclusivamente este tuplo;

irreduzibilidade - nenhum subconjunto de K tem a propriedade de exclusividade.

Podemos ter várias chaves candidatas para uma relação. Quando essa chave é constituída por mais do que um

atributo, chamamos-lhe chave composta.



Modelo Relacional

Chave primária - A chave candidata que for selecionado para identificar exclusivamente os duplos de uma relação.

Chave externa - Um atributo ou conjunto de atributos, de uma relação que corresponde à chave candidata de outra (possivelmente a mesma) relação.



Modelo Relacional

Branch

branchNo	street	city	postcode
B005	22 Deer Rd	London	SW1 4EH
B007	16 Argyll St	Aberdeen	AB2 3SU
B003	163 Main St	Glasgow	G11 9QX
B004	32 Manse Rd	Bristol	BS99 1NZ
B002	56 Clover Dr	London	NW10 6EU

Staff

staffNo	fName	IName	position	sex	DOB	salary	branchNo
SL21	John	White	Manager	M	1-Oct-45	30000	B005
SG37	Ann	Beech	Assistant	F	10-Nov-60	12000	B003
SG14	David	Ford	Supervisor	M	24-Mar-58	18000	B003
SA9	Mary	Howe	Assistant	F	19-Feb-70	9000	B007
SG5	Susan	Brand	Manager	F	3-Jun-40	24000	B003
SL41	Julie	Lee	Assistant	F	13-Jun-65	9000	B005

PropertyForRent

propertyNo	street	city	postcode	type	rooms	rent	ownerNo	staffNo	branchNo
PA14	16 Holhead	Aberdeen	AB7 5SU	House	6	650	CO46	SA9	B007
PL94	6 Argyll St	London	NW2	Flat	4	400	CO87	SL41	B005
PG4	6 Lawrence St	Glasgow	G11 9QX	Flat	3	350	CO40		B003
PG36	2 Manor Rd	Glasgow	G32 4QX	Flat	3	375	CO93	SG37	B003
PG21	18 Dale Rd	Glasgow	G12	House	5	600	CO87	SG37	B003
PG16	5 Novar Dr	Glasgow	G12 9AX	Flat	4	450	CO93	SG14	B003



Modelo Relacional

Client

clientNo	fName	IName	telNo	prefType	maxRent
CR76	John	Kay	0207-774-5632	Flat	425
CR56	Aline	Stewart	0141-848-1825	Flat	350
CR74	Mike	Ritchie	01475-392178	House	750
CR62	Mary	Tregear	01224-196720	Flat	600

PrivateOwner

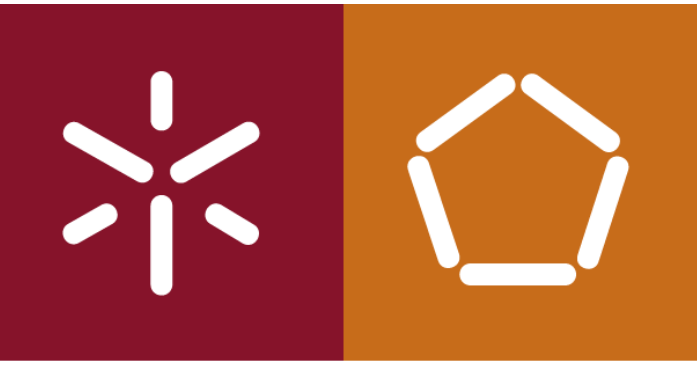
ownerNo	fName	IName	address	telNo
CO46	Joe	Keogh	2 Fergus Dr, Aberdeen AB2 7SX	01224-861212
CO87	Carol	Farrel	6 Achray St, Glasgow G32 9DX	0141-357-7419
CO40	Tina	Murphy	63 Well St, Glasgow G42	0141-943-1728
CO93	Tony	Shaw	12 Park Pl, Glasgow G4 0QR	0141-225-7025

Viewing

clientNo	propertyNo	viewDate	comment
CR56	PA14	24-May-04	too small
CR76	PG4	20-Apr-04	too remote
CR56	PG4	26-May-04	
CR62	PA14	14-May-04	no dining room
CR56	PG36	28-Apr-04	

Registration

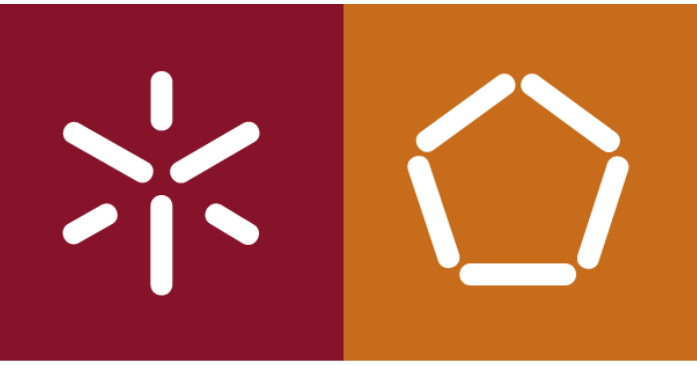
clientNo	branchNo	staffNo	dateJoined
CR76	B005	SL41	2-Jan-04
CR56	B003	SG37	11-Apr-03
CR74	B003	SG37	16-Nov-02
CR62	B007	SA9	7-Mar-03



Modelo Relacional

Branch	(<u>branchNo</u> , street, city, postcode)
Staff	(<u>staffNo</u> , fName, lName, position, sex, DOB, salary, branchNo)
PropertyForRent	(<u>propertyNo</u> , street, city, postcode, type, rooms, rent, ownerNo, staffNo, branchNo)
Client	(<u>clientNo</u> , fName, lName, telNo, prefType, maxRent)
PrivateOwner	(<u>ownerNo</u> , fName, lName, address, telNo)
Viewing	(<u>clientNo</u> , <u>propertyNo</u> , viewDate, comment)
Registration	(<u>clientNo</u> , <u>branchNo</u> , staffNo, dateJoined)

A convenção comum para representar um esquema de relação é utilizar o nome da relação seguido pelos nomes de atributos entre parênteses. Normalmente, a chave primária é sublinhada. Não existe nenhuma notação específica para a chave estrangeira.



Modelo Relacional

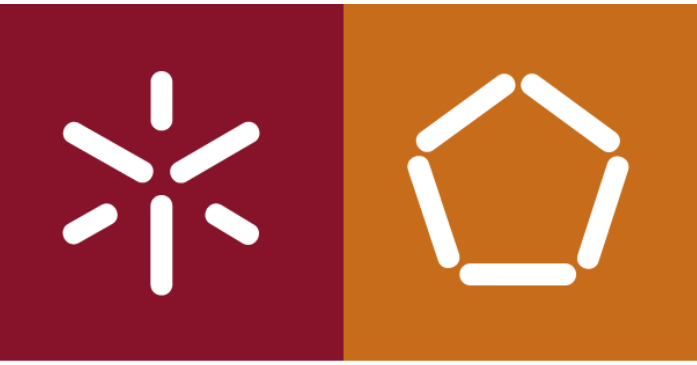
Restrições de Integridade

Null - Representa um valor para um atributo que é actualmente desconhecido ou não é aplicável para este tuplo.

Integridade da entidade - Numa relação base, nenhum atributo de uma chave primária pode ser nulo.

Integridade referencial - Se existe uma chave estrangeira em uma relação, ou o valor de chave estrangeira deve corresponder a um valor da chave candidata de um tuplo dessa relação ou o valor de chave estrangeira deve ser nula.

Restrições gerais - Regras adicionais podem ser especificadas pelos utilizadores ou administradores da base de dados de modo a definirem ou restringem algum aspecto particular da organização.



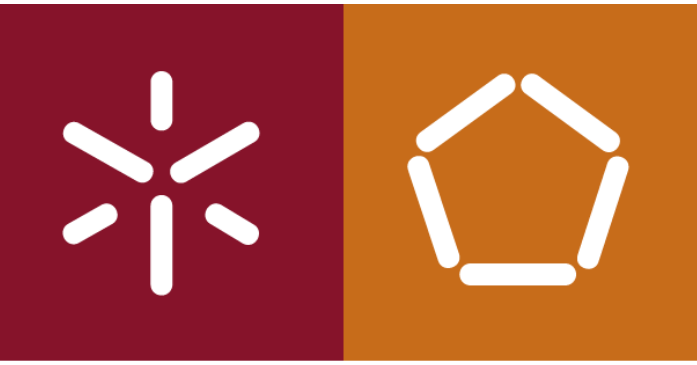
Modelo Relacional

Views (vista)

No modelo relacional o termo *view* é utilizado para definir uma relação virtual ou derivada. A relação não precisa de existir realmente, ela é derivada dinamicamente a partir de uma ou mais relações da base de dados.

Relação de Base - Relação identificada correspondente a uma entidade no esquema conceitual, cujos tuplos estão fisicamente armazenados na base de dados.

View - O resultado dinâmico de uma ou mais operações relacionais operando em relações de base para produzir uma outra relação. A view é uma relação virtual que não tem necessariamente existir na base de dados, mas pode ser produzida a pedido de um determinado utilizador, num dado momento.



Modelo Relacional

Finalidade das views

- Fornece um poderoso e flexível mecanismo de segurança, escondendo partes da base de dados para de determinados utilizadores.
- Permite que os utilizadores acessem aos dados de uma forma personalizada de acordo com as suas necessidades, os mesmos dados podem ser vistos por diferentes utilizadores de diferentes maneiras, ao mesmo tempo.
- Simplificar as operações complexas sobre as relações de base. Por exemplo, se a vista é definido como uma combinação (junção) de duas relações, os utilizadores podem realizar mais operações simples sobre a view.



Modelo Relacional

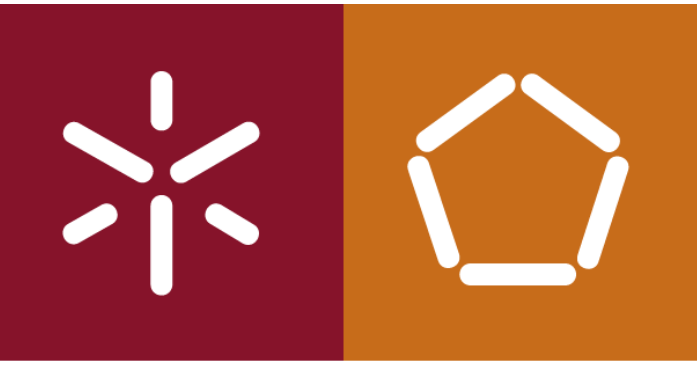
Alterando valores nas Views

Todas as alterações de uma relação de base deve ser imediatamente refletidas em todas as views que fazem referência a essa relação. Da mesma forma, se uma view é atualizada, então a relação de base subjacente deveria refletir essa alteração:

As atualizações são permitidas em views de consulta simples envolvendo uma única relação de base e contendo a chave primária ou uma chave candidata da relação.

As atualizações não são permitidas em views que envolvem múltiplas relações de base.

As atualizações não são permitidas para resultados que envolvam agregações ou agrupamentos de operações.



Modelo Relacional

Modelo lógico para o Modelo Relacional

Criar e validar modelo de dados lógico:

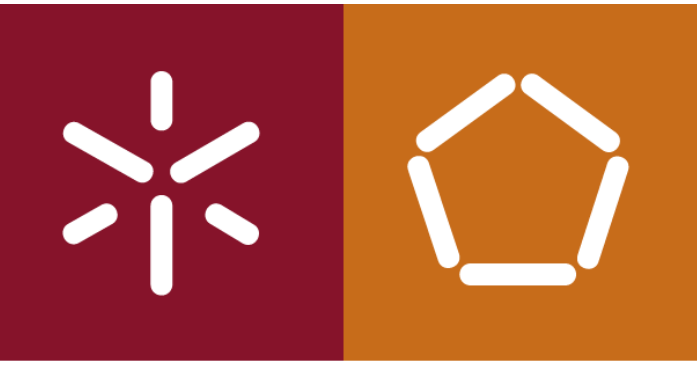
Para traduzir o modelo de dados conceptual num modelo de dados lógico, validar este modelo e verificar se ele é estruturalmente correto e para as operações necessárias.



Modelo Relacional

Modelo lógico para o Modelo Relacional

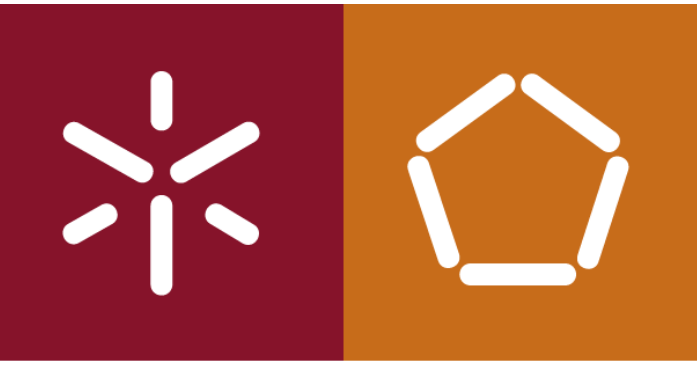
1. Derivar relações para o modelo de dados lógico
2. Validar relações usando a normalização
3. Validar as relações com as transações do utilizador
4. Verificar restrições de integridade
5. Analisar o modelo de dados lógico com o utilizador
6. Juntar modelos de dados lógicos no modelo global (etapa opcional)
7. Verificar crescimento futuro



Modelo Relacional

Derivar relações para modelo de dados lógico

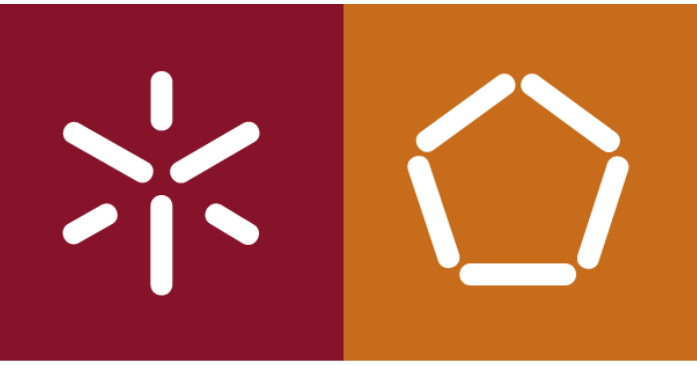
Criar relações para o modelo de dados lógico para representar as entidades, relacionamentos e atributos que foram identificados.



Modelo Relacional

Derivar relações para modelo de dados lógico

- (1) tipos de entidades fortes;
- (2) tipos de entidades fracas;
- (3) tipos de relação binária um-para-muitos (1: *);
- (4) tipos de relação binária um-para-um (1: 1);
- (5) tipos de relação recursiva um-para-um (1: 1);
- (6) tipos de relação de superclasse / subclasse;
- (7) tipos de relação binários many-to-many (*: *);
- (8) tipos de relação complexos;
- (9) atributos multi-valorizados.



Modelo Relacional

- ◆ Connolly, T., Begg, C., **Database Systems, A Practical Approach to Design, Implementation, and Management** , Addison-Wesley, 4a Edição, 2004