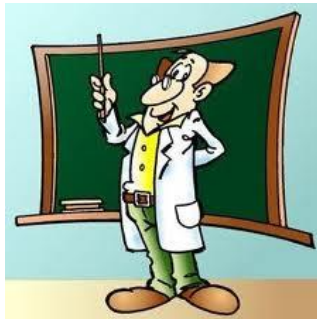




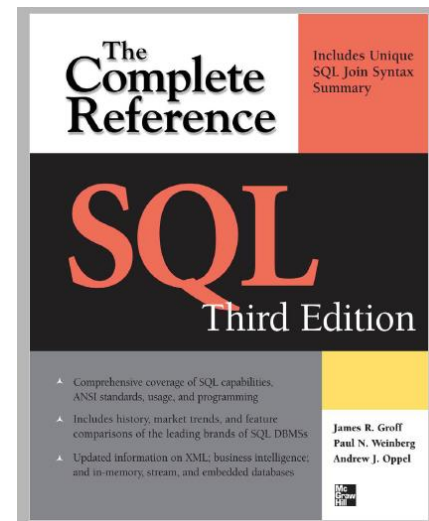
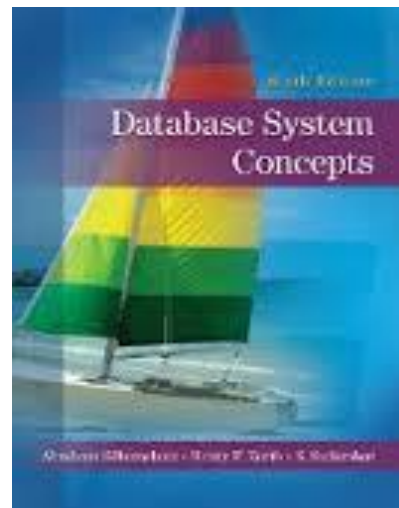
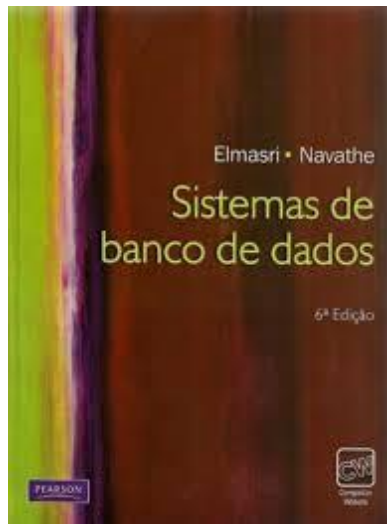
Unidade 6 – Processamento de Queries Multi-Tabelas, Sumarização e SubQueries



Prof. Aparecido V. de Freitas
Doutor em Engenharia
da Computação pela EPUSP
aparecidovfreitas@gmail.com



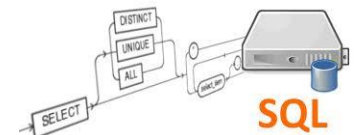
Bibliografia





Consultas Multi-Tabelas

- ✓ Muitas consultas úteis de requisições de dados são obtidas a partir de duas ou mais tabelas;
- ✓ Por exemplo: *Listar todos os pedidos, mostrando o número de pedido e quantidade, o nome e limite de crédito do cliente associado ao pedido.*
- ✓ Nesse exemplo, usaremos a tabela **ORDERS** (pedidos) que contém o número do pedido e quantidade de cada pedido, mas não tem os nomes e limites de crédito do cliente.
- ✓ A tabela **CUSTOMERS** (clientes) contém o nome do cliente e limite de crédito, mas não contém informações de pedidos.

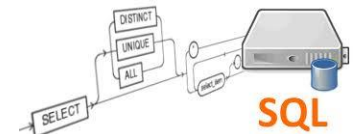
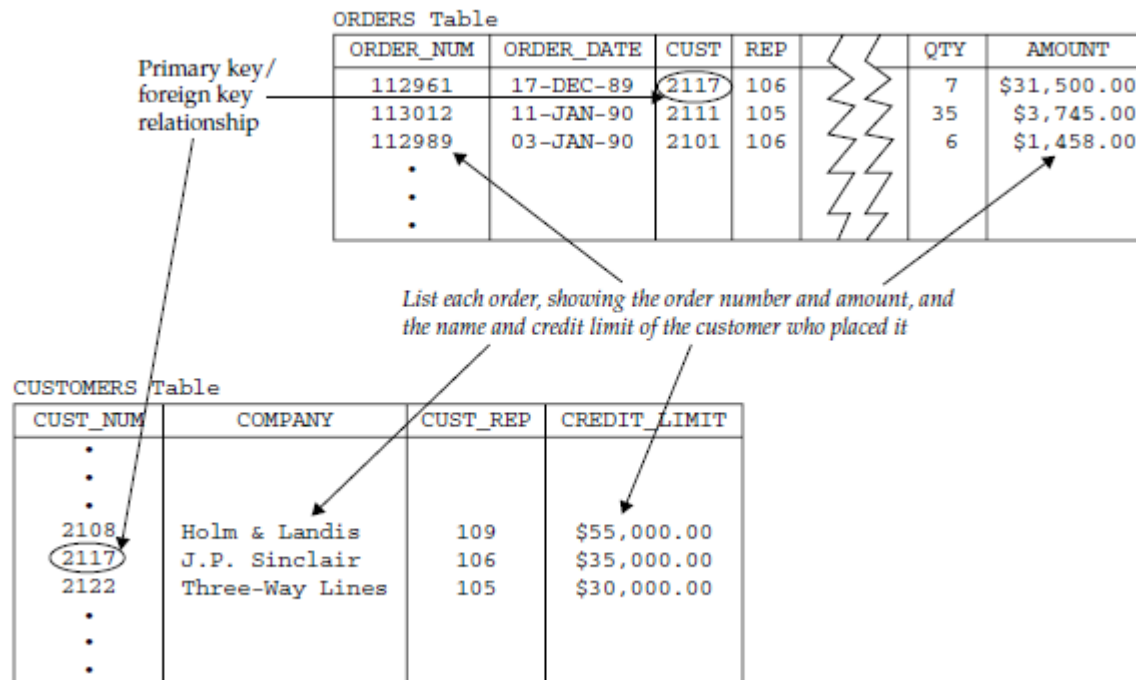


Fonte: The Complete Reference, Groff, Weinberg, Third Edition, McGraw Hill



Consultas Multi-Tabelas

- ✓ *Listar todos os pedidos, mostrando o número de pedido e quantidade, o nome e limite de crédito do cliente associado ao pedido.*



Fonte: The Complete Reference, Groff, Weinberg, Third Edition, McGraw Hill



Consultas Multi-Tabelas em SQL

- ✓ **Joins** são a base para o processamento de consultas multi-tabelas em SQL;
- ✓ **Joins** provêm uma poderosa facilidade para exercitar o relacionamento de dados em um banco de dados;
- ✓ Considerando que em banco de dados relacionais não há pointers ou quaisquer outros mecanismos para relacionamento de tuplas, **joins** correspondem ao único mecanismo disponível para exercitar relacionamentos de dados multi-tabelas.
- ✓ O comando **SELECT** é empregado para as operações **join**. Especifica-se uma condição de pesquisa para se estabelecer o match de colunas.



Banco de Dados Exemplo

PRODUCTS - Contém uma linha para cada tipo de produto que está disponível para venda.

OFFICES - Contém uma linha para cada escritório da organização onde as pessoas trabalham.

SALESREPS - Contém uma linha para cada pessoa da área de vendas.

CUSTOMERS - Contém uma linha para cada cliente da organização.

ORDERS – Contém uma linha para cada pedido colocado por um cliente.
Por simplicidade, cada pedido está associado a um único produto.



```
CREATE TABLE PRODUCTS (  
    MFR_ID CHAR(3) NOT NULL,  
    PRODUCT_ID CHAR(5) NOT NULL,  
    DESCRIPTION VARCHAR(20) NOT NULL,  
    PRICE DECIMAL(9,2) NOT NULL,  
    QTY_ON_HAND INTEGER NOT NULL,  
    PRIMARY KEY (MFR_ID, PRODUCT_ID) );
```

```
CREATE TABLE OFFICES (  
    OFFICE INTEGER NOT NULL,  
    CITY VARCHAR(15) NOT NULL,  
    REGION VARCHAR(10) NOT NULL,  
    MGR INTEGER,  
    TARGET DECIMAL(9,2),  
    SALES DECIMAL(9,2) NOT NULL,  
    PRIMARY KEY (OFFICE) );
```

Fonte: The Complete Reference, Groff, Weinberg, Third Edition, McGraw Hill



```
CREATE TABLE SALESREPS (  
    EMPL_NUM INTEGER NOT NULL,  
        CHECK (EMPL_NUM BETWEEN 101 AND 199),  
    NAME VARCHAR(15) NOT NULL,  
    AGE INTEGER,  
    REP_OFFICE INTEGER,  
    TITLE VARCHAR(10),  
    HIRE_DATE DATE NOT NULL,  
    MANAGER INTEGER,  
    QUOTA DECIMAL(9,2),  
    SALES DECIMAL(9,2) NOT NULL,  
    PRIMARY KEY (EMPL_NUM),  
    FOREIGN KEY (MANAGER)  
    REFERENCES SALESREPS(EMPL_NUM)  
    ON DELETE SET NULL,  
    FOREIGN KEY WORKSIN (REP_OFFICE)  
    REFERENCES OFFICES(OFFICE)  
    ON DELETE SET NULL ) ;
```

Fonte: The Complete Reference, Groff, Weinberg, Third Edition, McGraw Hill



```
CREATE TABLE CUSTOMERS (  
    CUST_NUM INTEGER NOT NULL,  
    COMPANY VARCHAR(20) NOT NULL,  
    CUST_REP INTEGER,  
    CREDIT_LIMIT DECIMAL(9,2),  
    PRIMARY KEY (CUST_NUM),  
    FOREIGN KEY HASREP (CUST_REP)  
    REFERENCES SALESREPS(EMPL_NUM)  
    ON DELETE SET NULL );
```

```
ALTER TABLE OFFICES  
    ADD CONSTRAINT HASMGR  
    FOREIGN KEY (MGR) REFERENCES  
    SALESREPS(EMPL_NUM)  
    ON DELETE SET NULL;
```



```
CREATE TABLE ORDERS (  
    ORDER_NUM INTEGER NOT NULL,  
    CHECK (ORDER_NUM > 100000),  
    ORDER_DATE DATE NOT NULL,  
    CUST INTEGER NOT NULL,  
    REP INTEGER,  
    MFR CHAR(3) NOT NULL,  
    PRODUCT CHAR(5) NOT NULL,  
    QTY INTEGER NOT NULL,  
    AMOUNT DECIMAL(9,2) NOT NULL,  
    PRIMARY KEY (ORDER_NUM),  
    FOREIGN KEY PLACEDBY (CUST)  
    REFERENCES CUSTOMERS(CUST_NUM)  
    ON DELETE CASCADE,  
    FOREIGN KEY TAKENBY (REP)  
    REFERENCES SALESREPS(EMPL_NUM)  
    ON DELETE SET NULL,  
    FOREIGN KEY ISFOR (MFR, PRODUCT)  
    REFERENCES PRODUCTS(MFR_ID, PRODUCT_ID)  
    ON DELETE RESTRICT );
```

Fonte: The Complete Reference, Groff, Weinberg, Third Edition, McGraw Hill



Simple Join (Equi-Join)

✓ Joins que se baseiam num **match exato** entre colunas provenientes de duas tabelas .

✓ *Listar todos os pedidos, mostrando o número de pedido e quantidade, o nome e limite de crédito do cliente associado ao pedido.*

```
SELECT ORDER_NUM, AMOUNT, COMPANY, CREDIT_LIMIT  
FROM ORDERS, CUSTOMERS  
WHERE CUST = CUST_NUM;
```

Resultado da QUERY

order_num	amount	company	credit_limit
112989	1458.00	Jones Mfg.	65000.00
112968	3978.00	First Corp.	65000.00
112963	3276.00	Acme Mfg.	50000.00
112983	702.00	Acme Mfg.	50000.00
112987	27500.00	Acme Mfg.	50000.00
113027	4104.00	Acme Mfg.	50000.00
112993	1896.00	Fred Lewis Corp.	65000.00
113065	2130.00	Fred Lewis Corp.	65000.00
113034	632.00	Ace International	35000.00
113036	22500.00	Ace International	35000.00
113003	5625.00	Holm & Landis	55000.00
113055	150.00	Holm & Landis	55000.00
113058	1480.00	Holm & Landis	55000.00
113069	31350.00	Chen Associates	25000.00
112975	2100.00	JCP Inc.	50000.00
113012	3745.00	JCP Inc.	50000.00
113057	600.00	JCP Inc.	50000.00
113007	2925.00	Zetacorp	50000.00
113045	45000.00	Zetacorp	50000.00
113042	22500.00	Ian & Schmidt	20000.00
112979	15000.00	Orion Corp.	20000.00
113024	7100.00	Orion Corp.	20000.00

Tabela ORDERS

Field	Type	Null	Key	Default	Extra
ORDER_NUM	int(11)	NO	PRI	NULL	
ORDER_DATE	date	NO		NULL	
CUST	int(11)	NO	MUL	NULL	
REP	int(11)	YES	MUL	NULL	
MFR	char(3)	NO	MUL	NULL	
PRODUCT	char(5)	NO		NULL	
QTY	int(11)	NO		NULL	
AMOUNT	decimal(9,2)	NO		NULL	

Tabela CUSTOMERS

Field	Type	Null	Key	Default	Extra
CUST_NUM	int(11)	NO	PRI	NULL	
COMPANY	varchar(20)	NO		NULL	
CUST_REP	int(11)	YES	MUL	NULL	
CREDIT_LIMIT	decimal(9,2)	YES		NULL	

Fonte: Sistemas de Banco de Dados – Elmasri & Navathe – 6ª edição – Editora Pearson



Matching Columns

```
SELECT ORDER_NUM, AMOUNT, COMPANY, CREDIT_LIMIT
FROM ORDERS, CUSTOMERS
WHERE CUST = CUST_NUM;
```

Tabela ORDERS

Field	Type	Null	Key	Default	Extra
ORDER_NUM	int(11)	NO	PRI	NULL	
ORDER_DATE	date	NO		NULL	
CUST	int(11)	NO	MUL	NULL	
REP	int(11)	YES	MUL	NULL	
MFR	char(3)	NO	MUL	NULL	
PRODUCT	char(5)	NO		NULL	
QTY	int(11)	NO		NULL	
AMOUNT	decimal(9,2)	NO		NULL	

Tabela CUSTOMERS

Field	Type	Null	Key	Default	E
CUST_NUM	int(11)	NO	PRI	NULL	
COMPANY	varchar(20)	NO		NULL	
CUST_REP	int(11)	YES	MUL	NULL	
CREDIT_LIMIT	decimal(9,2)	YES		NULL	

✓ *Liste somente as linhas onde CUST na tabela ORDERS coincide com CUST_NUM na tabela CUSTOMERS.*



Simple Join (Equi-Join)

- ✓ Exemplo: **Listar o nome de cada pessoa de Vendas e a cidade e região onde trabalham;**

```
SELECT NAME, CITY, REGION FROM OFFICES, SALESREPS
WHERE OFFICE = REP_OFFICE;
```

Resultado da QUERY

Result Grid			
Filter Rows:			
	NAME	CITY	REGION
▶	Sam Clark	New York	Eastern
	Mary Jones	New York	Eastern
	Dan Roberts	Chicago	Eastern
	Paul Cruz	Chicago	Eastern
	Bob Smith	Chicago	Eastern
	Bill Adams	Atlanta	Eastern
	Sue Smith	Los Angeles	Western
	Larry Fitch	Los Angeles	Western
	Nancy Angelli	Denver	Western

Tabela OFFICES

Field	Type	Null	Key	Default	Extra
OFFICE	int(11)	NO	PRI	NULL	
CITY	varchar(15)	NO		NULL	
REGION	varchar(10)	NO		NULL	
MGR	int(11)	YES	MUL	NULL	
TARGET	decimal(9,2)	YES		NULL	
SALES	decimal(9,2)	NO		NULL	

Tabela SALESREPS

Result Grid					
Filter Rows:					
Field	Type	Null	Key	Default	Extra
EMPL_NUM	int(11)	NO	PRI	NULL	
NAME	varchar(15)	NO		NULL	
AGE	int(11)	YES		NULL	
REP_OFFICE	int(11)	YES	MUL	NULL	
TITLE	varchar(10)	YES		NULL	
HIRE_DATE	date	NO		NULL	
MANAGER	int(11)	YES	MUL	NULL	
QUOTA	decimal(9,2)	YES		NULL	
SALES	decimal(9,2)	NO		NULL	

Fonte: Sistemas de Banco de Dados – Elmasri & Navathe – 6ª edição – Editora Pearson



SELECT NAME, CITY, REGION FROM OFFICES, SALESREPS
WHERE OFFICE = REP_OFFICE;

OFFICES Table

OFFICE	CITY	REGION	MGR	TARGET	SALES
22	Denver	Western	108	\$300,000.00	\$186,042.00
11	New York	Eastern	106	\$575,000.00	\$692,637.00
12	Chicago	Eastern	104	\$800,000.00	\$735,042.00
13	Atlanta	Eastern	NULL	\$350,000.00	\$367,911.00
21	Los Angeles	Western	108	\$725,000.00	\$835,915.00

SALESREPS Table

EMPL_NUM	NAME	AGE	REP_OFFICE	TITLE
105	Bill Adams	37	13	Sales Rep
109	Mary Jones	31	11	Sales Rep
102	Sue Smith	48	21	Sales Rep
106	Sam Clark	52	11	VP Sales
104	Bob Smith	33	12	Sales Mgr
101	Dan Roberts	45	12	Sales Rep
110	Tom Snyder	41	NULL	Sales Rep
108	Larry Fitch	62	21	Sales Mgr
103	Paul Cruz	29	12	Sales Rep
107	Nancy Angell	49	22	Sales Rep

Query Results

NAME	CITY	REGION

Observação:

- ✓ A tabela **SALESREPS** (filha) contém **REP_OFFICE**, uma chave estrangeira para a tabela **OFFICES** (pai).



Simple Join (Equi-Join)

- ✓ Exemplo: **Listar as cidades dos escritórios de vendas e nomes e cargos de seus gerentes.**

```
SELECT CITY , NAME , TITLE FROM OFFICES , SALESREPS
WHERE MGR = EMPL_NUM;
```

Resultado da QUERY

	CITY	NAME	TITLE
▶	New York	Sam Clark	VP Sales
	Chicago	Bob Smith	Sales Mgr
	Atlanta	Bill Adams	Sales Rep
	Los Angeles	Larry Fitch	Sales Mgr
	Denver	Larry Fitch	Sales Mgr

Observação:

- ✓ SQL não requer que as colunas de matching estejam inclusas nos resultados de uma query multi-tabelas.



Tabela OFFICES

Field	Type	Null	Key	Default
OFFICE	int(11)	NO	PRI	NULL
CITY	varchar(15)	NO		NULL
REGION	varchar(10)	NO		NULL
▶ MGR	int(11)	YES	MUL	NULL
TARGET	decimal(9,2)	YES		NULL
SALES	decimal(9,2)	NO		NULL

Tabela SALESREPS

Field	Type	Null	Key	Default
▶ EMPL_NUM	int(11)	NO	PRI	NULL
NAME	varchar(15)	NO		NULL
AGE	int(11)	YES		NULL
REP_OFFICE	int(11)	YES	MUL	NULL
TITLE	varchar(10)	YES		NULL
HIRE_DATE	date	NO		NULL
MANAGER	int(11)	YES	MUL	NULL
QUOTA	decimal(9,2)	YES		NULL
SALES	decimal(9,2)	NO		NULL

Fonte: Sistemas de Banco de Dados – Elmasri & Navathe – 6ª edição – Editora Pearson



Joins com múltiplos Matching de Colunas

- ✓ Exemplo: **Listar todos os números de pedidos acompanhados de totais e descrição dos produtos;**

```
SELECT ORDER_NUM, AMOUNT, DESCRIPTION FROM
ORDERS , PRODUCTS
WHERE MFR = MFR_ID
AND PRODUCT = PRODUCT_ID;
```

Tabela ORDERS

	Field	Type	Null	Key	Default	Extra
►	ORDER_NUM	int(11)	NO	PRI	NULL	
	ORDER_DATE	date	NO		NULL	
	CUST	int(11)	NO	MUL	NULL	
	REP	int(11)	YES	MUL	NULL	
	MFR	char(3)	NO	MUL	NULL	
	PRODUCT	char(5)	NO		NULL	
	QTY	int(11)	NO		NULL	
	AMOUNT	decimal(9,2)	NO		NULL	

Resultado da QUERY

Result Grid			
Filter Rows: <input type="text"/>			
	order_num	amount	description
	112992	760.00	Size 2 Widoet
	113027	4104.00	Size 2 Widoet
	113012	3745.00	Size 3 Widoet
	112963	3276.00	Size 4 Widoet
	112968	3978.00	Size 4 Widoet
	112983	702.00	Size 4 Widoet
	113055	150.00	Widoet Adiuster
	113057	600.00	Widoet Adiuster
	112987	27500.00	Widoet Remover
	112979	15000.00	Widoet Installer
	113036	22500.00	Widoet Installer
	112997	652.00	Handle
	113013	652.00	Handle

Tabela PRODUCTS

	Field	Type	Null	Key	Default
►	MFR_ID	char(3)	NO	PRI	NULL
	PRODUCT_ID	char(5)	NO	PRI	NULL
	DESCRIPTION	varchar(20)	NO		NULL
	PRICE	decimal(9,2)	NO		NULL
	QTY_ON_HAND	int(11)	NO		NULL

Fonte: Sistemas de Banco de Dados – Elmasri & Navathe – 6ª edição – Editora Pearson



Joins com mais de 2 tabelas

- ✓ Exemplo: Listar os números dos pedidos e respectivas quantidades acima de \$25000, incluindo o nome da pessoa de vendas que emitiu o pedido e o nome do cliente que fez o pedido;

```
SELECT ORDER_NUM , AMOUNT , NAME , COMPANY
FROM ORDERS , CUSTOMERS , SALESREPS
WHERE AMOUNT > 25000
AND REP = EMPL_NUM
AND CUST = CUST_NUM;
```

Resultado da QUERY

order_num	amount	name	company
112987	27500.00	Bill Adams	Acme Mfg.
112961	31500.00	Sam Clark	J.P. Sindair
113069	31350.00	Nancy Angelli	Chen Associates
113045	45000.00	Larry Fitch	Zetacorp

Tabela ORDERS

Field	Type	Null	Key	Default	Extra
ORDER_NUM	int(11)	NO	PRI	NULL	
ORDER_DATE	date	NO		NULL	
CUST	int(11)	NO	MUL	NULL	
REP	int(11)	YES	MUL	NULL	
MFR	char(3)	NO	MUL	NULL	
PRODUCT	char(5)	NO		NULL	
QTY	int(11)	NO		NULL	
AMOUNT	decimal(9,2)	NO		NULL	

Tabela CUSTOMERS

Field	Type	Null	Key	Default	Extra
CUST_NUM	int(11)	NO	PRI	NULL	
COMPANY	varchar(20)	NO		NULL	
CUST_REP	int(11)	YES	MUL	NULL	
CREDIT_LIMIT	decimal(9,2)	YES		NULL	

Tabela SALESREPS

Field	Type	Null	Key	Default
EMPL_NUM	int(11)	NO	PRI	NULL
NAME	varchar(15)	NO		NULL
AGE	int(11)	YES		NULL
REP_OFFICE	int(11)	YES	MUL	NULL
TITLE	varchar(10)	YES		NULL
HIRE_DATE	date	NO		NULL
MANAGER	int(11)	YES	MUL	NULL
QUOTA	decimal(9,2)	YES		NULL
SALES	decimal(9,2)	NO		NULL



Equi-Join

- ✓ Quaisquer pares de colunas de duas tabelas podem ser usadas como colunas de matching.
- ✓ Exemplo: *Encontre todos os números dos pedidos, recebidos nos dias em que um novo vendedor foi contratado, ordenados pelo nome do vendedor.*

```
SELECT ORDER_NUM , AMOUNT , ORDER_DATE , NAME
FROM ORDERS , SALESREPS
WHERE ORDER_DATE = HIRE_DATE
ORDER BY NAME;
```

Tabela ORDERS

Field	Type	Null	Key	Default	Extra
ORDER_NUM	int(11)	NO	PRI	NULL	
ORDER_DATE	date	NO		NULL	
CUST	int(11)	NO	MUL	NULL	
REP	int(11)	YES	MUL	NULL	
MFR	char(3)	NO	MUL	NULL	
PRODUCT	char(5)	NO		NULL	
QTY	int(11)	NO		NULL	
AMOUNT	decimal(9,2)	NO		NULL	

Tabela SALESREPS

Field	Type	Null	Key	Default
EMPL_NUM	int(11)	NO	PRI	NULL
NAME	varchar(15)	NO		NULL
AGE	int(11)	YES		NULL
REP_OFFICE	int(11)	YES	MUL	NULL
TITLE	varchar(10)	YES		NULL
HIRE_DATE	date	NO		NULL
MANAGER	int(11)	YES	MUL	NULL
QUOTA	decimal(9,2)	YES		NULL
SALES	decimal(9,2)	NO		NULL

Resultado da QUERY

ORDER_NUM	AMOUNT	ORDER_DATE	NAME
112975	2100.00	2007-10-12	Larry Fitch
112968	3978.00	2007-10-12	Larry Fitch
112979	15000.00	2007-10-12	Larry Fitch
112968	3978.00	2007-10-12	Mary Jones
112979	15000.00	2007-10-12	Mary Jones
112975	2100.00	2007-10-12	Mary Jones

Fonte: Sistemas de Banco de Dados – Elmasri & Navathe – 6ª edição – Editora Pearson



Queries Non-Equi

- ✓ Embora matching (operador equal) de colunas sejam as joins mais comuns, podem-se aplicar outros operadores, por exemplo (>).
- ✓ Exemplo: **Listar todas as combinações de vendedores e escritórios de vendas nos quais as quotas dos vendedores são maiores que o target dos escritórios, ordenados por nome de vendedor.**

```
SELECT NAME , QUOTA , CITY , TARGET
FROM OFFICES , SALESREPS
WHERE QUOTA > TARGET
ORDER BY NAME;
```

Resultado da QUERY

Result Grid	Filter Rows:			
	NAME	quota	CITY	target
▶	Bill Adams	350000.00	Denver	300000.00
	Larry Fitch	350000.00	Denver	300000.00
	Sue Smith	350000.00	Denver	300000.00

Tabela OFFICES

Result Grid	Filter Rows:				
Field	Type	Null	Key	Default	
OFFICE	int(11)	NO	PRI	NULL	
CITY	varchar(15)	NO		NULL	
REGION	varchar(10)	NO		NULL	
▶ MGR	int(11)	YES	MUL	NULL	
TARGET	decimal(9,2)	YES		NULL	
SALES	decimal(9,2)	NO		NULL	

Tabela SALESREPS

Result Grid	Filter Rows:				
Field	Type	Null	Key	Default	
▶ EMPL_NUM	int(11)	NO	PRI	NULL	
NAME	varchar(15)	NO		NULL	
AGE	int(11)	YES		NULL	
REP_OFFICE	int(11)	YES	MUL	NULL	
TITLE	varchar(10)	YES		NULL	
HIRE_DATE	date	NO		NULL	
MANAGER	int(11)	YES	MUL	NULL	
QUOTA	decimal(9,2)	YES		NULL	
SALES	decimal(9,2)	NO		NULL	

Fonte: Sistemas de Banco de Dados – Elmasri & Navathe – 6ª edição – Editora Pearson



Nomes de Colunas Qualificadas

- ✓ Queries de tabelas com nomes duplicados podem causar problemas de ambiguidade;
- ✓ Exemplo: **Listar os nomes, vendas e escritórios de cada vendedor.**

```
SELECT  NAME , SALES, CITY
FROM    OFFICES , SALESREPS
WHERE   OFFICE = REP_OFFICE ;
```

Tabela OFFICES

Field	Type	Null	Key	Default
OFFICE	int(11)	NO	PRI	NULL
CITY	varchar(15)	NO		NULL
REGION	varchar(10)	NO		NULL
MGR	int(11)	YES	MUL	NULL
TARGET	decimal(9,2)	YES		NULL
SALES	decimal(9,2)	NO		NULL

Tabela SALESREPS

Field	Type	Null	Key	Default
EMPL_NUM	int(11)	NO	PRI	NULL
NAME	varchar(15)	NO		NULL
AGE	int(11)	YES		NULL
REP_OFFICE	int(11)	YES	MUL	NULL
TITLE	varchar(10)	YES		NULL
HIRE_DATE	date	NO		NULL
MANAGER	int(11)	YES	MUL	NULL
QUOTA	decimal(9,2)	YES		NULL
SALES	decimal(9,2)	NO		NULL

Resultado da QUERY

Error Code: 1052. Column 'SALES' in field list is ambiguous



Resultado da QUERY

ORA-00918: column ambiguously defined






Nomes de Colunas Qualificadas

- ✓ Queries de tabelas com nomes duplicados podem causar problemas de ambiguidade;
- ✓ Exemplo: **Listar os nomes, vendas e escritórios de cada vendedor.**


```
SELECT  NAME , SALESREPS.SALES, CITY
FROM OFFICES , SALESREPS
WHERE OFFICE = REP_OFFICE ;
```

Resultado da QUERY




name	SALES	city
Sam Clark	299912.00	New York
Marv Jones	392725.00	New York
Dan Roberts	305673.00	Chicago
Paul Cruz	286775.00	Chicago
Bob Smith	142594.00	Chicago
Bill Adams	367911.00	Atlanta
Sue Smith	474050.00	Los Angeles
Larry Fitch	361865.00	Los Angeles
Nancy Angelli	186042.00	Denver

Tabela OFFICES



Field	Type	Null	Key	Default
OFFICE	int(11)	NO	PRI	NULL
CITY	varchar(15)	NO		NULL
REGION	varchar(10)	NO		NULL
MGR	int(11)	YES	MUL	NULL
TARGET	decimal(9,2)	YES		NULL
SALES	decimal(9,2)	NO		NULL

Tabela SALESREPS



Field	Type	Null	Key	Default
EMPL_NUM	int(11)	NO	PRI	NULL
NAME	varchar(15)	NO		NULL
AGE	int(11)	YES		NULL
REP_OFFICE	int(11)	YES	MUL	NULL
TITLE	varchar(10)	YES		NULL
HIRE_DATE	date	NO		NULL
MANAGER	int(11)	YES	MUL	NULL
QUOTA	decimal(9,2)	YES		NULL
SALES	decimal(9,2)	NO		NULL

Fonte: Sistemas de Banco de Dados – Elmasri & Navathe – 6ª edição – Editora Pearson



Seleção de todas colunas

- ✓ Exemplo: **Listar todas informações de vendedores e escritórios onde trabalham;**

```
SELECT *
FROM OFFICES , SALESREPS
WHERE OFFICE = REP_OFFICE ;
```

Resultado da QUERY

Tabela OFFICES

Field	Type	Null	Key	Default
OFFICE	int(11)	NO	PRI	NULL
CITY	varchar(15)	NO		NULL
REGION	varchar(10)	NO		NULL
MGR	int(11)	YES	MUL	NULL
TARGET	decimal(9,2)	YES		NULL
SALES	decimal(9,2)	NO		NULL

Tabela SALESREPS

Field	Type	Null	Key	Default
EMPL_NUM	int(11)	NO	PRI	NULL
NAME	varchar(15)	NO		NULL
AGE	int(11)	YES		NULL
REP_OFFICE	int(11)	YES	MUL	NULL
TITLE	varchar(10)	YES		NULL
HIRE_DATE	date	NO		NULL
MANAGER	int(11)	YES	MUL	NULL
QUOTA	decimal(9,2)	YES		NULL
SALES	decimal(9,2)	NO		NULL

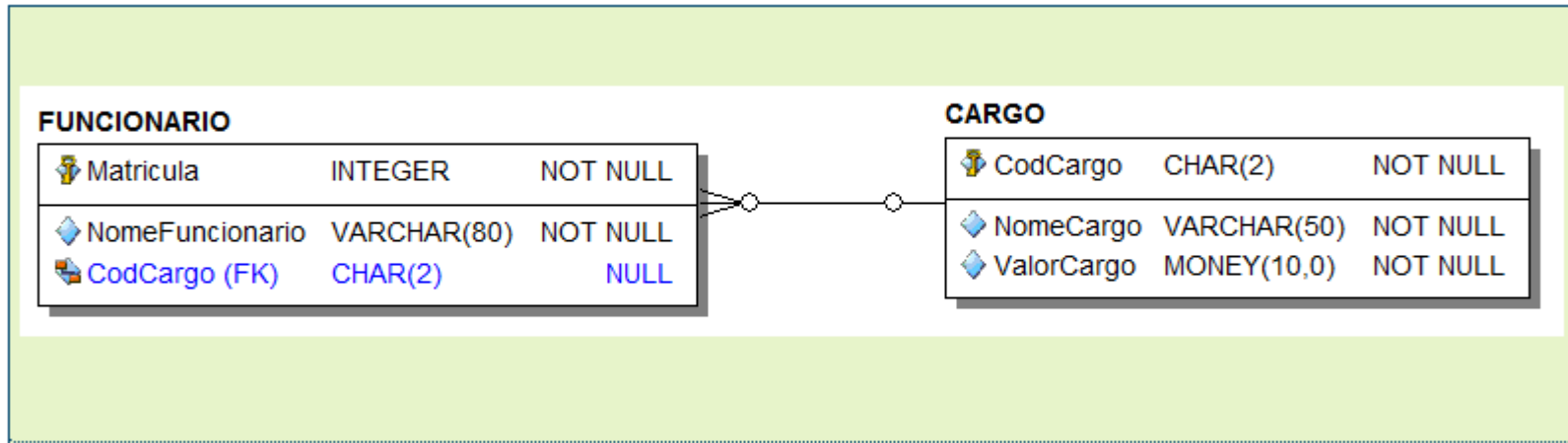
OFFICE	CITY	REGION	MGR	TARGET	SALES	EMPL_NUM	NAME	AGE	REP_OFFICE	TITLE	HIRE_DATE	MANAGER	QUOTA	SALES
11	New York	Eastern	106	575000.00	692637.00	106	Sam Clark	52	11	VP Sales	2006-06-14	106	275000.00	299912.00
11	New York	Eastern	106	575000.00	692637.00	109	Mary Jones	31	11	Sales Rep	2007-10-12	106	300000.00	392725.00
12	Chicago	Eastern	104	800000.00	735042.00	101	Dan Roberts	45	12	Sales Rep	2004-10-20	104	300000.00	305673.00
12	Chicago	Eastern	104	800000.00	735042.00	103	Paul Cruz	29	12	Sales Rep	2005-03-01	104	275000.00	286775.00
12	Chicago	Eastern	104	800000.00	735042.00	104	Bob Smith	33	12	Sales Mgr	2005-05-19	106	200000.00	142594.00
13	Atlanta	Eastern	105	350000.00	367911.00	105	Bill Adams	37	13	Sales Rep	2006-02-12	104	350000.00	367911.00
21	Los Angeles	Western	108	725000.00	835915.00	102	Sue Smith	48	21	Sales Rep	2004-12-10	108	350000.00	474050.00
21	Los Angeles	Western	108	725000.00	835915.00	108	Larry Fitch	62	21	Sales Mgr	2007-10-12	106	350000.00	361865.00
22	Denver	Western	108	300000.00	186042.00	107	Nancy Angelli	49	22	Sales Rep	2006-11-14	108	300000.00	186042.00

Fonte: Sistemas de Banco de Dados – Elmasri & Navathe – 6ª edição – Editora Pearson



INNER, LEFT e RIGHT JOINS

- ✓ Para ilustrar as diferentes opções de JOIN's, considere o seguinte modelo de dados;





INNER, LEFT e RIGHT JOINS

```
CREATE TABLE FUNCIONARIO (  
    matricula integer not null,  
    nome char(50) not null,  
    codcargo char (2) ,  
    primary key (matricula) );
```

```
CREATE TABLE CARGO (  
    codcargo char(2) not null,  
    nomecargo char(50) not null,  
    valorcargo char(40) not null ,  
    primary key(codcargo) );
```

```
ALTER TABLE FUNCIONARIO  
    ADD CONSTRAINT FK_CARGO FOREIGN KEY (codcargo) REFERENCES CARGO (codcargo) ;
```

```
INSERT INTO CARGO VALUES (10, 'PROFESSOR' , 4000 ) ;
```

```
INSERT INTO CARGO VALUES (20, 'MEDICO', 7000 );
```

```
INSERT INTO CARGO VALUES (30, 'ANALISTA', 7500 );
```

```
INSERT INTO FUNCIONARIO VALUES (1000, 'Antonio Carlos' , 10 ) ;
```

```
INSERT INTO FUNCIONARIO VALUES (2000, 'Ana Luiza', 20 );
```

```
INSERT INTO FUNCIONARIO VALUES (3000, 'Pedro Luz', NULL );
```

```
INSERT INTO FUNCIONARIO VALUES (4000, 'Vanderlei Gomes', 20) ;
```




INNER, LEFT e RIGHT JOINS

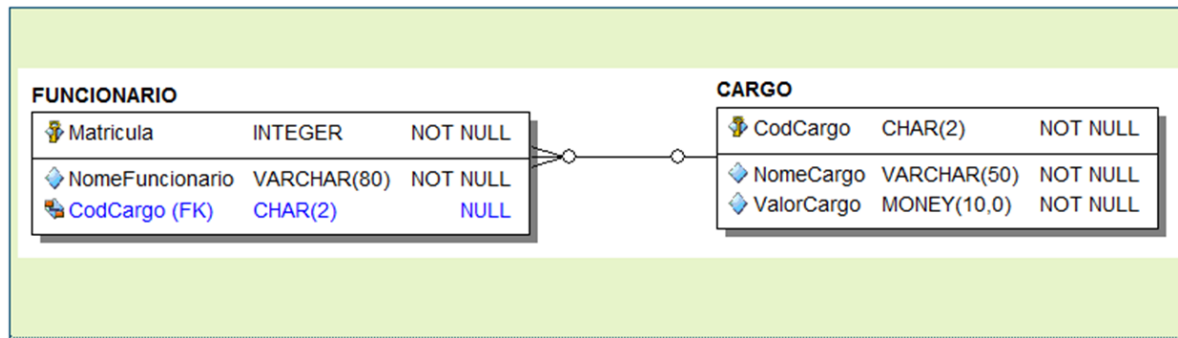


Tabela CARGO

Result Grid				Filter Rows:
	codcargo	nomecargo	valorcargo	
▶	10	PROFESSOR	4000	
	20	MEDICO	7000	
	30	ANALISTA	7500	
★	NULL	NULL	NULL	

Tabela FUNCIONARIO

Result Grid				Filter Rows:
	matricula	nome	codcargo	
▶	1000	Antonio Carlos	10	
	2000	Ana Luiza	20	
	3000	Pedro Luz	NULL	
	4000	Vanderlei Gomes	20	

Fonte: SQL – The Complete Reference – Groff J.R., Weinberg P.N. – Third Edition – McGraw Hill, 2010



INNER JOIN

✓ Junção de 2 tabelas MATCHING de colunas.

```
SELECT F.NOME, C.NOMECargo
FROM CARGO C , FUNCIONARIO F
WHERE C.CODCARGO = F.CODCARGO;
```

Resultado da QUERY

Result Grid			Filter Rows:
	NOME	NOMECargo	
	Antonio Carlos	PROFESSOR	
	Ana Luiza	MEDICO	
	Vanderlei Gomes	MEDICO	

Tabela CARGO

Result Grid				Filter Rows:
	codcargo	nomecargo	valorcargo	
▶	10	PROFESSOR	4000	
	20	MEDICO	7000	
	30	ANALISTA	7500	
★	NULL	NULL	NULL	

Tabela FUNCIONARIO

Result Grid				Filter Rows:
	matricula	nome	codcargo	
▶	1000	Antonio Carlos	10	
	2000	Ana Luiza	20	
	3000	Pedro Luz	NULL	
	4000	Vanderlei Gomes	20	

Fonte: The Complete Reference, Groff, Weinberg, Third Edition, McGraw Hill



INNER JOIN

✓ Junção de 2 tabelas MATCHING de colunas.

```
SELECT  F. NOME AS NOME_FUNCIONARIO, C.NOMECargo AS CARGO
FROM    CARGO C INNER JOIN FUNCIONARIO F
        ON ( C.CODCARGO = F.CODCARGO ) ;
```

Resultado da QUERY

Result Grid			Filter Rows:
	NOME_FUNCIONARIO	CARGO	
▶	Antonio Carlos	PROFESSOR	
	Ana Luiza	MEDICO	
	Vanderlei Gomes	MEDICO	

Tabela CARGO

Result Grid				Filter Rows:
	codcargo	nomecargo	valorcargo	
▶	10	PROFESSOR	4000	
	20	MEDICO	7000	
	30	ANALISTA	7500	
★	NULL	NULL	NULL	

Tabela FUNCIONARIO

Result Grid				Filter Rows:
	matricula	nome	codcargo	
▶	1000	Antonio Carlos	10	
	2000	Ana Luiza	20	
	3000	Pedro Luz	NULL	
	4000	Vanderlei Gomes	20	

Fonte: The Complete Reference, Groff, Weinberg, Third Edition, McGraw Hill



LEFT OUTER JOIN

- ✓ Todas as tuplas da tabela esquerda são mantidas, INCLUINDO os Funcionários sem cargo (null à direita);

```
SELECT  F.NOME AS NOME_FUNCIONARIO, C.NOME CARGO AS CARGO
FROM    FUNCIONARIO F LEFT OUTER JOIN CARGO C
        ON ( F.CODCARGO = C.CODCARGO ) ;
```

Resultado da QUERY

	NOME_FUNCIONARIO	CARGO
▶	Antonio Carlos	PROFESSOR
	Ana Luiza	MEDICO
	Vanderlei Gomes	MEDICO
	Pedro Luz	NULL

Tabela FUNCIONARIO

	matricula	nome	codcargo
▶	1000	Antonio Carlos	10
	2000	Ana Luiza	20
	3000	Pedro Luz	NULL
	4000	Vanderlei Gomes	20

Tabela CARGO

	codcargo	nomecargo	valorcargo
▶	10	PROFESSOR	4000
	20	MEDICO	7000
	30	ANALISTA	7500
★	NULL	NULL	NULL

Fonte: The Complete Reference, Groff, Weinberg, Third Edition, McGraw Hill



RIGHT OUTER JOIN

- ✓ Todas as tuplas da tabela direita são mantidas, INCLUINDO os cargos sem funcionários (null à esquerda);

```
SELECT  F.NOME AS NOME_FUNCIONARIO, C.NOME CARGO AS CARGO
FROM    FUNCIONARIO F RIGHT OUTER JOIN CARGO C
        ON ( F.CODCARGO = C.CODCARGO ) ;
```

Resultado da QUERY

Result Grid		Filter Rows:
	NOME_FUNCIONARIO	CARGO
▶	Antonio Carlos	PROFESSOR
	Ana Luiza	MEDICO
	Vanderlei Gomes	MEDICO
	NULL	ANALISTA

Tabela FUNCIONARIO

Result Grid				Filter Rows:
	matricula	nome	codcargo	
▶	1000	Antonio Carlos	10	
	2000	Ana Luiza	20	
	3000	Pedro Luz	NULL	
	4000	Vanderlei Gomes	20	

Tabela CARGO

Result Grid				Filter Rows:
	codcargo	nomecargo	valorcargo	
▶	10	PROFESSOR	4000	
	20	MEDICO	7000	
	30	ANALISTA	7500	
★	NULL	NULL	NULL	

Fonte: The Complete Reference, Groff, Weinberg, Third Edition, McGraw Hill



Queries de Sumarização

- ✓ SQL permite que dados sejam sumarizados por meio de um conjunto de funções de colunas.
- ✓ Uma função SQL de coluna atua em uma coluna inteira de dados como seu argumento e produz um simples item de dado que sumariza a coluna.
- ✓ Por exemplo, a função de coluna **AVG()** toma uma coluna de dados e computa sua média.
- ✓ **Qual a quota média e a média de vendas de todos os vendedores ?**

```
SELECT  AVG(QUOTA) , AVG(SALES)
        FROM SALESREPS ;
```

Resultado da QUERY

Result Grid			Filter Rows:
	AVG(QUOTA)	AVG(SALES)	
▶	300000.000000	289353.200000	

Tabela SALESREPS

Result Grid						Filter Rows:	Export:
	Field	Type	Null	Key	Default		
▶	EMPL_NUM	int(11)	NO	PRI	NULL		
	NAME	varchar(15)	NO		NULL		
	AGE	int(11)	YES		NULL		
	REP_OFFICE	int(11)	YES	MUL	NULL		
	TITLE	varchar(10)	YES		NULL		
	HIRE_DATE	date	NO		NULL		
	MANAGER	int(11)	YES	MUL	NULL		
	QUOTA	decimal(9,2)	YES		NULL		
	SALES	decimal(9,2)	NO		NULL		



Fonte: The Complete Reference, Groff, Weinberg, Third Edition, McGraw Hill



Funções de Sumarização SQL

- ✓ **SUM()** - computa o total de uma coluna de dados;
- ✓ **AVG()** - computa a média de uma coluna de dados;
- ✓ **MIN()** - computa o menor valor de uma coluna de dados;
- ✓ **MAX()** - computa o maior valor de uma coluna de dados;
- ✓ **COUNT()** - conta a quantidade de valores em uma coluna de dados;
- ✓ **COUNT(*)** - conta a quantidade de tuplas do resultado de uma query.

Fonte: The Complete Reference, Groff, Weinberg, Third Edition, McGraw Hill



Funções de Sumarização

- ✓ Qual a performance média dos vendedores?
- ✓ Obs. performance = $100 * (SALES / QUOTA)$

```
SELECT  AVG(100 * (SALES/QUOTA) )  
FROM    SALESREPS ;
```

Resultado da QUERY

Result Grid		Filter Rows:
	avg(100 * (SALES/QUOTA))	
▶	102.6001495911	

Tabela SALESREPS

Result Grid		Filter Rows:	Export		
	Field	Type	Null	Key	Default
▶	EMPL_NUM	int(11)	NO	PRI	NULL
	NAME	varchar(15)	NO		NULL
	AGE	int(11)	YES		NULL
	REP_OFFICE	int(11)	YES	MUL	NULL
	TITLE	varchar(10)	YES		NULL
	HIRE_DATE	date	NO		NULL
	MANAGER	int(11)	YES	MUL	NULL
	QUOTA	decimal(9,2)	YES		NULL
	SALES	decimal(9,2)	NO		NULL

Fonte: The Complete Reference, Groff, Weinberg, Third Edition, McGraw Hill



Funções de Sumarização

- ✓ Quais os totais de quotas e vendas de todos os vendedores?

```
SELECT SUM(QUOTA) , SUM(SALES)
FROM SALESREPS ;
```

Resultado da QUERY

Result Grid			Filter Rows:
	SUM(QUOTA)	SUM(SALES)	
▶	2700000.00	2893532.00	

Tabela SALESREPS

Result Grid						Filter Rows:	Export:
Field	Type	Null	Key	Default			
▶ EMPL_NUM	int(11)	NO	PRI	NULL			
NAME	varchar(15)	NO		NULL			
AGE	int(11)	YES		NULL			
REP_OFFICE	int(11)	YES	MUL	NULL			
TITLE	varchar(10)	YES		NULL			
HIRE_DATE	date	NO		NULL			
MANAGER	int(11)	YES	MUL	NULL			
QUOTA	decimal(9,2)	YES		NULL			
SALES	decimal(9,2)	NO		NULL			

Obs. Para a função SUM(), os dados da coluna devem ser numéricos.

Fonte: The Complete Reference, Groff, Weinberg, Third Edition, McGraw Hill



Funções de Sumarização

- ✓ Quantos pedidos o vendedor Bill Adams colocou ?
- ✓ Listar esses pedidos.
- ✓ Qual o valor total desses pedidos ?

```
SELECT COUNT(ORDER_NUM)
FROM ORDERS , SALESREPS
WHERE EMPL_NUM = REP
AND NAME = 'BILL ADAMS';
```

```
SELECT ORDERS.*
FROM ORDERS , SALESREPS
WHERE EMPL_NUM = REP
AND NAME = 'BILL ADAMS';
```

```
SELECT SUM(AMOUNT)
FROM ORDERS , SALESREPS
WHERE EMPL_NUM = REP
AND NAME = 'BILL ADAMS';
```

Tabela ORDERS

	Field	Type	Null	Key	Default	Extra
▶	ORDER_NUM	int(11)	NO	PRI	<input type="text" value="NULL"/>	
	ORDER_DATE	date	NO		<input type="text" value="NULL"/>	
	CUST	int(11)	NO	MUL	<input type="text" value="NULL"/>	
	REP	int(11)	YES	MUL	<input type="text" value="NULL"/>	
	MFR	char(3)	NO	MUL	<input type="text" value="NULL"/>	
	PRODUCT	char(5)	NO		<input type="text" value="NULL"/>	
	QTY	int(11)	NO		<input type="text" value="NULL"/>	
	AMOUNT	decimal(9,2)	NO		<input type="text" value="NULL"/>	

Tabela SALESREPS

Result Grid		Filter Rows:	<input type="text"/>		Export:
	Field	Type	Null	Key	Default
▶	EMPL_NUM	int(11)	NO	PRI	<input type="text" value="NULL"/>
	NAME	varchar(15)	NO		<input type="text" value="NULL"/>
	AGE	int(11)	YES		<input type="text" value="NULL"/>
	REP_OFFICE	int(11)	YES	MUL	<input type="text" value="NULL"/>
	TITLE	varchar(10)	YES		<input type="text" value="NULL"/>
	HIRE_DATE	date	NO		<input type="text" value="NULL"/>
	MANAGER	int(11)	YES	MUL	<input type="text" value="NULL"/>
	QUOTA	decimal(9,2)	YES		<input type="text" value="NULL"/>
	SALES	decimal(9,2)	NO		<input type="text" value="NULL"/>

Fonte: The Complete Reference, Groff, Weinberg, Third Edition, McGraw Hill



Funções de Sumarização

- ✓ Quantos pedidos o vendedor Bill Adams colocou ?
- ✓ Listar esses pedidos.
- ✓ Qual o valor total desses pedidos ?

Resultados da QUERY

Result Grid	
count(order_num)	
▶	5

Result Grid	
SUM(AMOUNT)	
▶	39327.00






Result Grid			 Filter Rows: <input type="text"/>	Export: 	Wrap Cell Content: 			
	ORDER_NUM	ORDER_DATE	CUST	REP	MFR	PRODUCT	QTY	AMOUNT
	112963	2007-12-17	2103	105	ACI	41004	28	3276.00
	112983	2007-12-27	2103	105	ACI	41004	6	702.00
	112987	2007-12-31	2103	105	ACI	4100Y	11	27500.00
	113012	2008-01-11	2111	105	ACI	41003	35	3745.00
	113027	2008-01-22	2103	105	ACI	41002	54	4104.00

Tabela ORDERS

Field	Type	Null	Key	Default	Extra
▶ ORDER_NUM	int(11)	NO	PRI	NULL	
ORDER_DATE	date	NO		NULL	
CUST	int(11)	NO	MUL	NULL	
REP	int(11)	YES	MUL	NULL	
MFR	char(3)	NO	MUL	NULL	
PRODUCT	char(5)	NO		NULL	
QTY	int(11)	NO		NULL	
AMOUNT	decimal(9,2)	NO		NULL	

Tabela SALESREPS

Result Grid		Filter Rows:	Export:		
	Field	Type	Null	Key	Default
▶	EMPL_NUM	int(11)	NO	PRI	NULL
	NAME	varchar(15)	NO		NULL
	AGE	int(11)	YES		NULL
	REP_OFFICE	int(11)	YES	MUL	NULL
	TITLE	varchar(10)	YES		NULL
	HIRE_DATE	date	NO		NULL
	MANAGER	int(11)	YES	MUL	NULL
	QUOTA	decimal(9,2)	YES		NULL
	SALES	decimal(9,2)	NO		NULL

Fonte: The Complete Reference, Groff, Weinberg, Third Edition, McGraw Hill



Funções de Sumarização

✓ Qual o preço médio dos produtos da empresa ACI?

```
SELECT AVG(PRICE)
FROM PRODUCTS
WHERE MFR_ID = 'ACI' ;
```

Resultado da QUERY

Result Grid	
	AVG(PRICE)
▶	804.285714

Tabela PRODUCTS

Result Grid					
Filter Rows:					
Export:					
	Field	Type	Null	Key	Default
▶	MFR_ID	char(3)	NO	PRI	NULL
	PRODUCT_ID	char(5)	NO	PRI	NULL
	DESCRIPTION	varchar(20)	NO		NULL
	PRICE	decimal(9,2)	NO		NULL
	QTY_ON_HAND	int(11)	NO		NULL

Fonte: The Complete Reference, Groff, Weinberg, Third Edition, McGraw Hill



Funções de Sumarização

- ✓ Calcule o valor total médio dos pedidos emitidos pelo cliente 2103.

```
SELECT  AVG(AMOUNT)
FROM    ORDERS
WHERE   CUST = 2103;
```

Resultado da QUERY

Result Grid	
	AVG(AMOUNT)
▶	8895.500000

Tabela ORDERS

Field	Type	Null	Key	Default	Extra
ORDER_NUM	int(11)	NO	PRI	NULL	
ORDER_DATE	date	NO		NULL	
CUST	int(11)	NO	MUL	NULL	
REP	int(11)	YES	MUL	NULL	
MFR	char(3)	NO	MUL	NULL	
PRODUCT	char(5)	NO		NULL	
QTY	int(11)	NO		NULL	
AMOUNT	decimal(9,2)	NO		NULL	

Fonte: The Complete Reference, Groff, Weinberg, Third Edition, McGraw Hill



Funções de Sumarização

- ✓ Quais as maiores e menores quotas de vendas atribuídas aos vendedores ?

```
SELECT  MAX(QUOTA) , MIN(QUOTA)
        FROM SALESREPS ;
```

Resultado da QUERY

	max(quota)	min(quota)
▶	350000.00	200000.00

Tabela SALESREPS

Result Grid		Filter Rows:		Exports:	
	Field	Type	Null	Key	Default
▶	EMPL_NUM	int(11)	NO	PRI	NULL
	NAME	varchar(15)	NO		NULL
	AGE	int(11)	YES		NULL
	REP_OFFICE	int(11)	YES	MUL	NULL
	TITLE	varchar(10)	YES		NULL
	HIRE_DATE	date	NO		NULL
	MANAGER	int(11)	YES	MUL	NULL
	QUOTA	decimal(9,2)	YES		NULL
	SALES	decimal(9,2)	NO		NULL

Fonte: The Complete Reference, Groff, Weinberg, Third Edition, McGraw Hill



Funções de Sumarização

- ✓ Qual o pedido mais antigo registrado no banco de dados ?

```
SELECT MIN(ORDER_DATE)
FROM ORDERS;
```

Resultado da QUERY

Result Grid		
	min(order_date)	
▶	2007-01-04	

Tabela ORDERS

	Field	Type	Null	Key	Default	Extra
▶	ORDER_NUM	int(11)	NO	PRI	NULL	
	ORDER_DATE	date	NO		NULL	
	CUST	int(11)	NO	MUL	NULL	
	REP	int(11)	YES	MUL	NULL	
	MFR	char(3)	NO	MUL	NULL	
	PRODUCT	char(5)	NO		NULL	
	QTY	int(11)	NO		NULL	
	AMOUNT	decimal(9,2)	NO		NULL	



Funções de Sumarização

- ✓ Qual a melhor performance de vendas obtida por um vendedor ?

```
SELECT  MAX(100 * SALES /QUOTA)
FROM    SALESREPS;
```

Resultado da QUERY

Result Grid		Filter Rows
	MAX(100 * SALES /QUOTA)	
▶	135.442857	

Tabela SALESREPS

Result Grid		Filter Rows:			Export:
	Field	Type	Null	Key	Default
▶	EMPL_NUM	int(11)	NO	PRI	NULL
	NAME	varchar(15)	NO		NULL
	AGE	int(11)	YES		NULL
	REP_OFFICE	int(11)	YES	MUL	NULL
	TITLE	varchar(10)	YES		NULL
	HIRE_DATE	date	NO		NULL
	MANAGER	int(11)	YES	MUL	NULL
	QUOTA	decimal(9,2)	YES		NULL
	SALES	decimal(9,2)	NO		NULL



Funções de Sumarização

- ✓ Quantos clientes existem cadastrados no banco de dados?

```
SELECT COUNT(CUST_NUM)
FROM CUSTOMERS;
```

Resultado da QUERY

Result Grid	
	COUNT(CUST_NUM)
▶	21

Tabela CUSTOMERS

Result Grid						
		Filter Rows:			Export:	
	Field	Type	Null	Key	Default	Ex
▶	CUST_NUM	int(11)	NO	PRI	NULL	
	COMPANY	varchar(20)	NO		NULL	
	CUST_REP	int(11)	YES	MUL	NULL	
	CREDIT_LIMIT	decimal(9,2)	YES		NULL	



Funções de Sumarização

- ✓ Quantos vendedores estão com vendas acima da quota?

```
SELECT COUNT(NAME)
FROM SALESREPS
WHERE SALES > QUOTA ;
```

Resultado da QUERY

Result Grid	
	COUNT(name)
▶	7

Tabela SALESREPS

Result Grid		Filter Rows:	Export:		
	Field	Type	Null	Key	Default
▶	EMPL_NUM	int(11)	NO	PRI	<input type="text" value="NULL"/>
	NAME	varchar(15)	NO		<input type="text" value="NULL"/>
	AGE	int(11)	YES		<input type="text" value="NULL"/>
	REP_OFFICE	int(11)	YES	MUL	<input type="text" value="NULL"/>
	TITLE	varchar(10)	YES		<input type="text" value="NULL"/>
	HIRE_DATE	date	NO		<input type="text" value="NULL"/>
	MANAGER	int(11)	YES	MUL	<input type="text" value="NULL"/>
	QUOTA	decimal(9,2)	YES		<input type="text" value="NULL"/>
	SALES	decimal(9,2)	NO		<input type="text" value="NULL"/>



Funções de Sumarização

- ✓ Quantos pedidos estão com valor total de vendas acima de 25.000?

```
SELECT COUNT(AMOUNT)
FROM ORDERS
WHERE AMOUNT > 25000;
```

Tabela ORDERS

	Field	Type	Null	Key	Default	Extra
▶	ORDER_NUM	int(11)	NO	PRI	NULL	
	ORDER_DATE	date	NO		NULL	
	CUST	int(11)	NO	MUL	NULL	
	REP	int(11)	YES	MUL	NULL	
	MFR	char(3)	NO	MUL	NULL	
	PRODUCT	char(5)	NO		NULL	
	QTY	int(11)	NO		NULL	
	AMOUNT	decimal(9,2)	NO		NULL	

Resultado da QUERY

Result Grid	
	COUNT(amount)
▶	4

Fonte: The Complete Reference, Groff, Weinberg, Third Edition, McGraw Hill



Funções de Sumarização

- ✓ Quantos pedidos estão com valor total de vendas acima de 25.000?

```
SELECT COUNT(*)
FROM ORDERS
WHERE AMOUNT > 25000;
```

Resultado da QUERY

Result Grid		Filter Row
	COUNT(*)	
	4	

Tabela ORDERS

Field	Type	Null	Key	Default	Extra
ORDER_NUM	int(11)	NO	PRI	NULL	
ORDER_DATE	date	NO		NULL	
CUST	int(11)	NO	MUL	NULL	
REP	int(11)	YES	MUL	NULL	
MFR	char(3)	NO	MUL	NULL	
PRODUCT	char(5)	NO		NULL	
QTY	int(11)	NO		NULL	
AMOUNT	decimal(9,2)	NO		NULL	

- ✓ Obs. SQL suporta uma função de coluna especial (*) que conta linhas ao invés de valores de dados.



Queries agrupadas

- ✓ É frequentemente conveniente sumarizar-se resultados de queries por sub-totais.
- ✓ Essa capacidade é providenciada pela cláusula **GROUP BY** do comando **SELECT**.
- ✓ Exemplo: Qual a média de total de pedido colocado por cada vendedor ?

```
SELECT REP , AVG(AMOUNT)
FROM ORDERS
GROUP BY REP ;
```

Resultado da QUERY

Result Grid		
	REP	AVG(AMOUNT)
▶	101	8876.000000
	102	5694.000000
	103	1350.000000
	105	7865.400000
	106	16479.000000
	107	11477.333333
	108	8376.142857
	109	3552.500000
	110	11566.000000

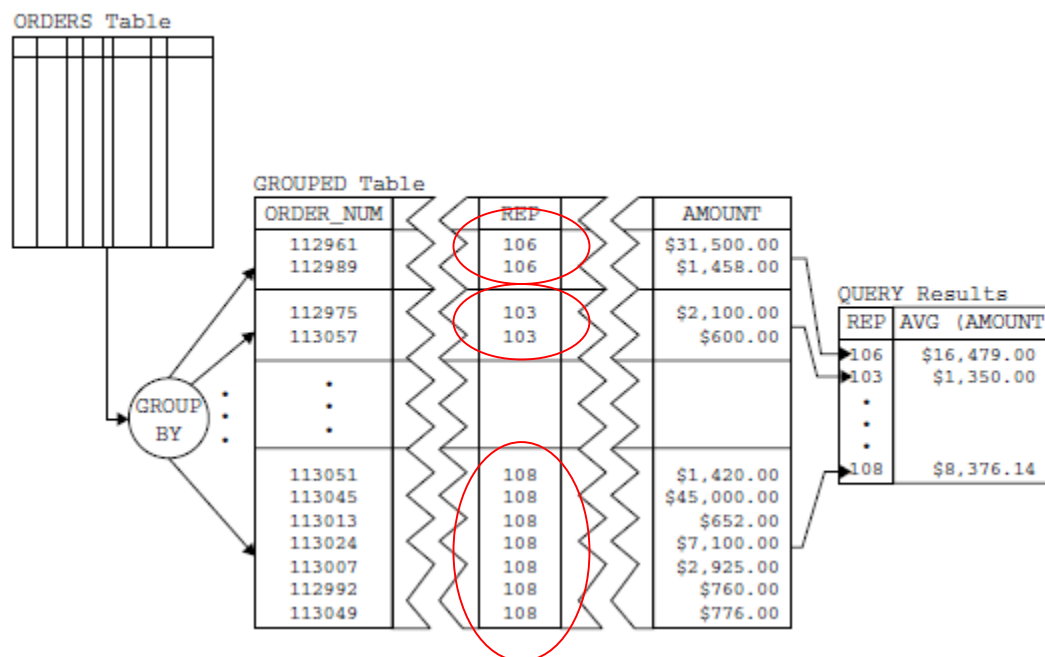
Tabela ORDERS

	Field	Type	Null	Key	Default	Extra
▶	ORDER_NUM	int(11)	NO	PRI	NULL	
	ORDER_DATE	date	NO		NULL	
	CUST	int(11)	NO	MUL	NULL	
	REP	int(11)	YES	MUL	NULL	
	MFR	char(3)	NO	MUL	NULL	
	PRODUCT	char(5)	NO		NULL	
	QTY	int(11)	NO		NULL	
	AMOUNT	decimal(9,2)	NO		NULL	



Queries agrupadas

- ✓ No exemplo anterior, SQL **particionou** os pedidos em grupos (um grupo para cada vendedor).
- ✓ Para cada grupo, computou-se o valor médio da coluna AMOUNT.



Fonte: The Complete Reference, Groff, Weinberg, Third Edition, McGraw Hill





Queries agrupadas



- ✓ Qual a mínima e máxima quota de vendas atribuídas aos vendedores em cada escritório de vendas?

```
SELECT REP_OFFICE , MIN(QUOTA) , MAX(QUOTA)
FROM SALESREPS
GROUP BY REP_OFFICE ;
```

Resultado da QUERY

Result Grid			 Filter Rows: <input type="text"/>
	rep_office	MIN(quota)	max(quota)
	NULL	NULL	NULL
▶	11	275000.00	300000.00
	12	200000.00	300000.00
	13	350000.00	350000.00
	21	350000.00	350000.00
	22	300000.00	300000.00

Estado - Tabela SALESREPS

Result Grid		 Filter Rows:	<input type="text"/>	Edit:
	EMPL_NUM	NAME	REP_OFFICE	QUOTA
▶	101	Dan Roberts	12	300000.00
	102	Sue Smith	21	350000.00
	103	Paul Cruz	12	275000.00
	104	Bob Smith	12	200000.00
	105	Bill Adams	13	350000.00
	106	Sam Clark	11	275000.00
	107	Nancy Angelli	22	300000.00
	108	Larry Fitch	21	350000.00
	109	Mary Jones	11	300000.00
	110	Tom Snyder	NULL	NULL

Fonte: The Complete Reference, Groff, Weinberg, Third Edition, McGraw Hill



Queries agrupadas

- ✓ Quantos vendedores estão alocados em cada escritório de vendas ?

```
SELECT COUNT(*) , REP_OFFICE
FROM SALESREPS
GROUP BY REP_OFFICE ;
```

Resultado da QUERY

Result Grid		
	count(*)	rep_office
	1	NULL
	2	11
	3	12
	1	13
	2	21
	1	22

Estado - Tabela SALESREPS

Result Grid				
Filter Rows:				
	EMPL_NUM	NAME	REP_OFFICE	QUOTA
▶	101	Dan Roberts	12	300000.00
	102	Sue Smith	21	350000.00
	103	Paul Cruz	12	275000.00
	104	Bob Smith	12	200000.00
	105	Bill Adams	13	350000.00
	106	Sam Clark	11	275000.00
	107	Nancy Angelli	22	300000.00
	108	Larry Fitch	21	350000.00
	109	Mary Jones	11	300000.00
	110	Tom Snyder	NULL	NULL



Queries agrupadas

- ✓ Quantos vendedores estão alocados em cada escritório de vendas ?

```
SELECT REP_OFFICE , COUNT(REP_OFFICE) AS 'TOTAL DE VENDEDORES'
FROM SALESREPS
WHERE REP_OFFICE IS NOT NULL
GROUP BY REP_OFFICE
ORDER BY REP_OFFICE ;
```

Resultado da QUERY

Result Grid		
Filter Rows:		
	REP_OFFICE	TOTAL DE VENDEDORES
▶	11	2
	12	3
	13	1
	21	2
	22	1

Estado - Tabela SALESREPS

Result Grid				
Filter Rows:				
	EMPL_NUM	NAME	REP_OFFICE	QUOTA
▶	101	Dan Roberts	12	300000.00
	102	Sue Smith	21	350000.00
	103	Paul Cruz	12	275000.00
	104	Bob Smith	12	200000.00
	105	Bill Adams	13	350000.00
	106	Sam Clark	11	275000.00
	107	Nancy Angelli	22	300000.00
	108	Larry Fitch	21	350000.00
	109	Mary Jones	11	300000.00
	110	Tom Snyder	NULL	NULL

Fonte: The Complete Reference, Groff, Weinberg, Third Edition, McGraw Hill



Queries agrupadas

- ✓ Quantos vendedores estão alocados em cada escritório de vendas ?

```
SELECT REP_OFFICE , COUNT(REP_OFFICE) AS "TOTAL DE VENDEDORES"
FROM SALESREPS
WHERE REP_OFFICE IS NOT NULL
GROUP BY REP_OFFICE
ORDER BY REP_OFFICE ;
```

Oracle

Resultado da QUERY

Result Grid		
Filter Rows:		
	REP_OFFICE	TOTAL DE VENDEDORES
▶	11	2
	12	3
	13	1
	21	2
	22	1

Estado - Tabela SALESREPS

Result Grid				
Filter Rows:				
	EMPL_NUM	NAME	REP_OFFICE	QUOTA
▶	101	Dan Roberts	12	300000.00
	102	Sue Smith	21	350000.00
	103	Paul Cruz	12	275000.00
	104	Bob Smith	12	200000.00
	105	Bill Adams	13	350000.00
	106	Sam Clark	11	275000.00
	107	Nancy Angelli	22	300000.00
	108	Larry Fitch	21	350000.00
	109	Mary Jones	11	300000.00
	110	Tom Snyder	NULL	NULL

Fonte: The Complete Reference, Groff, Weinberg, Third Edition, McGraw Hill



Queries agrupadas

✓ Quantos diferentes clientes são atendidos por cada vendedor ?

```
SELECT CUST_REP , COUNT(CUST_NUM) AS 'TOTAL DE CLIENTES'
FROM CUSTOMERS
GROUP BY CUST_REP
ORDER BY CUST_REP ;
```

Resultado da QUERY

Result Grid Filter Rows: <input type="text"/>		
	CUST_REP	TOTAL DE CLIENTES
▶	101	3
	102	4
	103	3
	104	1
	105	2
	106	2
	107	1
	108	2
	109	2
	110	1

Estado - Tabela CUSTOMERS

Result Grid Filter Rows: <input type="text"/> Edit:				
	CUST_NUM	COMPANY	CUST_REP	CREDIT_LIM
▶	2101	Jones Mfg.	106	65000.00
	2102	First Corp.	101	65000.00
	2103	Acme Mfg.	105	50000.00
	2105	AAA Investments	101	45000.00
	2106	Fred Lewis Corp.	102	65000.00
	2107	Ace International	110	35000.00
	2108	Holm & Landis	109	55000.00
	2109	Chen Associates	103	25000.00
	2111	JCP Inc.	103	50000.00
	2112	Zetacorp	108	50000.00
	2113	Ian & Schmidt	104	20000.00
	2114	Orion Corp.	102	20000.00
	2115	Smithson Corp.	101	20000.00
	2117	J.P. Sinclair	106	35000.00
	2118	Midwest Systems	108	60000.00
	2119	Solomon Inc.	109	25000.00
	2120	Rico Enterprises	102	50000.00
	2121	QMA Assoc.	103	45000.00
	2122	Three Way Lines	105	30000.00
	2123	Carter & Sons	102	40000.00

Fonte: The Complete Reference, Groff, Weinberg, Third Edition, McGraw Hill



Queries agrupadas

✓ Quantos diferentes clientes são atendidos por cada vendedor ?

```
SELECT CUST_REP , COUNT(CUST_NUM) AS "TOTAL DE CLIENTES"
FROM CUSTOMERS
GROUP BY CUST_REP
ORDER BY CUST_REP ;
```



Resultado da QUERY

Result Grid Filter Rows: <input type="text"/>		
	CUST_REP	TOTAL DE CLIENTES
▶	101	3
	102	4
	103	3
	104	1
	105	2
	106	2
	107	1
	108	2
	109	2
	110	1

Estado - Tabela CUSTOMERS

Result Grid Filter Rows: <input type="text"/> Edit: <input type="button" value="X"/>				
	CUST_NUM	COMPANY	CUST_REP	CREDIT_LIM
▶	2101	Jones Mfg.	106	65000.00
	2102	First Corp.	101	65000.00
	2103	Acme Mfg.	105	50000.00
	2105	AAA Investments	101	45000.00
	2106	Fred Lewis Corp.	102	65000.00
	2107	Ace International	110	35000.00
	2108	Holm & Landis	109	55000.00
	2109	Chen Associates	103	25000.00
	2111	JCP Inc.	103	50000.00
	2112	Zetacorp	108	50000.00
	2113	Ian & Schmidt	104	20000.00
	2114	Orion Corp.	102	20000.00
	2115	Smithson Corp.	101	20000.00
	2117	J.P. Sinclair	106	35000.00
	2118	Midwest Systems	108	60000.00
	2119	Solomon Inc.	109	25000.00
	2120	Rico Enterprises	102	50000.00
	2121	QMA Assoc.	103	45000.00
	2122	Three Way Lines	105	30000.00
	2123	Carter & Sons	102	40000.00

Fonte: The Complete Reference, Groff, Weinberg, Third Edition, McGraw Hill





Queries agrupadas





- ✓ SQL pode agrupar resultados de queries baseando-se nos conteúdos de duas ou mais colunas;
- ✓ Exemplo: Calcule o valor total dos pedidos para cada cliente de cada vendedor.

```
SELECT REP , CUST , SUM(AMOUNT)
FROM ORDERS
GROUP BY REP , CUST ;
```

Resultado da QUERY

Result Grid			 Filter Rows:
	REP	CUST	SUM(AMOUNT)
▶	101	2102	3978.00
	101	2108	150.00
	101	2113	22500.00
	102	2106	4026.00
	102	2114	15000.00
	102	2120	3750.00
	103	2111	2700.00
	105	2103	35582.00
	105	2111	3745.00
	106	2101	1458.00
	106	2117	31500.00
	107	2109	31350.00
	107	2124	3082.00
	108	2112	47925.00
	108	2114	7100.00
	108	2118	3608.00
	109	2108	7105.00
	110	2107	23132.00

Estado - Tabela ORDERS

Result Grid		 Filter Rows:		<input type="text"/>		Edit:   		Export/Import:	
	ORDER_NUM	ORDER_DATE	CUST	REP	MFR	PRODUCT	QTY	AMOUNT	
	112987	2007-12-31	2103	105	ACI	4100Y	11	27500.00	
	112989	2008-01-03	2101	106	FEA	114	6	1458.00	
	112992	2007-11-04	2118	108	ACI	41002	10	760.00	
	112993	2007-01-04	2106	102	REI	2A45C	24	1896.00	
	112997	2008-01-08	2124	107	BIC	41003	1	652.00	
	113003	2008-01-25	2108	109	IMM	779C	3	5625.00	
	113007	2008-01-08	2112	108	IMM	773C	3	2925.00	
	113012	2008-01-11	2111	105	ACI	41003	35	3745.00	
	113013	2008-01-14	2118	108	BIC	41003	1	652.00	
	113024	2008-01-20	2114	108	QSA	XK47	20	7100.00	
	113027	2008-01-22	2103	105	ACI	41002	54	4104.00	
	113034	2008-01-29	2107	110	REI	2A45C	8	632.00	
	113036	2008-01-30	2107	110	ACI	4100Z	9	22500.00	
	113042	2008-02-20	2113	101	REI	2A44R	5	22500.00	
	113045	2008-02-02	2112	108	REI	2A44R	10	45000.00	
	113048	2008-02-10	2120	102	IMM	779C	2	3750.00	
	113049	2008-02-10	2118	108	QSA	XK47	2	776.00	
	113051	2008-02-10	2118	108	QSA	XK47	4	1420.00	
	113055	2008-02-15	2108	101	ACI	4100X	6	150.00	
	113057	2008-02-18	2111	103	ACI	4100X	24	600.00	
	113058	2008-02-23	2108	109	FEA	112	10	1480.00	
	113062	2008-02-24	2124	107	FEA	114	10	2430.00	
	113065	2008-02-27	2106	102	QSA	XK47	6	2130.00	
	113069	2008-03-02	2109	107	IMM	775C	22	31350.00	

Fonte: The Complete Reference, Groff, Weinberg, Third Edition, McGraw Hill






Queries agrupadas





- ✓ Calcule o valor total dos pedidos de cada cliente para cada vendedor, ordenado por cliente e dentro de cada cliente ordenado pelo vendedor.

```
SELECT CUST, REP, SUM(AMOUNT)
FROM ORDERS
GROUP BY CUST, REP
ORDER BY CUST, REP;
```

Resultado da QUERY

Result Grid			 Filter Rows:
	CUST	REP	SUM(AMOUNT)
	2101	106	1458.00
	2102	101	3978.00
	2103	105	35582.00
	2106	102	4026.00
	2107	110	23132.00
	2108	101	150.00
	2108	109	7105.00
	2109	107	31350.00
	2111	103	2700.00
	2111	105	3745.00
	2112	108	47925.00
	2113	101	22500.00
	2114	102	15000.00
	2114	108	7100.00
	2117	106	31500.00
	2118	108	3608.00
	2120	102	3750.00
	2124	107	3082.00

Estado - Tabela ORDERS

Result Grid		 Filter Rows:		<input type="text"/>		Edit:   		Export/Import:	
	ORDER_NUM	ORDER_DATE	CUST	REP	MFR	PRODUCT	QTY	AMOUNT	
	112987	2007-12-31	2103	105	ACI	4100Y	11	27500.00	
	112989	2008-01-03	2101	106	FEA	114	6	1458.00	
	112992	2007-11-04	2118	108	ACI	41002	10	760.00	
	112993	2007-01-04	2106	102	REI	2A45C	24	1896.00	
	112997	2008-01-08	2124	107	BIC	41003	1	652.00	
	113003	2008-01-25	2108	109	IMM	779C	3	5625.00	
	113007	2008-01-08	2112	108	IMM	773C	3	2925.00	
	113012	2008-01-11	2111	105	ACI	41003	35	3745.00	
	113013	2008-01-14	2118	108	BIC	41003	1	652.00	
	113024	2008-01-20	2114	108	QSA	XK47	20	7100.00	
	113027	2008-01-22	2103	105	ACI	41002	54	4104.00	
	113034	2008-01-29	2107	110	REI	2A45C	8	632.00	
	113036	2008-01-30	2107	110	ACI	4100Z	9	22500.00	
	113042	2008-02-20	2113	101	REI	2A44R	5	22500.00	
	113045	2008-02-02	2112	108	REI	2A44R	10	45000.00	
	113048	2008-02-10	2120	102	IMM	779C	2	3750.00	
	113049	2008-02-10	2118	108	QSA	XK47	2	776.00	
	113051	2008-02-10	2118	108	QSA	XK47	4	1420.00	
	113055	2008-02-15	2108	101	ACI	4100X	6	150.00	
	113057	2008-02-18	2111	103	ACI	4100X	24	600.00	
	113058	2008-02-23	2108	109	FEA	112	10	1480.00	
	113062	2008-02-24	2124	107	FEA	114	10	2430.00	
	113065	2008-02-27	2106	102	QSA	XK47	6	2130.00	
	113069	2008-03-02	2109	107	IMM	775C	22	31350.00	

Fonte: The Complete Reference, Groff, Weinberg, Third Edition, McGraw Hill



Cláusula HAVING

- ✓ A cláusula **HAVING** permite seleção e rejeição de agrupamento de tuplas;
- ✓ É seguida por uma condição de pesquisa para agrupamentos.
- ✓ Exemplo: Qual é o valor médio de cada vendedor dos pedidos com total acima de 30.000 ?

```
SELECT REP , AVG(AMOUNT)
FROM ORDERS
GROUP BY REP
HAVING SUM(AMOUNT) > 30000 ;
```

Resultado da QUERY

Result Grid		
	REP	AVG(AMOUNT)
▶	105	7865.400000
	106	16479.000000
	107	11477.333333
	108	8376.142857

Estado - Tabela ORDERS

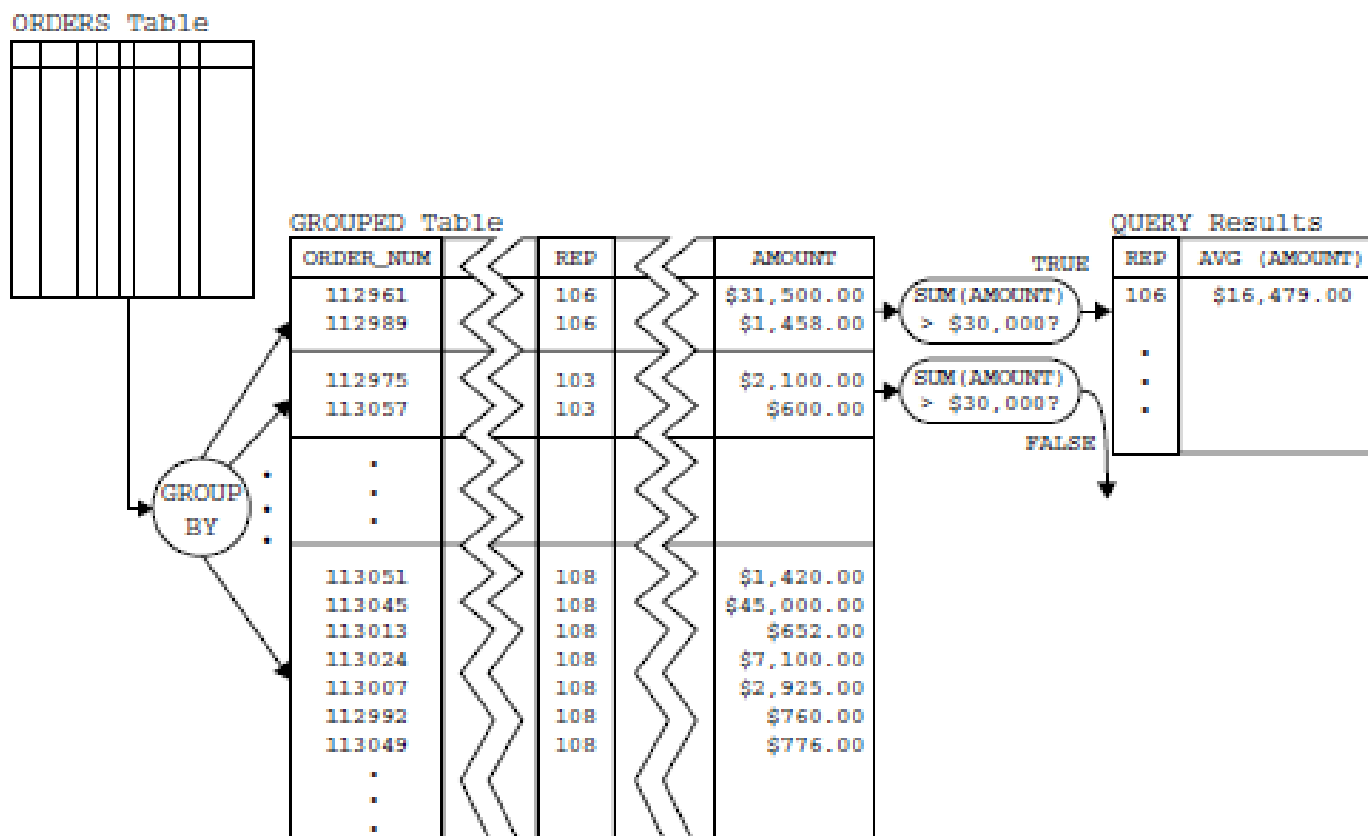
Result Grid								
Filter Rows:								
Edit:								
Export/Import:								
ORDER_NUM	ORDER_DATE	CUST	REP	MFR	PRODUCT	QTY	AMOUNT	
112987	2007-12-31	2103	105	ACI	4100Y	11	27500.00	
112989	2008-01-03	2101	106	FEA	114	6	1458.00	
112992	2007-11-04	2118	108	ACI	41002	10	760.00	
112993	2007-01-04	2106	102	REI	2A45C	24	1896.00	
112997	2008-01-08	2124	107	BIC	41003	1	652.00	
113003	2008-01-25	2108	109	IMM	779C	3	5625.00	
113007	2008-01-08	2112	108	IMM	773C	3	2925.00	
113012	2008-01-11	2111	105	ACI	41003	35	3745.00	
113013	2008-01-14	2118	108	BIC	41003	1	652.00	
113024	2008-01-20	2114	108	QSA	XK47	20	7100.00	
113027	2008-01-22	2103	105	ACI	41002	54	4104.00	
113034	2008-01-29	2107	110	REI	2A45C	8	632.00	
113036	2008-01-30	2107	110	ACI	4100Z	9	22500.00	
113042	2008-02-20	2113	101	REI	2A44R	5	22500.00	
113045	2008-02-02	2112	108	REI	2A44R	10	45000.00	
113048	2008-02-10	2120	102	IMM	779C	2	3750.00	
113049	2008-02-10	2118	108	QSA	XK47	2	776.00	
113051	2008-02-10	2118	108	QSA	XK47	4	1420.00	
113055	2008-02-15	2108	101	ACI	4100X	6	150.00	
113057	2008-02-18	2111	103	ACI	4100X	24	600.00	
113058	2008-02-23	2108	109	FEA	112	10	1480.00	
113062	2008-02-24	2124	107	FEA	114	10	2430.00	
113065	2008-02-27	2106	102	QSA	XK47	6	2130.00	
113069	2008-03-02	2109	107	IMM	775C	22	31350.00	

Fonte: The Complete Reference, Groff, Weinberg, Third Edition, McGraw Hill



Cláusula HAVING

- ✓ Qual é o valor médio de cada vendedor dos pedidos com total acima de 30.000 ?





Subqueries

- ✓ Permitem que se use o resultado de uma query como parte de outra query;
- ✓ Permite que se **estruture** uma query em queries menores (subqueries);
- ✓ Corresponde a uma query dentro de outra query. São mais frequentemente usadas na cláusula **WHERE** de um comando SQL.
- ✓ Exemplo: Listar os vendedores cuja quota é menor que 10% do target de vendas de toda a empresa.



```
SELECT NAME
FROM SALESREPS
WHERE
  QUOTA < ( 0.1 *
    (
      SELECT SUM(TARGET)
      FROM OFFICES
    )
  )
```

```
SELECT SUM(TARGET)
FROM OFFICES
```

Resultado da QUERY

Result Grid	
	NAME
▶	Bob Smith

Estado - Tabela SALESREPS

Result Grid			 Filter Rows: <input type="text"/>	Edit: <input type="text"/>
	EMPL_NUM	NAME	REP_OFFICE	QUOTA
▶	101	Dan Roberts	12	300000.00
	102	Sue Smith	21	350000.00
	103	Paul Cruz	12	275000.00
	104	Bob Smith	12	200000.00
	105	Bill Adams	13	350000.00
	106	Sam Clark	11	275000.00
	107	Nancy Angelli	22	300000.00
	108	Larry Fitch	21	350000.00
	109	Mary Jones	11	300000.00
	110	Tom Snyder	NULL	NULL

Estado - Tabela OFFICES

	OFFICE	CITY	REGION	MGR	TARGET	SALES
▶	11	New York	Eastern	106	575000.00	692637.00
	12	Chicago	Eastern	104	800000.00	735042.00
	13	Atlanta	Eastern	105	350000.00	367911.00
	21	Los Angeles	Western	108	725000.00	835915.00
	22	Denver	Western	108	300000.00	186042.00

Fonte: The Complete Reference, Groff, Weinberg, Third Edition, McGraw Hill



Subqueries

- ✓ Listar os vendedores cujas quotas de vendas são iguais ou maiores que o target do escritório de Atlanta.

```
SELECT NAME
FROM SALESREPS
WHERE
  QUOTA >=
  (
    SELECT TARGET
    FROM OFFICES
    WHERE CITY = 'Atlanta'
  )
```

Resultado da QUERY

Result Grid	
	NAME
▶	Sue Smith
	Bill Adams
	Larry Fitch

Estado - Tabela SALESREPS

Result Grid		Filter Rows:	Edit:
EMPL_NUM	NAME	REP_OFFICE	QUOTA
101	Dan Roberts	12	300000.00
102	Sue Smith	21	350000.00
103	Paul Cruz	12	275000.00
104	Bob Smith	12	200000.00
105	Bill Adams	13	350000.00
106	Sam Clark	11	275000.00
107	Nancy Angelli	22	300000.00
108	Larry Fitch	21	350000.00
109	Mary Jones	11	300000.00
110	Tom Snyder	NULL	NULL

Estado - Tabela OFFICES




OFFICE	CITY	REGION	MGR	TARGET	SALES
11	New York	Eastern	106	575000.00	692637.00
12	Chicago	Eastern	104	800000.00	735042.00
13	Atlanta	Eastern	105	350000.00	367911.00
21	Los Angeles	Western	108	725000.00	835915.00
22	Denver	Western	108	300000.00	186042.00

Fonte: The Complete Reference, Groff, Weinberg, Third Edition, McGraw Hill



Subqueries

✓ Listar todos os clientes atendidos por Bill Adams.

Estado - Tabela SALESREPS

Result Grid		 Filter Rows:	<input type="text"/>	Edit:	
	EMPL_NUM	NAME	REP_OFFICE	QUOTA	
	101	Dan Roberts	12	300000.00	
	102	Sue Smith	21	350000.00	
	103	Paul Cruz	12	275000.00	
	104	Bob Smith	12	200000.00	
	105	Bill Adams	13	350000.00	
	106	Sam Clark	11	275000.00	
	107	Nancy Angelli	22	300000.00	
	108	Larry Fitch	21	350000.00	
	109	Mary Jones	11	300000.00	
	110	Tom Snyder	NULL	NULL	

Estado - Tabela CUSTOMERS

Result Grid			Filter Rows:	<input type="text"/>	Edit: 
	CUST_NUM	COMPANY	CUST_REP	CREDIT_LIM	
▶	2101	Jones Mfg.	106	65000.00	
	2102	First Corp.	101	65000.00	
	2103	Acme Mfg.	105	50000.00	
	2105	AAA Investments	101	45000.00	
	2106	Fred Lewis Corp.	102	65000.00	
	2107	Ace International	110	35000.00	
	2108	Holm & Landis	109	55000.00	
	2109	Chen Associates	103	25000.00	
	2111	JCP Inc.	103	50000.00	
	2112	Zetacorp	108	50000.00	
	2113	Ian & Schmidt	104	20000.00	
	2114	Orion Corp.	102	20000.00	
	2115	Smithson Corp.	101	20000.00	
	2117	J.P. Sindair	106	35000.00	
	2118	Midwest Systems	108	60000.00	
	2119	Solomon Inc.	109	25000.00	
	2120	Rico Enterprises	102	50000.00	
	2121	QMA Assoc.	103	45000.00	
	2122	Three Way Lines	105	30000.00	
	2123	Carter & Sons	102	40000.00	

```
SELECT COMPANY
FROM CUSTOMERS
WHERE CUST_REP =
```

Resultado da QUERY

(

```
SELECT EMPL_NUM
FROM SALESREPS
WHERE NAME = 'Bill Adams'
```

)

Result Grid	Filter Rows:	Edit:
COMPANY		
Acme Mfg.		
Three Way Lines		

Fonte: The Complete Reference, Groff, Weinberg, Third Edition, McGraw Hill

Subqueries

- ✓ Listar todos produtos do fabricante ACI onde a quantidade disponível está acima da quantidade disponível do produto ACI-41004.

```
SELECT DESCRIPTION , QTY_ON_HAND
FROM PRODUCTS
WHERE MFR_ID = 'ACI'
AND QTY_ON_HAND >
(
  SELECT QTY_ON_HAND
  FROM PRODUCTS
  WHERE PRODUCT_ID = '41004'
)
```

Resultado da QUERY

Result Grid		
	DESCRIPTION	QTY_ON_HAND
▶	Size 1 Widget	277
	Size 2 Widget	167
	Size 3 Widget	207

Estado - Tabela PRODUCTS

Result Grid					
	MFR_ID	PRODUCT_ID	DESCRIPTION	PRICE	QTY_ON_HAND
	ACI	41001	Size 1 Widget	55.00	277
	ACI	41002	Size 2 Widget	76.00	167
	ACI	41003	Size 3 Widget	107.00	207
▶	ACI	41004	Size 4 Widget	117.00	139
	ACI	4100X	Widget Adjuster	25.00	37
	ACI	4100Y	Widget Remover	2750.00	25
	ACI	4100Z	Widget Installer	2500.00	28
	BIC	41003	Handle	652.00	3
	BIC	41089	Retainer	225.00	78
	BIC	41627	Plate	180.00	0
	FEA	112	Housing	148.00	115
	FEA	114	Motor Mount	243.00	15
	IMM	773C	300-lb Brace	975.00	28
	IMM	775C	500-lb Brace	1425.00	5
	IMM	779C	900-LB Brace	1875.00	9
	IMM	887H	Brace Holder	54.00	223
	IMM	887P	Brace Pin	250.00	24
	IMM	887X	Brace Retainer	475.00	32
	QSA	XK47	Reducer	355.00	38
	QSA	XK48	Reducer	134.00	203
	QSA	XK48A	Reducer	177.00	37
	REI	2A44G	Hinge Pin	350.00	14
	REI	2A44L	Left Hinge	4500.00	12
	REI	2A44R	Right Hinge	4500.00	12
	REI	2A45C	Ratchet Link	79.00	210

Fonte: The Complete Reference, Groff, Weinberg, Third Edition, McGraw Hill



Subqueries

- ✓ Listar o pessoal de vendas que trabalha em escritórios que estão over target.

```
SELECT NAME
FROM SALESREPS
WHERE REP_OFFICE IN
(
    SELECT OFFICE
    FROM OFFICES
    WHERE SALES > TARGET
)
```

Resultado da QUERY

Result Grid	
	NAME
▶	Sam Clark
	Mary Jones
	Bill Adams
	Sue Smith
	Larry Fitch

Estado - Tabela SALESREPS

Result Grid			Filter Rows:	Edit:
	EMPL_NUM	NAME	REP_OFFICE	QUOTA
▶	101	Dan Roberts	12	300000.00
	102	Sue Smith	21	350000.00
	103	Paul Cruz	12	275000.00
	104	Bob Smith	12	200000.00
	105	Bill Adams	13	350000.00
	106	Sam Clark	11	275000.00
	107	Nancy Angelli	22	300000.00
	108	Larry Fitch	21	350000.00
	109	Mary Jones	11	300000.00
	110	Tom Snyder	NULL	NULL

Estado - Tabela OFFICES

	OFFICE	CITY	REGION	MGR	TARGET	SALES
▶	11	New York	Eastern	106	575000.00	692637.00
	12	Chicago	Eastern	104	800000.00	735042.00
	13	Atlanta	Eastern	105	350000.00	367911.00
	21	Los Angeles	Western	108	725000.00	835915.00
	22	Denver	Western	108	300000.00	186042.00

Obs. A cláusula IN compara um simples valor de dado a uma coluna de valores de dados produzidos por uma subquery e retorna TRUE se houve matching com algum valor da coluna.





Subqueries

- ✓ Listar o pessoal de vendas que não trabalham em escritórios gerenciados pelo empregado (vendedor) 108.

```
SELECT NAME
FROM SALESREPS
WHERE REP_OFFICE NOT IN
(
  SELECT OFFICE
  FROM OFFICES
  WHERE MGR = 108
)
```

Resultado da QUERY

Result Grid	
	NAME
▶	Dan Roberts
	Paul Cruz
	Bob Smith
	Bill Adams
	Sam Clark
	Mary Jones

Estado - Tabela SALESREPS

Result Grid		Filter Rows:	Edit:	
	EMPL_NUM	NAME	REP_OFFICE	QUOTA
▶	101	Dan Roberts	12	300000.00
	102	Sue Smith	21	350000.00
	103	Paul Cruz	12	275000.00
	104	Bob Smith	12	200000.00
	105	Bill Adams	13	350000.00
	106	Sam Clark	11	275000.00
	107	Nancy Angelli	22	300000.00
	108	Larry Fitch	21	350000.00
	109	Mary Jones	11	300000.00
	110	Tom Snyder	NULL	NULL

Estado - Tabela OFFICES

	OFFICE	CITY	REGION	MGR	TARGET	SALES
▶	11	New York	Eastern	106	575000.00	692637.00
	12	Chicago	Eastern	104	800000.00	735042.00
	13	Atlanta	Eastern	105	350000.00	367911.00
	21	Los Angeles	Western	108	725000.00	835915.00
	22	Denver	Western	108	300000.00	186042.00

Fonte: The Complete Reference, Groff, Weinberg, Third Edition, McGraw Hill



Subqueries

- ✓ Listar todos os clientes que tenham feito pedido da empresa ACI, com os produtos iniciando com '4100' entre Janeiro e Junho de 1990.

Estado - Tabela CUSTOMERS

	CUST_NUM	COMPANY	CUST_REP	CREDIT_LIM
▶	2101	Jones Mfg.	106	65000.00
	2102	First Corp.	101	65000.00
	2103	Acme Mfg.	105	50000.00
	2105	AAA Investments	101	45000.00
	2106	Fred Lewis Corp.	102	65000.00
	2107	Ace International	110	35000.00
	2108	Holm & Landis	109	55000.00
	2109	Chen Associates	103	25000.00
	2111	JCP Inc.	103	50000.00
	2112	Zetacorp	108	50000.00
	2113	Ian & Schmidt	104	20000.00
	2114	Orion Corp.	102	20000.00
	2115	Smithson Corp.	101	20000.00
	2117	J.P. Sinclair	106	35000.00
	2118	Midwest Systems	108	60000.00
	2119	Solomon Inc.	109	25000.00
	2120	Rico Enterprises	102	50000.00
	2121	QMA Assoc.	103	45000.00
	2122	Three Way Lines	105	30000.00
	2123	Carter & Sons	102	40000.00

Estado - Tabela ORDERS

	ORDER_NUM	ORDER_DATE	CUST	REP	MFR	PRODUCT	QTY	AMOUNT
	112987	2007-12-31	2103	105	ACI	4100Y	11	27500.00
	112989	2008-01-03	2101	106	FEA	114	6	1458.00
	112992	2007-11-04	2118	108	ACI	4100Z	10	760.00
	112993	2007-01-04	2106	102	REI	2A45C	24	1896.00
	112997	2008-01-08	2124	107	BIC	41003	1	652.00
	113003	2008-01-25	2108	109	IMM	779C	3	5625.00
	113007	2008-01-08	2112	108	IMM	773C	3	2925.00
	113012	2008-01-11	2111	105	ACI	41003	35	3745.00
	113013	2008-01-14	2118	108	BIC	41003	1	652.00
	113024	2008-01-20	2114	108	QSA	XK47	20	7100.00
	113027	2008-01-22	2103	105	ACI	4100Z	54	4104.00
	113034	2008-01-29	2107	110	REI	2A45C	8	632.00
	113036	2008-01-30	2107	110	ACI	4100Z	9	22500.00
	113042	2008-02-20	2113	101	REI	2A44R	5	22500.00
	113045	2008-02-02	2112	108	REI	2A44R	10	45000.00
	113048	2008-02-10	2120	102	IMM	779C	2	3750.00
	113049	2008-02-10	2118	108	QSA	XK47	2	776.00
	113051	2008-02-10	2118	108	QSA	XK47	4	1420.00
	113055	2008-02-15	2108	101	ACI	4100X	6	150.00
	113057	2008-02-18	2111	103	ACI	4100X	24	600.00
	113058	2008-02-23	2108	109	FEA	112	10	1480.00
	113062	2008-02-24	2124	107	FEA	114	10	2430.00
	113065	2008-02-27	2106	102	QSA	XK47	6	2130.00
	113069	2008-03-02	2109	107	IMM	775C	22	31350.00

Fonte: The Complete Reference, Groff, Weinberg, Third Edition, McGraw Hill



Subqueries

- ✓ Listar todos os clientes que tenham feito pedido da empresa ACI, com os produtos iniciando com '4100' entre Janeiro e Junho de 2008.

```
SELECT COMPANY
FROM CUSTOMERS
WHERE CUST_NUM IN
```

```
(
  SELECT DISTINCT CUST
  FROM ORDERS
  WHERE MFR = 'ACI'
    AND PRODUCT LIKE '4100%'
    AND ORDER_DATE
      BETWEEN '2008-01-01'
        AND '2008-06-30'
)
```

Resultado da QUERY

Result Grid	
COMPANY	
▶	Acme Mfg.
	JCP Inc.
	Holm & Landis
	Ace International

Estado - Tabela ORDERS

Result Grid		Filter Rows:		Edit:		Export/Import:	
ORDER_NUM	ORDER_DATE	CUST	REP	MFR	PRODUCT	QTY	AMOUNT
112987	2007-12-31	2103	105	ACI	4100Y	11	27500.00
112989	2008-01-03	2101	106	FEA	114	6	1458.00
112992	2007-11-04	2118	108	ACI	4100Z	10	760.00
112993	2007-01-04	2106	102	REI	2A45C	24	1896.00
112997	2008-01-08	2124	107	BIC	41003	1	652.00
113003	2008-01-25	2108	109	IMM	779C	3	5625.00
113007	2008-01-08	2112	108	IMM	773C	3	2925.00
113012	2008-01-11	2111	105	ACI	41003	35	3745.00
113013	2008-01-14	2118	108	BIC	41003	1	652.00
113024	2008-01-20	2114	108	QSA	XK47	20	7100.00
113027	2008-01-22	2103	105	ACI	4100Z	54	4104.00
113034	2008-01-29	2107	110	REI	2A45C	8	632.00
113036	2008-01-30	2107	110	ACI	4100Z	9	22500.00
113042	2008-02-20	2113	101	REI	2A44R	5	22500.00
113045	2008-02-02	2112	108	REI	2A44R	10	45000.00
113048	2008-02-10	2120	102	IMM	779C	2	3750.00
113049	2008-02-10	2118	108	QSA	XK47	2	776.00
113051	2008-02-10	2118	108	QSA	XK47	4	1420.00
113055	2008-02-15	2108	101	ACI	4100X	6	150.00
113057	2008-02-18	2111	103	ACI	4100X	24	600.00
113058	2008-02-23	2108	109	FEA	112	10	1480.00
113062	2008-02-24	2124	107	FEA	114	10	2430.00
113065	2008-02-27	2106	102	QSA	XK47	6	2130.00
113069	2008-03-02	2109	107	IMM	775C	22	31350.00

Fonte: The Complete Reference, Groff, Weinberg, Third Edition, McGraw Hill



Subqueries


- ✓ Listar os escritórios onde o target de vendas do escritório excede a soma das quotas individuais de cada vendedor.

```
SELECT CITY
FROM OFFICES
WHERE
    TARGET >
    (
        SELECT SUM(QUOTA)
        FROM SALESREPS
        WHERE REP_OFFICE = OFFICE
    )
```

Resultado da QUERY

Result Grid	
	CITY
▶	Chicago
	Los Angeles

Estado - Tabela SALESREPS

Result Grid			Filter Rows: <input type="text"/>	Edit: <input type="text"/>
	EMPL_NUM	NAME	REP_OFFICE	QUOTA
▶	101	Dan Roberts	12	300000.00
	102	Sue Smith	21	350000.00
	103	Paul Cruz	12	275000.00
	104	Bob Smith	12	200000.00
	105	Bill Adams	13	350000.00
	106	Sam Clark	11	275000.00
	107	Nancy Angelli	22	300000.00
	108	Larry Fitch	21	350000.00
	109	Mary Jones	11	300000.00
	110	Tom Snyder	NULL	NULL

Estado - Tabela OFFICES

	OFFICE	CITY	REGION	MGR	TARGET	SALES
▶	11	New York	Eastern	106	575000.00	692637.00
	12	Chicago	Eastern	104	800000.00	735042.00
	13	Atlanta	Eastern	105	350000.00	367911.00
	21	Los Angeles	Western	108	725000.00	835915.00
	22	Denver	Western	108	300000.00	186042.00

Fonte: The Complete Reference, Groff, Weinberg, Third Edition, McGraw Hill

Subqueries

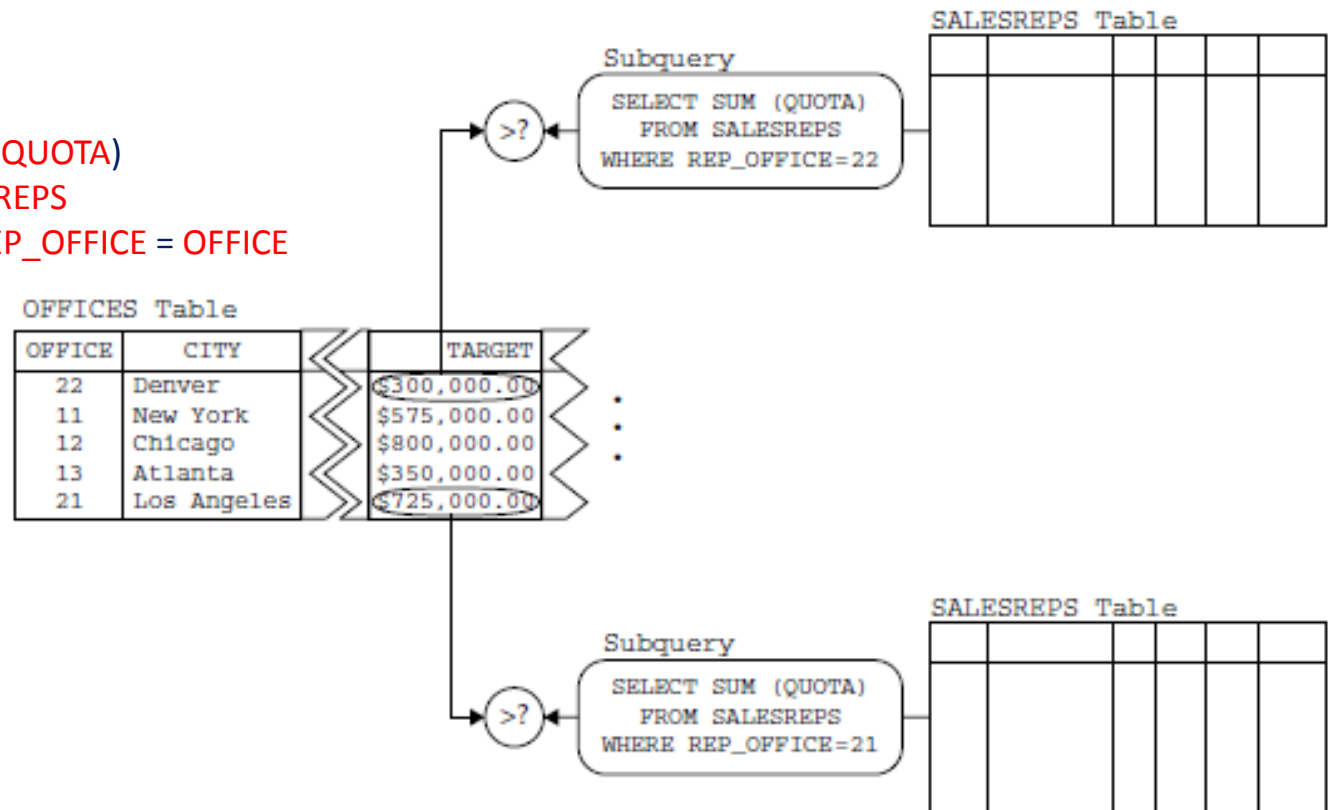
- ✓ Listar os escritórios onde o target de vendas do escritório excede a soma das quotas individuais de cada vendedor.

```
SELECT CITY
FROM OFFICES
WHERE TARGET >
```

```
(
  SELECT SUM(QUOTA)
  FROM SALESREPS
  WHERE REP_OFFICE = OFFICE
)
```

Resultado da QUERY

Result Grid	
	CITY
▶	Chicago
	Los Angeles



Obs. A subquery produz um valor diferente para cada Office, baseando-se nas quotas dos vendedores do respectivo Office. SQL efetua a comparação das linhas da subquery com cada linha da tabela Office.

