Relational Concepts

1. A database “model” describes

   - How does the DBMS physically store the data?

   - How does the DBMS allow programs to do operations against the data?

   - How does the DBMS maintain constraints to protect the data?

The Relational Model

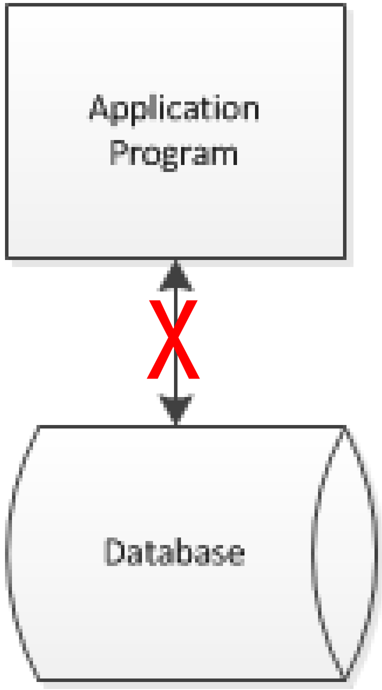
The relational model

- How does the DBMS physically store the data?
  - In tables (relations)

- How does the DBMS allow programs to do operations against the data?
  - Using SQL (an implementation of relational algebra)

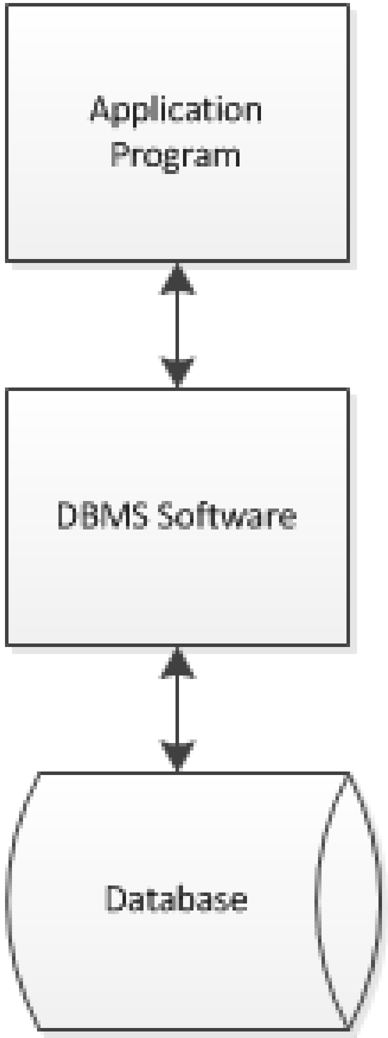
- How does the DBMS maintain constraints to protect the data?
  - Constraint definitions in SQL, OS level files permissions

Constraints



Constraint:

When the DBMS is running, it will not allow any program to directly access any DBMS-managed files (through OS permissions.)



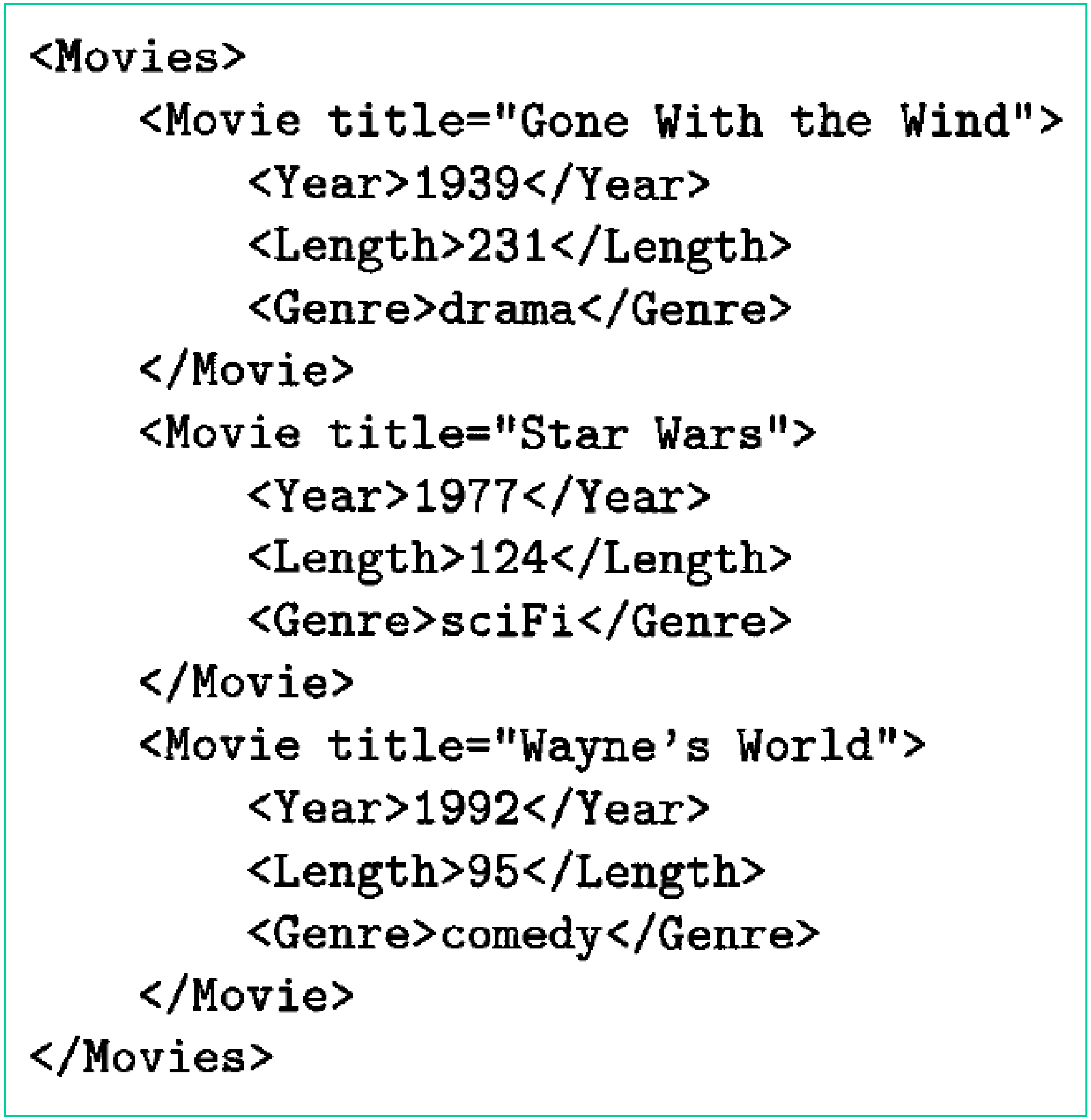
Non-Relational Models

Other Models: semi-structured, unstructured

- Example: XML
- Example: key:value pairs
- Blocks of unstructured text
- Object-Oriented Database

XML

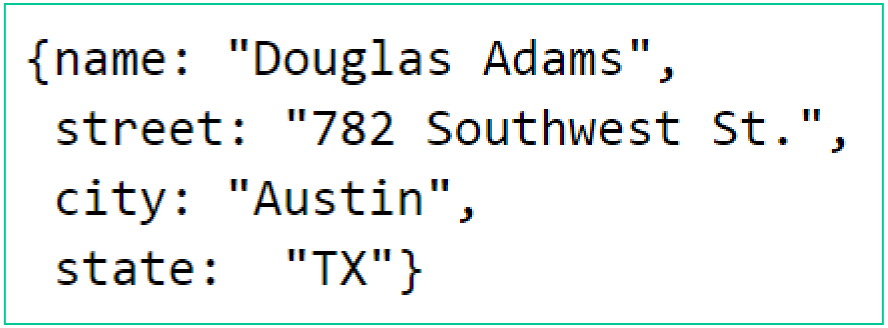
XML – Brief Example



```
<Movies>
    <Movie title="Gone With the Wind">
        <Year>1939</Year>
        <Length>231</Length>
        <Genre>drama</Genre>
    </Movie>
    <Movie title="Star Wars">
        <Year>1977</Year>
        <Length>124</Length>
        <Genre>sciFi</Genre>
    </Movie>
    <Movie title="Wayne's World">
        <Year>1992</Year>
        <Length>95</Length>
        <Genre>comedy</Genre>
    </Movie>
</Movies>
```

Key:Value Pairs

Key-Value Pairs



```
{name: "Douglas Adams",
 street: "782 Southwest St.",
 city: "Austin",
 state:  "TX"}
```

Query = GET, PUT, DELETE

Why the Relational Model?

The Relational Model has prevailed for over 30 years:

- Solid, Reliable Operating Principles
- Can grow very large (multi-terabytes)
- Efficient to manage, maintain
- Available talent in the marketplace
- Fabulous DBMS software alternatives
- Design approach is simple, limited yet versatile
- Data operations (via SQL) are simple, limited yet versatile
- SQL is very powerful: few lines of code ➔ big results

Relational Basics

Basics of the relational model

- We store data about a person, place, thing or event
- Something we store data about is an entity
- Data is stored in a two-dimensional table (relation), rows & columns
- Each row represents one occurrence of an entity
- Each row is unique, has a unique identifier
- We store characteristics, or “attributes” about each occurrence of the entity
- Each column of the table holds one attribute
- A row = a “tuple”

Relational Basics

The “schema”

A description of the relation in this format:

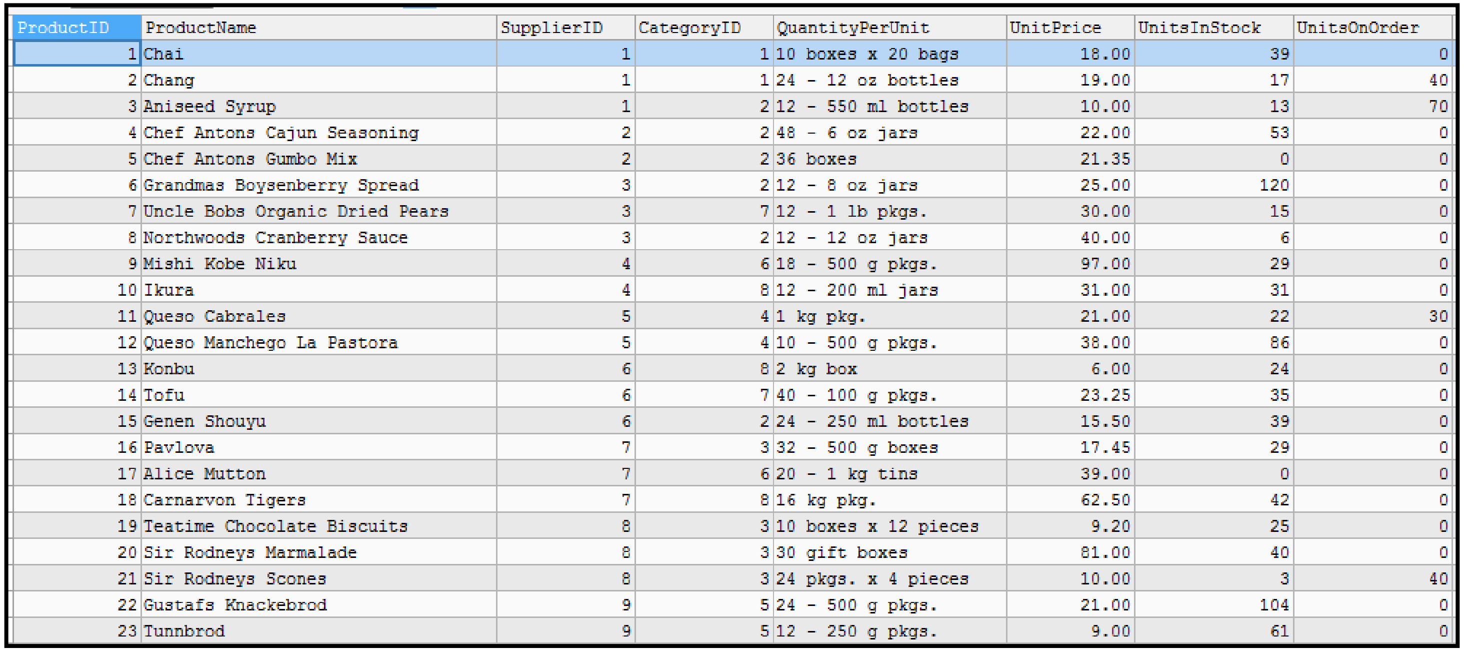
```
RelationName(column-name1, column-name2, column-name3)
```

Or, in this format: (with domain)

```
RelationName(column-name1:datatype,
column-name2:datatype, column-name3:datatype)
```

Relational Basics

Table

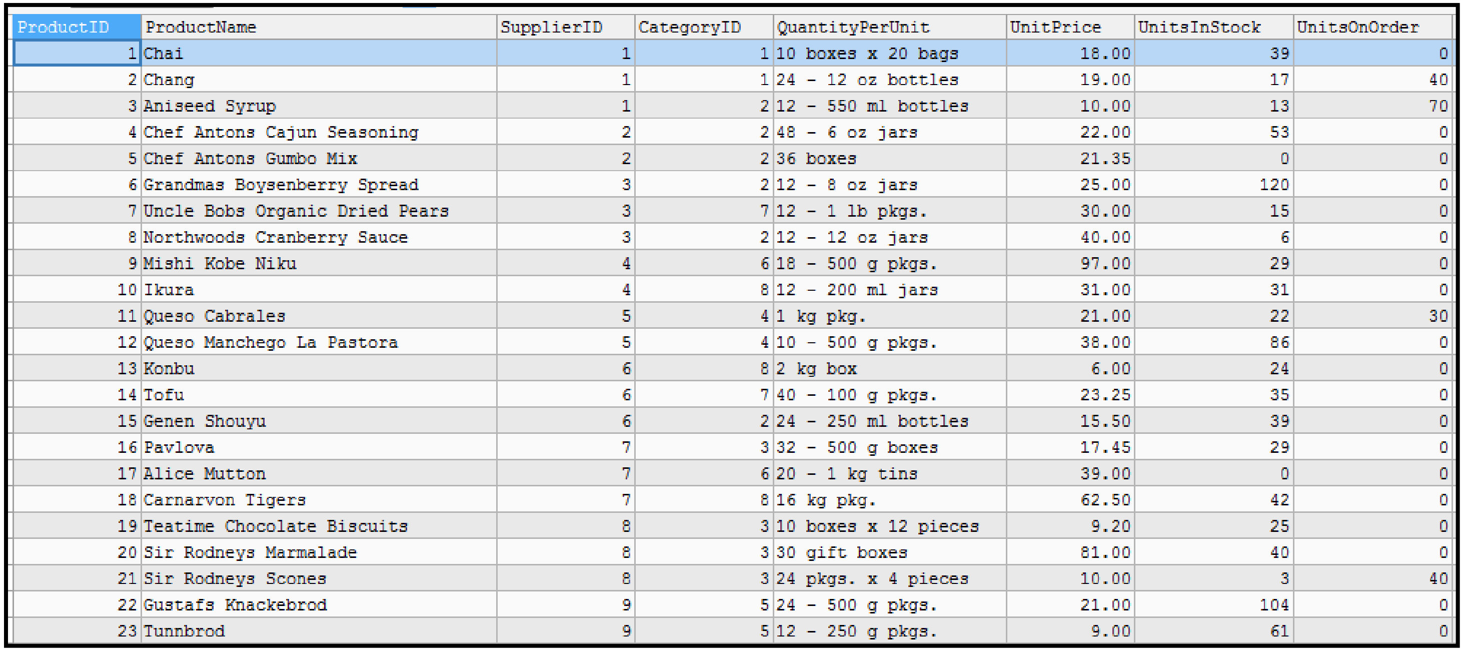


ProductID	ProductName	SupplierID	CategoryID	QuantityPerUnit	UnitPrice	UnitsInStock	UnitsOnOrder
1	Chai	1	1	10 boxes x 20 bags	18.00	39	0
2	Chang	1	1	24 - 12 oz bottles	19.00	17	40
3	Aniseed Syrup	1	2	12 - 550 ml bottles	10.00	13	70
4	Chef Antons Cajun Seasoning	2	2	48 - 6 oz jars	22.00	53	0
5	Chef Antons Gumbo Mix	2	2	36 boxes	21.35	0	0
6	Grandmas Boysenberry Spread	3	2	12 - 8 oz jars	25.00	120	0
7	Uncle Bobs Organic Dried Pears	3	7	12 - 1 lb pkgs.	30.00	15	0
8	Northwoods Cranberry Sauce	3	2	12 - 12 oz jars	40.00	6	0
9	Mishi Kobe Niku	4	6	18 - 500 g pkgs.	97.00	29	0
10	Ikura	4	8	12 - 200 ml jars	31.00	31	0
11	Queso Cabrales	5	4	1 kg pkg.	21.00	22	30
12	Queso Manchego La Pastora	5	4	10 - 500 g pkgs.	38.00	86	0
13	Konbu	6	8	2 kg box	6.00	24	0
14	Tofu	6	7	40 - 100 g pkgs.	23.25	35	0
15	Genen Shouyu	6	2	24 - 250 ml bottles	15.50	39	0
16	Pavlova	7	3	32 - 500 g boxes	17.45	29	0
17	Alice Mutton	7	6	20 - 1 kg tins	39.00	0	0
18	Carnarvon Tigers	7	8	16 kg pkg.	62.50	42	0
19	Teatime Chocolate Biscuits	8	3	10 boxes x 12 pieces	9.20	25	0
20	Sir Rodneys Marmalade	8	3	30 gift boxes	81.00	40	0
21	Sir Rodneys Scones	8	3	24 pkgs. x 4 pieces	10.00	3	40
22	Gustafs Knackebrod	9	5	24 - 500 g pkgs.	21.00	104	0
23	Tunnbrod	9	5	12 - 250 g pkgs.	9.00	61	0

Schema:

```
Product(ProductID, ProductName, SupplierID, CategoryID,
   QuantityPerUnit, UnitPrice, UnitsInStock,UnitsOnOrder)
```

Table



ProductID	ProductName	SupplierID	CategoryID	QuantityPerUnit	UnitPrice	UnitsInStock	UnitsOnOrder
1	Chai	1	1	10 boxes x 20 bags	18.00	39	0
2	Chang	1	1	24 - 12 oz bottles	19.00	17	40
3	Aniseed Syrup	1	2	12 - 550 ml bottles	10.00	13	70
4	Chef Antons Cajun Seasoning	2	2	48 - 6 oz jars	22.00	53	0
5	Chef Antons Gumbo Mix	2	2	36 boxes	21.35	0	0
6	Grandmas Boysenberry Spread	3	2	12 - 8 oz jars	25.00	120	0
7	Uncle Bobs Organic Dried Pears	3	7	12 - 1 lb pkgs.	30.00	15	0
8	Northwoods Cranberry Sauce	3	2	12 - 12 oz jars	40.00	6	0
9	Mishi Kobe Niku	4	6	18 - 500 g pkgs.	97.00	29	0
10	Ikura	4	8	12 - 200 ml jars	31.00	31	0
11	Queso Cabrales	5	4	1 kg pkg.	21.00	22	30
12	Queso Manchego La Pastora	5	4	10 - 500 g pkgs.	38.00	86	0
13	Konbu	6	8	2 kg box	6.00	24	0
14	Tofu	6	7	40 - 100 g pkgs.	23.25	35	0
15	Genen Shouyu	6	2	24 - 250 ml bottles	15.50	39	0
16	Pavlova	7	3	32 - 500 g boxes	17.45	29	0
17	Alice Mutton	7	6	20 - 1 kg tins	39.00	0	0
18	Carnarvon Tigers	7	8	16 kg pkg.	62.50	42	0
19	Teatime Chocolate Biscuits	8	3	10 boxes x 12 pieces	9.20	25	0
20	Sir Rodneys Marmalade	8	3	30 gift boxes	81.00	40	0
21	Sir Rodneys Scones	8	3	24 pkgs. x 4 pieces	10.00	3	40
22	Gustafs Knackebrod	9	5	24 - 500 g pkgs.	21.00	104	0
23	Tunnbrod	9	5	12 - 250 g pkgs.	9.00	61	0

Schema: (with domain)

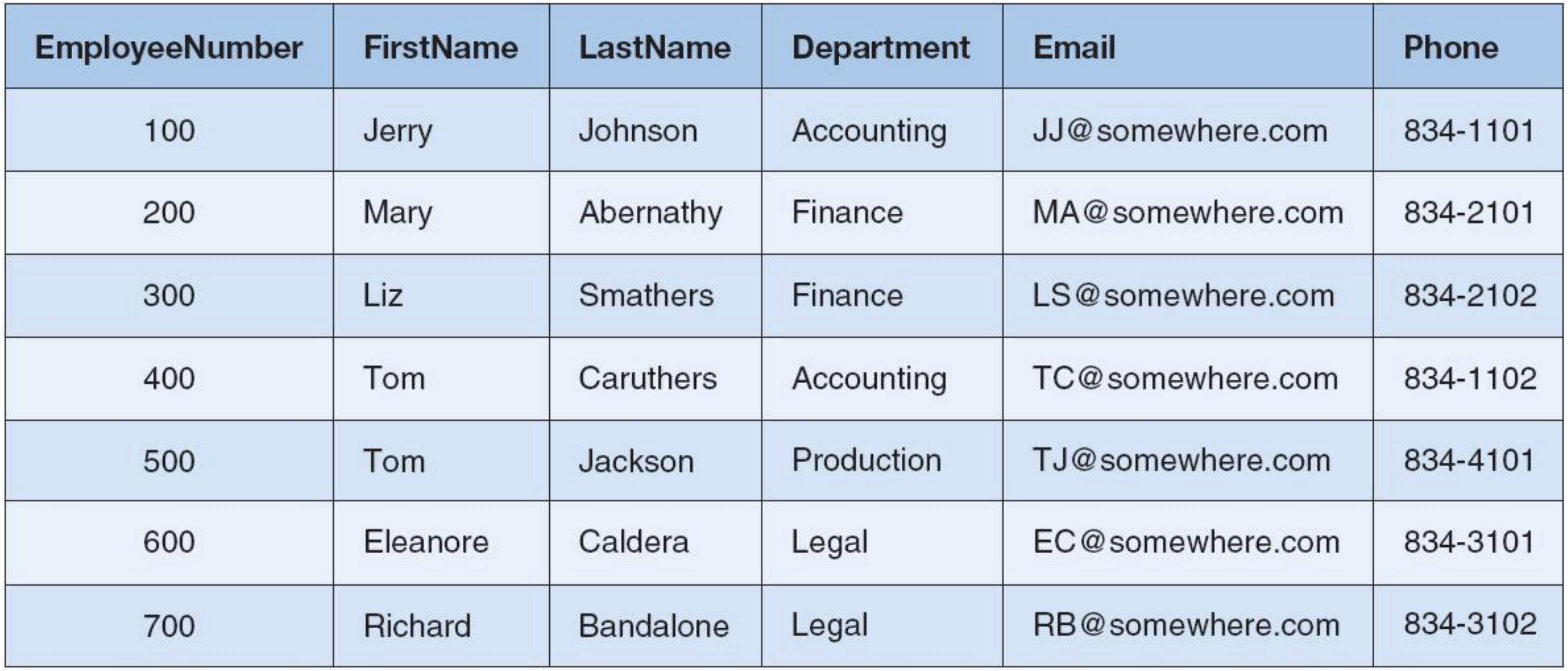
```
Product(ProductID:integer, ProductName:text,
    SupplierID:integer, CategoryID:integer,
    QuantityPerUnit:text, UnitPrice:number,
    UnitsInStock:number,UnitsOnOrder:number)
```

Relational Basics

Relational Rules:

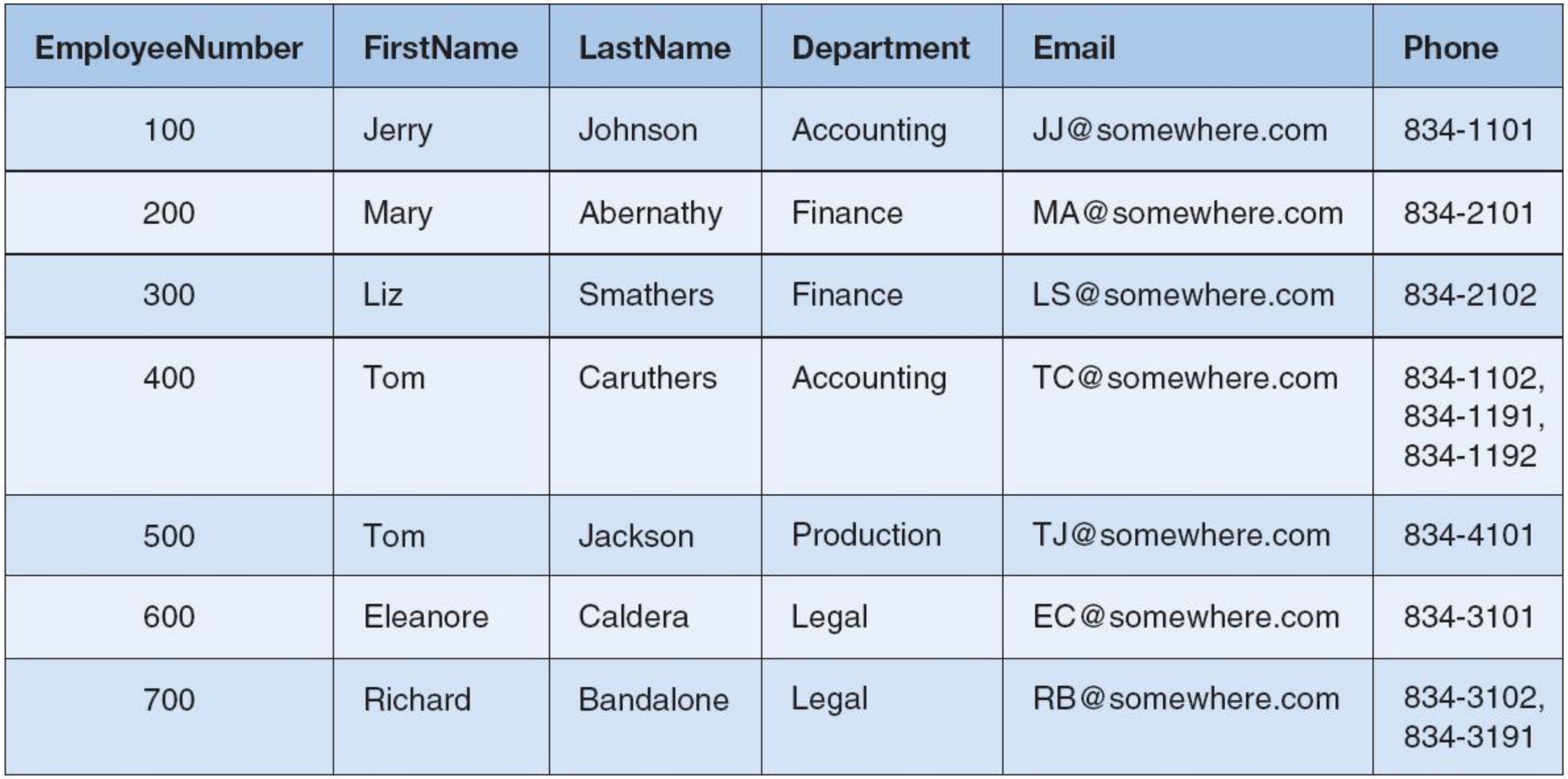
- Each attribute of a tuple must be atomic: that is, one column in one row may contain only a single, atomic value – cannot be broken down further
- Each column of a relation defines a domain: that is, all values in that column are all of the same data type (number, text, date, etc., or null)
- The tuples are a SET, not a list
  - Therefore, their order is immaterial
- The columns are a SET of attributes
  - Therefore, their order is immaterial
- Each tuple must have an attribute that uniquely identifies that tuple. There can be NO duplicate tuples.
  - Called a "Primary Key Constraint"

A Relation



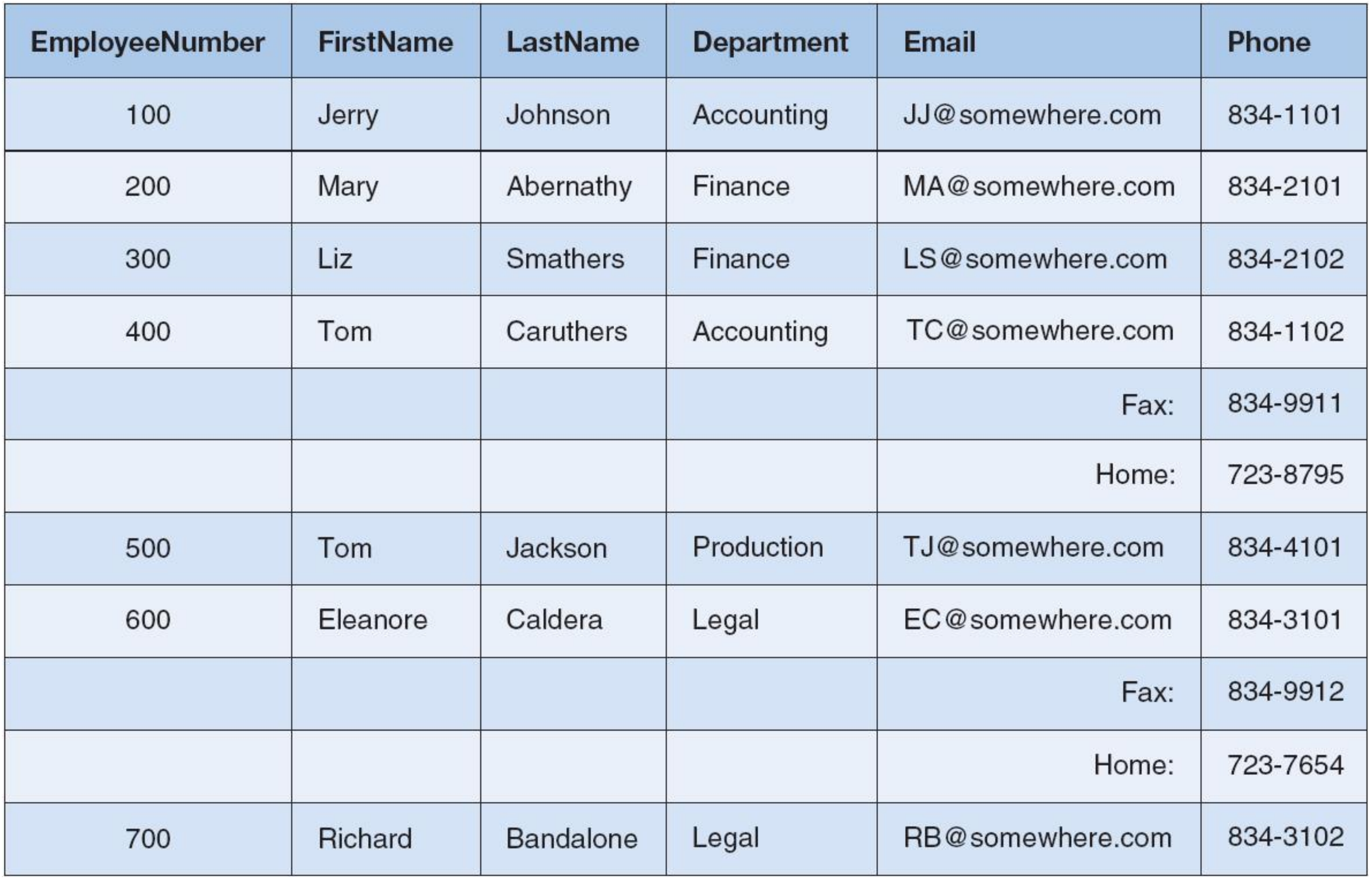
EmployeeNumber	FirstName	LastName	Department	Email	Phone
100	Jerry	Johnson	Accounting	JJ@somewhere.com	834-1101
200	Mary	Abernathy	Finance	MA@somewhere.com	834-2101
300	Liz	Smathers	Finance	LS@somewhere.com	834-2102
400	Tom	Caruthers	Accounting	TC@somewhere.com	834-1102
500	Tom	Jackson	Production	TJ@somewhere.com	834-4101
600	Eleanore	Caldera	Legal	EC@somewhere.com	834-3101
700	Richard	Bandalone	Legal	RB@somewhere.com	834-3102

Tables That Are Not Relations: Multiple Entries per Cell



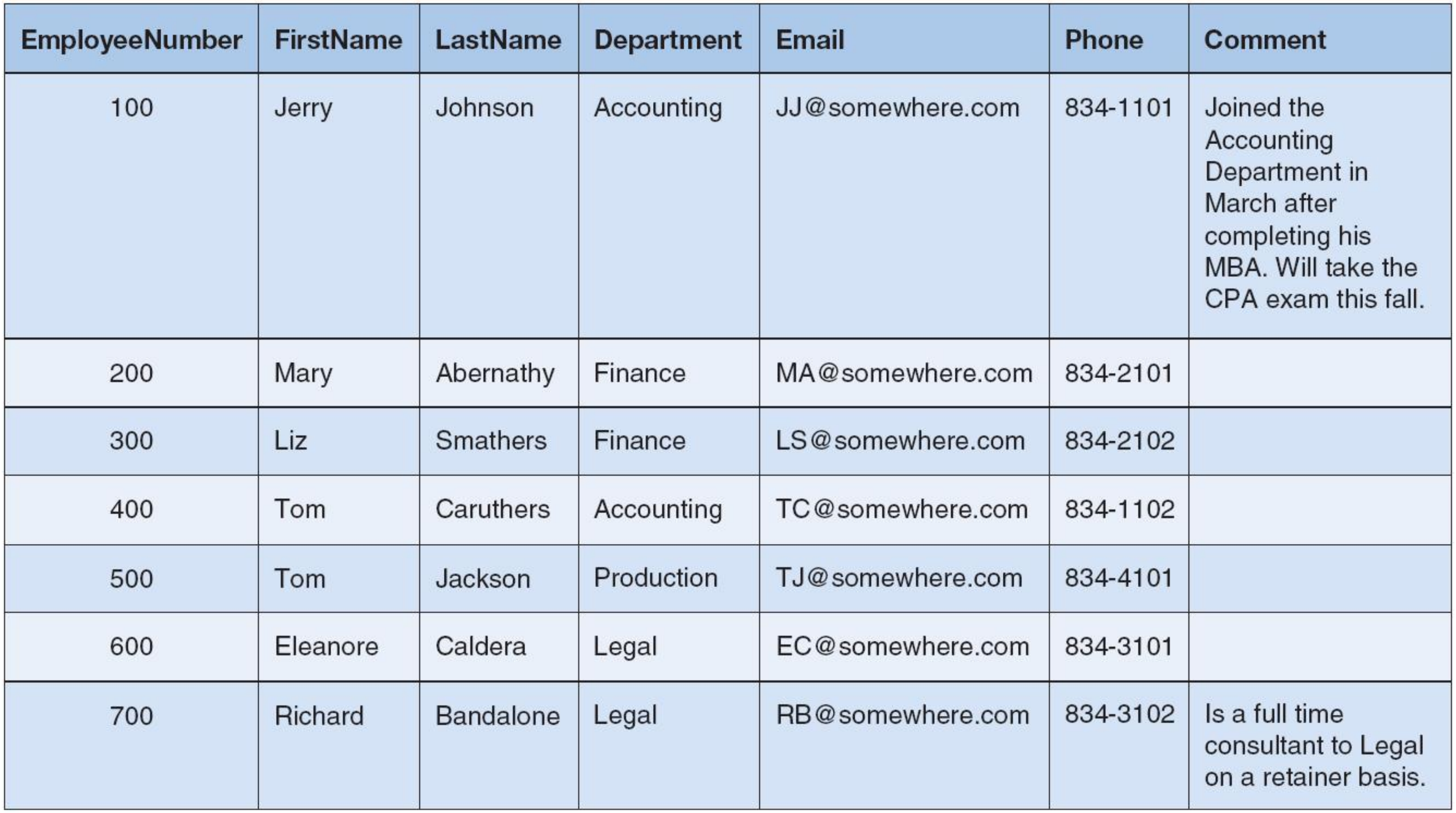
EmployeeNumber	FirstName	LastName	Department	Email	Phone
100	Jerry	Johnson	Accounting	JJ@somewhere.com	834-1101
200	Mary	Abernathy	Finance	MA@somewhere.com	834-2101
300	Liz	Smathers	Finance	LS@somewhere.com	834-2102
400	Tom	Caruthers	Accounting	TC@somewhere.com	834-1102, 834-1191, 834-1192
500	Tom	Jackson	Production	TJ@somewhere.com	834-4101
600	Eleanore	Caldera	Legal	EC@somewhere.com	834-3101
700	Richard	Bandalone	Legal	RB@somewhere.com	834-3102, 834-3191

Tables That Are Not Relations: Table with Required Row Order



EmployeeNumber	FirstName	LastName	Department	Email	Phone
100	Jerry	Johnson	Accounting	JJ@somewhere.com	834-1101
200	Mary	Abernathy	Finance	MA@somewhere.com	834-2101
300	Liz	Smathers	Finance	LS@somewhere.com	834-2102
400	Tom	Caruthers	Accounting	TC@somewhere.com	834-1102
				Fax:	834-9911
				Home:	723-8795
500	Tom	Jackson	Production	TJ@somewhere.com	834-4101
600	Eleanore	Caldera	Legal	EC@somewhere.com	834-3101
				Fax:	834-9912
				Home:	723-7654
700	Richard	Bandalone	Legal	RB@somewhere.com	834-3102

A Relation with Values of Varying Length



EmployeeNumber	FirstName	LastName	Department	Email	Phone	Comment
100	Jerry	Johnson	Accounting	JJ@somewhere.com	834-1101	Joined the Accounting Department in March after completing his MBA. Will take the CPA exam this fall.
200	Mary	Abernathy	Finance	MA@somewhere.com	834-2101	
300	Liz	Smathers	Finance	LS@somewhere.com	834-2102	
400	Tom	Caruthers	Accounting	TC@somewhere.com	834-1102	
500	Tom	Jackson	Production	TJ@somewhere.com	834-4101	
600	Eleanore	Caldera	Legal	EC@somewhere.com	834-3101	
700	Richard	Bandalone	Legal	RB@somewhere.com	834-3102	Is a full time consultant to Legal on a retainer basis.

Terms

We often use terms inter-changeably

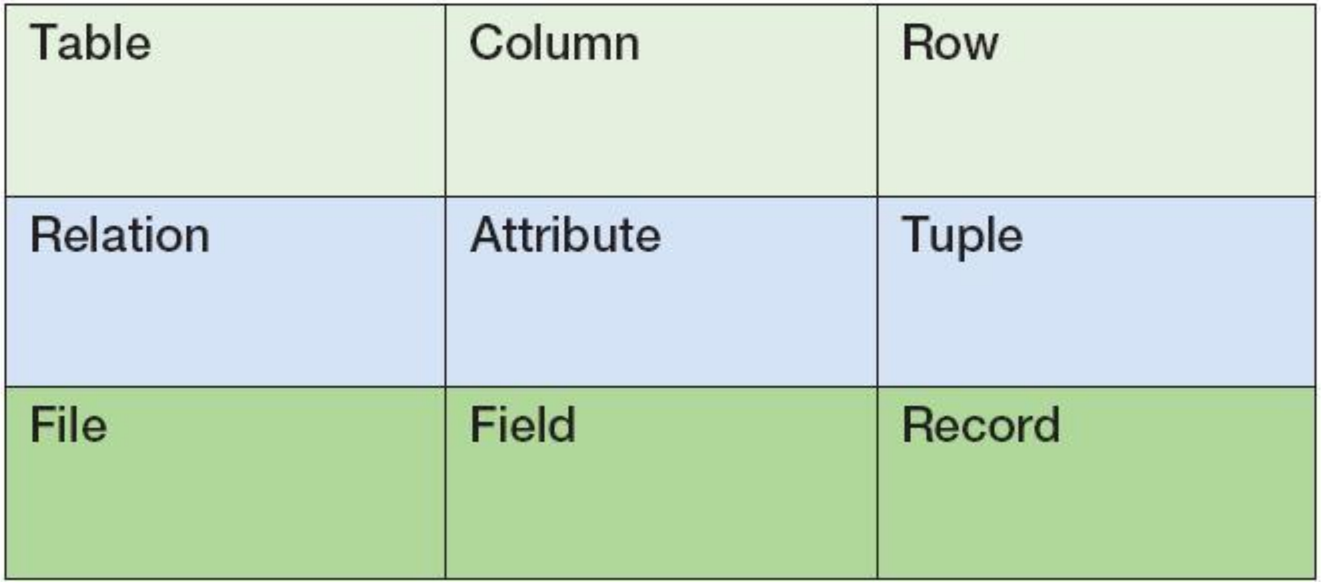


Table	Column	Row
Relation	Attribute	Tuple
File	Field	Record

Keys

- A key is a combination of one or more columns that is used to identify rows in a relation.

- A composite key is a key that consists of two or more columns. Need NOT be contiguous. Also known as concatenated key.

- A candidate key is a key that determines all of the other columns in a relation.

- A primary key is a candidate key selected as the primary means of identifying rows in a relation.
  - There is only one primary key per relation.
  - The primary key may be a composite key.
  - The ideal primary key is short, numeric, and never changes.

Keys

- **Example**

```
Student(StudentID, FirstName, MiddleName, LastName, Phone_1,
Phone_2, Phone_3, StreetAddress, City, State, PostalCode,
SocialSecurityNumber, Major_1, Major_2, Major_3, CumulativeGPA,
CumulativeCreditHours, YearInSchool)
```

What are the candidate keys?

What is the best candidate key?

Keys

- **Example**

```
Student(StudentID, [FirstName, MiddleName, LastName],
[Phone_1], Phone_2, Phone_3, StreetAddress, City, State,
PostalCode, [SocialSecurityNumber], Major_1, Major_2, Major_3,
CumulativeGPA, CumulativeCreditHours, YearInSchool)
```

What are the candidate keys? Name, phone, SSN

What is the best candidate key? StudentID

Business Rules:
What can I tell about the constraints of an organization’s data by examining their data model?

Keys

- **A surrogate key is an artificial column added to a relation to serve as a primary key.**
  - DBMS supplied
  - Short, numeric, and never changes—an ideal primary key
  - Has artificial values that are meaningless to users
  - Normally hidden in forms and reports

Keys

RENTAL_PROPERTY without surrogate key:

RENTAL_PROPERTY (Street, City, State/Province, Zip/PostalCode, Country, Rental_Rate)

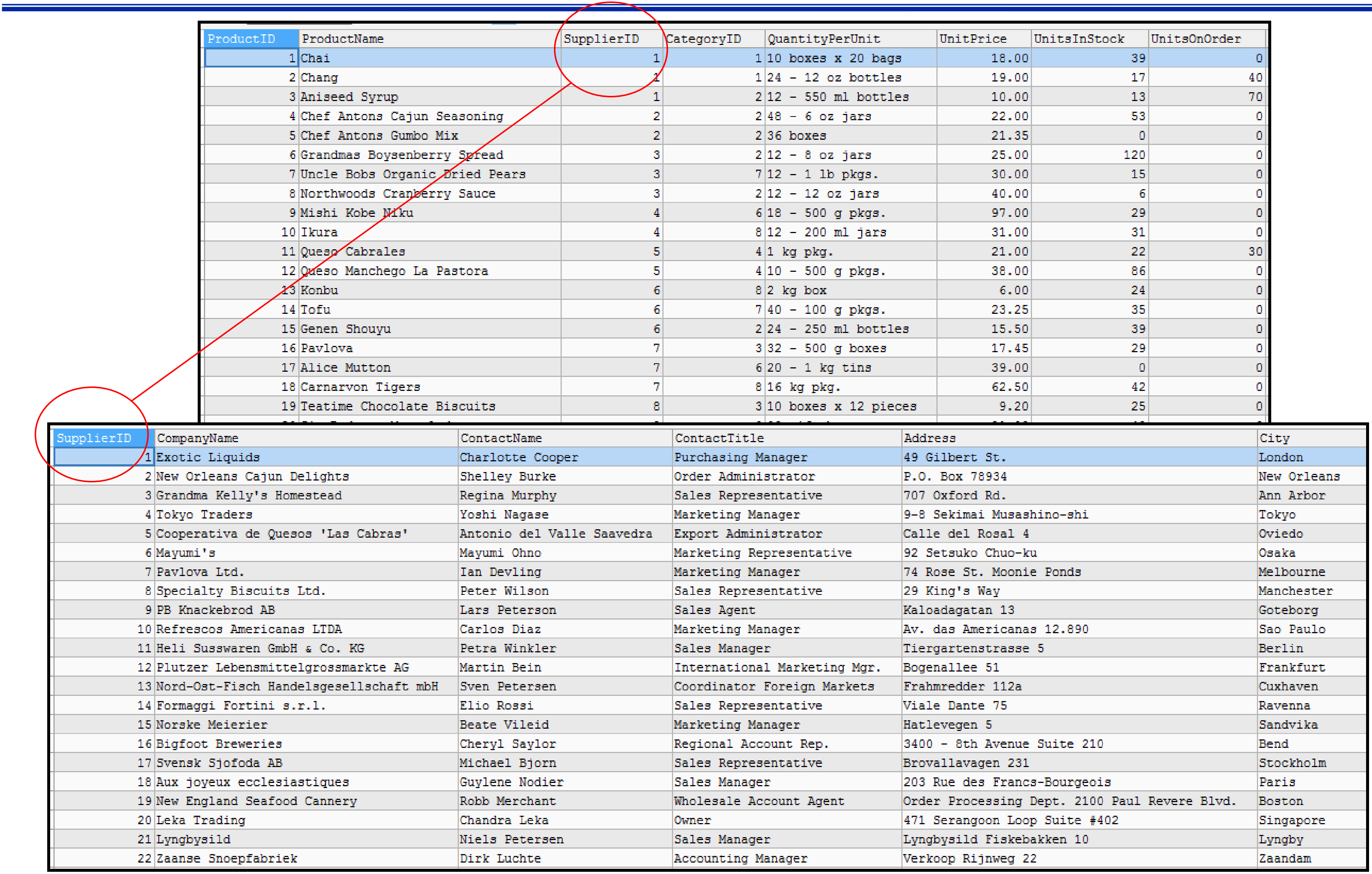
- **RENTAL_PROPERTY with surrogate key:**

RENTAL_PROPERTY (PropertyID, Street, City, State/Province, Zip/PostalCode, Country, Rental_Rate)

Keys

- **A foreign key is the primary key of one relation that is placed into another relation to form a link between the relations.**
  - A foreign key can be a single column or a composite key.

Keys



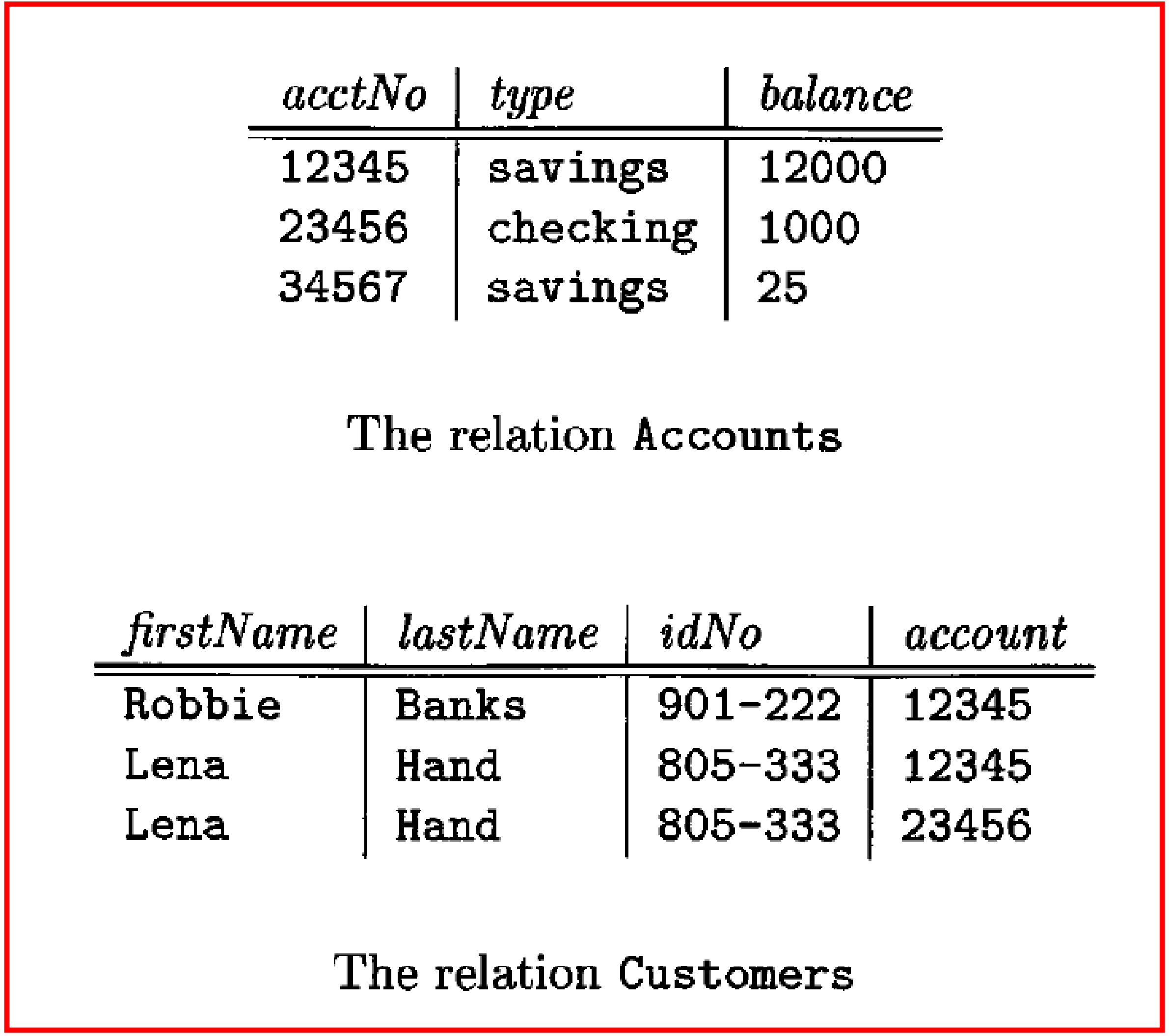
ProductID	ProductName	SupplierID	CategoryID	QuantityPerUnit	UnitPrice	UnitsInStock	UnitsOnOrder
1	Chai	1	1	10 boxes x 20 bags	18.00	39	0
2	Chang	1	1	24 - 12 oz bottles	19.00	17	40
3	Aniseed Syrup	1	2	12 - 550 ml bottles	10.00	13	70
4	Chef Antons Cajun Seasoning	2	2	48 - 6 oz jars	22.00	53	0
5	Chef Antons Gumbo Mix	2	2	36 boxes	21.35	0	0
6	Grandmas Boysenberry Spread	3	2	12 - 8 oz jars	25.00	120	0
7	Uncle Bobs Organic Dried Pears	3	7	12 - 1 lb pkgs.	30.00	15	0
8	Northwoods Cranberry Sauce	3	2	12 - 12 oz jars	40.00	6	0
9	Mishi Kobe Niku	4	6	18 - 500 g pkgs.	97.00	29	0
10	Ikura	4	8	12 - 200 ml jars	31.00	31	0
11	Queso Cabrales	5	4	1 kg pkg.	21.00	22	30
12	Queso Manchego La Pastora	5	4	10 - 500 g pkgs.	38.00	86	0
13	Konbu	6	8	2 kg box	6.00	24	0
14	Tofu	6	7	40 - 100 g pkgs.	23.25	35	0
15	Genen Shouyu	6	2	24 - 250 ml bottles	15.50	39	0
16	Pavlova	7	3	32 - 500 g boxes	17.45	29	0
17	Alice Mutton	7	6	20 - 1 kg tins	39.00	0	0
18	Carnarvon Tigers	7	8	16 kg pkg.	62.50	42	0
19	Teatime Chocolate Biscuits	8	3	10 boxes x 12 pieces	9.20	25	0

SupplierID	CompanyName	ContactName	ContactTitle	Address	City
1	Exotic Liquids	Charlotte Cooper	Purchasing Manager	49 Gilbert St.	London
2	New Orleans Cajun Delights	Shelley Burke	Order Administrator	P.O. Box 78934	New Orleans
3	Grandma Kelly's Homestead	Regina Murphy	Sales Representative	707 Oxford Rd.	Ann Arbor
4	Tokyo Traders	Yoshi Nagase	Marketing Manager	9-8 Sekimai Musashino-shi	Tokyo
5	Cooperativa de Quesos 'Las Cabras'	Antonio del Valle Saavedra	Export Administrator	Calle del Rosal 4	Oviedo
6	Mayumi's	Mayumi Ohno	Marketing Representative	92 Setsuko Chuo-ku	Osaka
7	Pavlova Ltd.	Ian Devling	Marketing Manager	74 Rose St. Moonie Ponds	Melbourne
8	Specialty Biscuits Ltd.	Peter Wilson	Sales Representative	29 King's Way	Manchester
9	PB Knackebrod AB	Lars Peterson	Sales Agent	Kaloadagatan 13	Goteborg
10	Refrescos Americanas LTDA	Carlos Diaz	Marketing Manager	Av. das Americanas 12.890	Sao Paulo
11	Heli Susswaren GmbH & Co. KG	Petra Winkler	Sales Manager	Tiergartenstrasse 5	Berlin
12	Plutzer Lebensmittelgrossmarkte AG	Martin Bein	International Marketing Mgr.	Bogenallee 51	Frankfurt
13	Nord-Ost-Fisch Handelsgesellschaft mbH	Sven Petersen	Coordinator Foreign Markets	Frahmredder 112a	Cuxhaven
14	Formaggi Fortini s.r.l.	Elio Rossi	Sales Representative	Viale Dante 75	Ravenna
15	Norske Meierier	Beate Vileid	Marketing Manager	Hatlevegen 5	Sandvika
16	Bigfoot Breweries	Cheryl Saylor	Regional Account Rep.	3400 - 8th Avenue Suite 210	Bend
17	Svensk Sjofoda AB	Michael Bjorn	Sales Representative	Brovallavagen 231	Stockholm
18	Aux joyeux ecclesiastiques	Guylene Nodier	Sales Manager	203 Rue des Francs-Bourgeois	Paris
19	New England Seafood Cannery	Robb Merchant	Wholesale Account Agent	Order Processing Dept. 2100 Paul Revere Blvd.	Boston
20	Leka Trading	Chandra Leka	Owner	471 Serangoon Loop Suite #402	Singapore
21	Lyngbysild	Niels Petersen	Sales Manager	Lyngbysild Fiskebakken 10	Lyngby
22	Zaanse Snoepfabriek	Dirk Luchte	Accounting Manager	Verkoop Rijnweg 22	Zaandam

Keys

- **A foreign key constraint**
  - Also called “Referential Integrity” or “RI”
  - In the previous example, the DBMS can prevent me from adding a PRODUCT whose SUPPLIER is not found in the database
  - Likewise, the constraint can prevent me from deleting a supplier if that supplier has products in the database

A Little Exercise



acctNo	*type*	*balance*
12345	savings	12000
23456	checking	1000
34567	savings	25

The relation Accounts

firstName	*lastName*	*idNo*	*account*
Robbie	Banks	901-222	12345
Lena	Hand	805-333	12345
Lena	Hand	805-333	23456

The relation Customers

Identify:

1. **Attributes of each relation**
2. **Tuples of each relation**
3. **The schema for each relation**
4. **The database schema**
5. **The domain for each attribute**

A Little Exercise

Identify:

1. **Attributes of each relation**
2. **Tuples of each relation**
3. **The schema for each relation**

   Accounts(acctNo, type, balance)

   Customers(firstName, lastName, idNo, account)
4. **The database schema**

```
Accounts(acctNo:integer,
         type:string,
         balance:number)
Customers(firstName:string,
          lastName:string,
          idNo:string,
          account:integer)
```

5. **The domain for each attribute**

Business rules?

Constraints?