Chapter 3

The Relational Data Model

Outline

- Relational model basics
- Integrity rules
- Rules about referenced rows

Relational Database

- Collection of tables

Owner

OwnerID	OwnerName
1	Fred
2	Amy
3	Esmerelda

Pet

PetID	PetName	Type	Age	OwnerID
1	Freddy	Fish	1	1
2	Butch	Dog	3	1
3	Fifi	Dog	8	2
4	Scraps	Cat		3
5		Kangaroo	36	3

Toy

ToyID	Descr	PetID
100	Toy cat	2
101	Sweater	3
102	Ball	3
103	Catnip	4

Table

- Table = Table Definition + Rows of data
- Table Definition = Table name + Column defs

Student

StdSSN	StdLastName	StdMajor	StdClass	StdGPA
123-45-6789	WELLS	IS	FR	3.00
124-56-7890	NORBERT	FIN	JR	2.70
234-56-7890	KENDALL	ACCT	JR	3.50

SQL: CREATE TABLE Statement

```
CREATE TABLE Student
(    StdSSN           CHAR(11),
     StdFirstName     VARCHAR(50),
     StdLastName      VARCHAR(50),
     StdCity          VARCHAR(50),
     StdState         CHAR(2),
     StdZip           CHAR(10),
     StdMajor         CHAR(6),
     StdClass         CHAR(6),
     StdGPA           DECIMAL(3,2)    )
```

Common Data Types

- CHAR
- VARCHAR
- INTEGER
- DECIMAL
- Date/Time: DATE, TIME, TIMESTAMP

Records

- Each table records information about a collection of entities
- Each row in a table is called a **record**
- A record represents an **entity**
- The columns represent **attributes** of the entities

Student

StdSSN	StdLastName	StdMajor	StdClass	StdGPA
123-45-6789	WELLS	IS	FR	3.00
124-56-7890	NORBERT	FIN	JR	2.70
234-56-7890	KENDALL	ACCT	JR	3.50

Relationships

- Entities in different tables are related by common attribute values

- Example (next slide):

  - Student Wells (in the Student table) is related to two records in the Enrollment table by the common StdSSN attribute values

Relationships

Student

StdSSN	StdLastName
123-45-6789	WELLS
124-56-7890	KENDALL
234-56-7890	NORBERT

Offering

OfferNo	CourseNo
1234	IS320
4321	IS320

Enrollment

StdSSN	OfferNo
123-45-6789	1234
234-56-7890	1234
123-45-6789	4321
124-56-7890	4321

Alternative Terminology

Table-oriented	Set-oriented	Record-oriented	Entity-oriented
Table	Relation	Table	Entity set
Row	Tuple	Record	Entity
Column	Attribute	Field	Attribute

Primary Keys

- Each table has **primary key** column
  - No duplicate values are permitted in the primary key column
  - Primary key value uniquely identifies each entity
  - Primary keys enforce principle of **entity integrity**
- **Entity integrity**
  - Each row ("entity") can be distinguished from other rows in the same table

Course Table Example

```
CREATE TABLE Course
(       CourseNo        CHAR(6) PRIMARY KEY,
        CrsDesc         VARCHAR(250) NOT NULL,
        CrsUnits        INTEGER NOT NULL

)
```

Null Values

- Some record columns contain null values
- Null signifies information that is
  - Unknown
  - Missing
  - Not applicable
- Columns defined as NOT NULL are not permitted to have null values

Foreign Keys

- Some columns reference records in another table
- Example: Enrollment table
  - OfferNo references primary key in Offering
  - StdSSN references primary key in Student
- If the column is defined as a foreign key, the DBMS enforces **referential integrity**
- **Referential integrity –** foreign key columns may contain only values that appear in the related primary key column

Enrollment Table Example

```
CREATE TABLE Enrollment
(    OfferNo    INTEGER,
     StdSSN     CHAR(11),
     EnrGrade   DECIMAL(3,2),
 PRIMARY KEY (OfferNo, StdSSN),
 FOREIGN KEY (OfferNo) REFERENCES Offering,
 FOREIGN KEY (StdSSN)  REFERENCES Student
)
```

Primary Keys II

- A primary key may consist of more than one column
  - Example: Enrollment table
  - Primary key: OfferNo, StdSSN
- Multi-column primary keys may not contain a duplicate combination of values
  - Example: In Enrollment, several records may have the same StdSSN value, but…
  - No two records may have exactly the same combination of OfferNo and StdSSN

Formal Definitions I

- **Superkey**
  - A set of column(s) in a table for which two distinct records are not permitted to have duplicate combined values
  - Example: Student table
    - Combination of StdFirstName, StdLastName, StdSSN is a Superkey
  - By definition, in a table that has a primary key, any set of column(s) in the table that includes the primary key column(s) forms a superkey

Formal Definitions I

- **Candidate key**: minimal superkey
  - A superkey which does not contain any superfluous columns
- **Primary key**: a designated candidate key
  - Selected by database designer as the primary identifier for the table
  - Cannot contain null values
- **Foreign key**: column(s) whose values must match the values in a candidate key of another table

Formal Definitions Recap

- **Superkey:** A set of columns that could serve as a primary key (no duplicate values)
- **Candidate key**: A superkey that contains no unnecessary columns
- **Primary key**: a candidate key chosen to be the identifier for the table
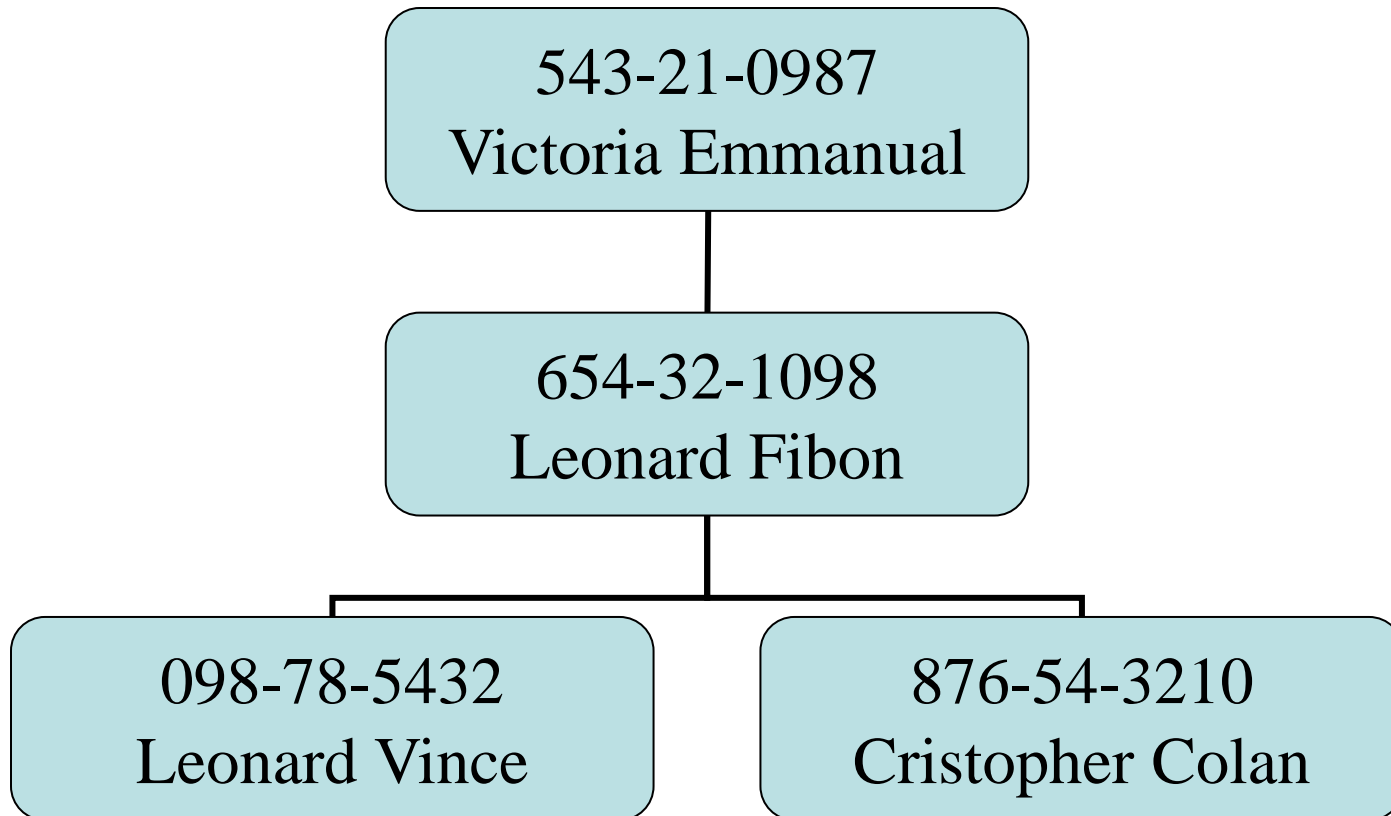
Formal Definitions II

- **Entity integrity**
  - No two rows with the same primary key value
  - No null values in any part of a primary key
- **Referential integrity**
  - Foreign key values must match primary key values in associated table
  - Foreign keys can sometimes be null

Hierarchical Relationships

- Faculty have supervisors
- How to represent this?

Hierarchical Organization

543-21-0987
Victoria Emmanual

654-32-1098
Leonard Fibon

098-78-5432
Leonard Vince

876-54-3210
Cristopher Colan

FacSupervisor Column

FacSSN	FacFirstName	FacLastName	FacRank	FacSalary	FacSupervisor
098-76-5432	LEONARD	VINCE	ASST	$35,000	654-32-1098
543-21-0987	VICTORIA	EMMANUEL	PROF	$120,000	
654-32-1098	LEONARD	FIBON	ASSC	$70,000	543-21-0987
765-43-2109	NICKI	MACON	PROF	$65,000	
876-54-3210	CRISTOPHER	COLAN	ASST	$40,000	654-32-1098
987-65-4321	JULIA	MILLS	ASSC	$75,000	765-43-2109

- FacSupervisor refers to another record in the same table
- Need to ensure that FacSupervisor contains a valid FacSSN
  - Foreign key can reference primary key of same table

Solution: Foreign Key

- A Foreign key can reference its own table's primary key

- Represents relationships among members of the same set

- Not common but important in specialized situations

Summary

- Relational model is commercially dominant

- Learn primary keys, data types, and foreign keys

- Visualize relationships