CS 327E Lecture 2

Shirley Cohen

August 31, 2016
Question 1: Which of the following are reasons for modeling hierarchies?

A. Represents different levels of granularity among related entity classes
B. Protects data integrity through the use of constraints
C. Simplifies joins between diverse entity classes
D. Clarifies the meaning of the entity classes in the ERD
E. All of the above
Question 2: You should create a subtype if groups of attributes are always null or not null together

A. True

B. False
Question 3: Which of the following is a supertype of the entity class Graduate Student?

A. Masters Student
B. PhD Student
C. Student
D. Undergraduate Student
E. None of the above
Question 4: An Administrator is:

A. sometimes a Technician
B. sometimes an Employee
C. always an Employee
D. never an Employee
Question 5: Subtypes cannot have their own subtypes

A. True
B. False
Modeling Hierarchies

Key Concepts:

- Supertypes
- Subtypes

Example 1

Example 2
Product Catalog Hierarchy
## Anti-pattern

### Product Line

<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>101</td>
<td>Apparel</td>
</tr>
<tr>
<td>104</td>
<td>Food Products</td>
</tr>
</tbody>
</table>

### Property Type

<table>
<thead>
<tr>
<th>Prop. Type ID</th>
<th>Prop. Name</th>
<th>Prop. Data Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>P10001</td>
<td>Size</td>
<td>String</td>
</tr>
<tr>
<td>P10005</td>
<td>Garment Type Name</td>
<td>String</td>
</tr>
<tr>
<td>P10012</td>
<td>Calorie Count</td>
<td>Integer</td>
</tr>
</tbody>
</table>

### Property Type Applicability

<table>
<thead>
<tr>
<th>Prod. Line ID</th>
<th>Prop. Type ID</th>
</tr>
</thead>
<tbody>
<tr>
<td>101</td>
<td>P10001</td>
</tr>
<tr>
<td>101</td>
<td>P10005</td>
</tr>
<tr>
<td>104</td>
<td>P10001</td>
</tr>
<tr>
<td>104</td>
<td>P10012</td>
</tr>
</tbody>
</table>

### Product

<table>
<thead>
<tr>
<th>Prod. ID</th>
<th>Prod. Name</th>
<th>Prod. Description</th>
<th>List Price</th>
<th>Prod. Line ID</th>
</tr>
</thead>
<tbody>
<tr>
<td>1254678</td>
<td>ComfoSteer Glove</td>
<td>Men’s leather driving glove</td>
<td>22.99</td>
<td>101</td>
</tr>
<tr>
<td>3549076</td>
<td>CalDry Apricots</td>
<td>Dried California apricots</td>
<td>3.25</td>
<td>104</td>
</tr>
</tbody>
</table>

### Product Property Value

<table>
<thead>
<tr>
<th>Prod. Prop. ID</th>
<th>Prod. ID</th>
<th>Prop. Type ID</th>
<th>Display Order Num.</th>
<th>Prop. Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>33341461</td>
<td>1254678</td>
<td>P10001</td>
<td>2</td>
<td>Large</td>
</tr>
<tr>
<td>86743573</td>
<td>1254678</td>
<td>P10005</td>
<td>1</td>
<td>Men’s gloves</td>
</tr>
<tr>
<td>77303926</td>
<td>3549076</td>
<td>P10001</td>
<td>1</td>
<td>6 oz.</td>
</tr>
<tr>
<td>96901490</td>
<td>3549076</td>
<td>P10012</td>
<td>2</td>
<td>110</td>
</tr>
</tbody>
</table>
Class Enrollment ERD Exercise
Concept Question 1

This is an entity class for storing information on job applicants and their employment history. What constraints have we used on this table that help to protect the integrity of the data?

CREATE TABLE Job_Applicant
(
    candidate_name VARCHAR(50) PRIMARY KEY,
    job_title VARCHAR(50) NOT NULL,
    start_date DATE NOT NULL,
    end_date DATE,
    CHECK (end_date >= start_date)
)

A. primary key on candidate_name
B. not null on job_title and start_date
C. check constraint on date fields
D. all of the above
Concept Question 2

How can we improve the design of the `Job_Applicant` table to be able to track multiple previous employments per applicant?

A. change the unique identifier on the table
B. separate first and last name
C. add a column for each job title
D. none of the above

CREATE TABLE Job_Applicant
(
    candidate_name VARCHAR(50) PRIMARY KEY,
    job_title VARCHAR(50) NOT NULL,
    start_date DATE NOT NULL,
    end_date DATE,
    CHECK (end_date >= start_date)
)
Concept Question 3

This is a table that is intended for storing the room reservations of hotel guests. Can you figure out what’s wrong with the design?

```
CREATE TABLE Hotel_Reservation
(
    guest_name VARCHAR(50) NOT NULL,
    room_nbr INTEGER NOT NULL,
    arrival_date DATE NOT NULL,
    departure_date DATE,
    PRIMARY KEY (room_nbr, arrival_date),
    CHECK (departure_date >= arrival_date)
)
```

A. it doesn’t let you store the contact information for the guest
B. it doesn’t let you store multiple rooms per guest
C. it allows for double-bookings
D. it doesn’t let you check-in and check-out on the same day
E. none of the above
Homework for Next Time

• Read chapter 11 from our Data Modeling textbook