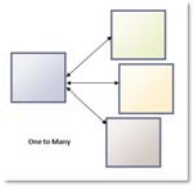
# THE ONE-TO-MANY RELATIONSHIP

Reading: Chapter 4
(pages 89-95)



ISE305 – DATABASE DESIGN & PRACTICE

SOME SLIDE MATERIAL FROM WATSON

## Objectives

- Understand the one-to-many relationship between two entities
- Model a one-to-many relationship using an automated tool
- Define a database with a on-to-many relationship

 3

# DB Design Diagrams

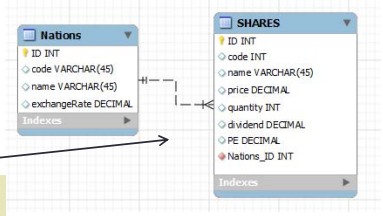
- Once we go beyond single table DBs, diagrams are helpful in design/specification
- Variations in level of diagram
  - Entity Relationship Diagram (ERD)
    - Higher level (entities, not tables)
    - Independent of DB implementation (relational, hierarchical, etc.)
    - Falling out of favor – textbooks favor it, but industry does not
  - Relational Data Model (RDM)
    - Assumes relational implementation
    - Deals with tables and relations
- Variation in notation
  - Dependent on automated tool used

 4

# The One-to-Many Relationship

- Entities are related to other entities
- One-to-many is usually shortened to 1:m
- Example
  - Shares of stock are listed on an exchange in a country
  - One country may have many stocks listed
  - A given stock is listed in only one country

The connector between the two entities shows the relationship

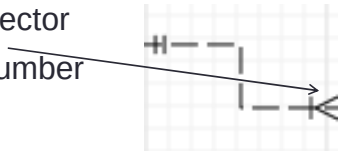



Workbench notation

 5

# Relationship

- Identifies the multiplicity between two entities (e.g., one-to-one, one-to-many, many-to-many)
- Illustrated in the RDM
- We use the crows feet notation for relationships
  - Many side shows a crows foot
  - One side shows a simple connector
  - Vertical lines show minimum number



Helpful if ERDs resemble OO models

Text does not show minimum

 6

# ER Modeling

- Provide a visual representation of data
- Top-down approach to database design
- Identify
  - Entities (implemented as tables)
  - relationships between the entities
- Add details, such as
  - information about the entities and relationships
  - constraints on the entities, relationships, and attributes.

Our emphasis is primarily notation, with emphasis on concrete design (entity modeling is typically done through OO SW design)

 7

# Multiplicity Constraints

- Multiplicity constraints on relationships
  - Number of occurrences of one entity that may relate to a single occurrence of an associated entity

E.g., One nation lists many stocks

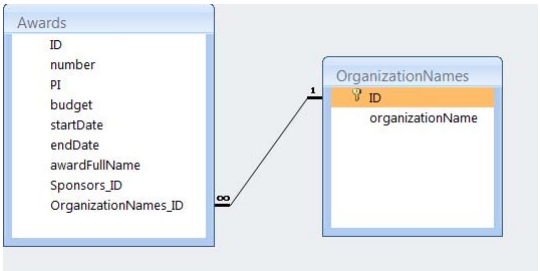
 8

# E-R Notation

- Chen
- IDEF1X
- Bachman
- Crow's Foot
- ISO
- UML

Some data modeling techniques (e.g., Chen) falling into disuse as OO modeling (e.g., UML) dominates

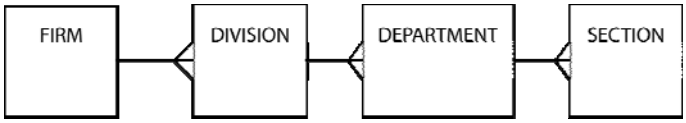
Access diagrams in Design View use a different notation



 9

# Hierarchical Relationships

- Form of 1-m relationship
- Occurs frequently
- Multiple 1:m relationships




 10

# Example – Shares DB

- Techniques to represent the Nations information
  1. Include Nation information in the Shares entity
  2. Provide Nation as a separate entity
- Including Nations information in the Shares entity might cause anomalies

 11

# Single Table Approach

- Add columns for nationName and exchangeRate

NationShares

| ID | code | shareName | price | quantity | dividend | pe | nationName | exchangeRate |
|---|---|---|---|---|---|---|---|---|
| 1 | FC | Freedonia Copper | $27.50 | 10529 | $1.84 | 16 | UK | 1 |
| 2 | PT | Patagonian Tea | $55.25 | 12635 | $2.50 | 10 | UK | 1 |
| 3 | AR | Abyssinian Ruby | $31.82 | 22010 | $1.32 | 13 | UK | 1 |
| 4 | SLG | Sri Lankan Gold | $50.37 | 32868 | $2.68 | 16 | UK | 1 |
| 5 | ILZ | Indian Lead & Zinc | $37.75 | 6390 | $3.00 | 12 | UK | 1 |
| 6 | BE | Burmese Elephant | $0.07 | 154713 | $0.01 | 3 | UK | 1 |
| 7 | BS | Bolivian Sheep | $12.75 | 231678 | $1.78 | 11 | UK | 1 |
| 8 | NG | Nigerian Geese | $35.00 | 12323 | $1.68 | 10 | UK | 1 |
| 9 | CS | Canadian Sugar | $52.78 | 4716 | $2.50 | 15 | UK | 1 |
| 10 | ROF | Royal Ostrich Farms | $33.75 | 1234923 | $3.00 | 6 | UK | 1 |
| 11 | MG | Minnesota Gold | $53.87 | 816122 | $1.00 | 25 | USA | 0.67 |
| 12 | GP | Georgia Peach | $2.35 | 387333 | $0.20 | 5 | USA | 0.67 |
| 13 | NE | Narembeen Emu | $12.34 | 45619 | $1.00 | 8 | Australia | 0.46 |
| 14 | QD | Queensland Diamond | $6.73 | 89251 | $0.50 | 7 | Australia | 0.46 |

 12

# Update Anomalies

- Tables that contain redundant information may potentially suffer from update anomalies
- Types of update anomalies include:
  - Insertion – how do you insert details of a new nation (e.g., exchange rate) that has no stock?
  - Deletion – when we delete the last stock, we lose the information about the nation
  - Modification – changes to a nation must be made for all records containing that nation

We will deal with this issue in more detail later in the semester

 13

# Mapping to a Relational Database

- Each entity becomes a table
- The entity name becomes the table name
- Each attribute becomes a column
- Add a column to the table at the many end of a 1:m relationship
- Put the identifier of the one end in the added column

 14

# Nations Table

- We create a new table using the definition of the Nations entity

NationShares | **Nations**

| ID | code | nationName | exchangeRate |
|---|---|---|---|
| 1 | UK | United Kingdom | 1.00 |
| 2 | USA | United States | 0.67 |
| 3 | AUS | Australia | 0.46 |
| 4 | IND | India | 0.02 |

India exchange rate is displayed with 2 digits of precision, but stored as .0228

 15

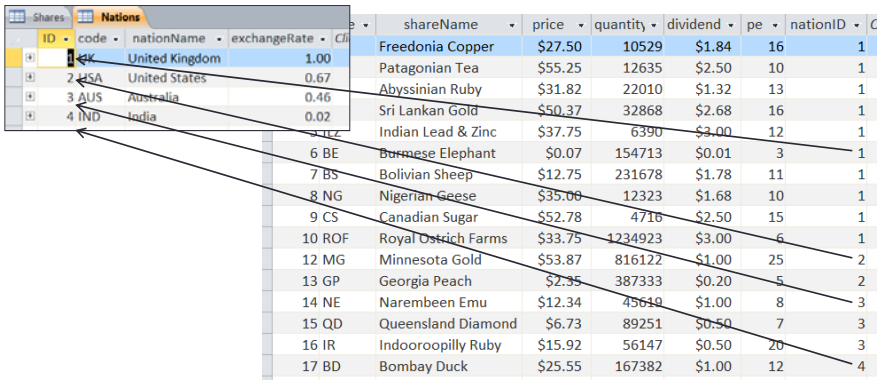
# Foreign Keys

- The 1:m relationship between 2 entities is implemented in a DB with a foreign key
- A foreign key is a column whose values are the primary keys of another table

 16

# Foreign Key

- nationID in the new Shares table is a foreign key that refers to the Nations table



Shares

| shareName | price | quantity | dividend | pe | nationID |
|---|---|---|---|---|---|
| Freedonia Copper | $27.50 | 10529 | $1.84 | 16 | 1 |
| Patagonian Tea | $55.25 | 12635 | $2.50 | 10 | 1 |
| Abyssinian Ruby | $31.82 | 22010 | $1.32 | 13 | 1 |
| Sri Lankan Gold | $50.37 | 32868 | $2.68 | 16 | 1 |
| Indian Lead & Zinc | $37.75 | 6390 | $3.00 | 12 | 1 |
| Burmese Elephant | $0.07 | 154713 | $0.01 | 3 | 1 |
| Bolivian Sheep | $12.75 | 231678 | $1.78 | 11 | 1 |
| Nigerian Geese | $35.00 | 12323 | $1.68 | 10 | 1 |
| Canadian Sugar | $52.78 | 4716 | $2.50 | 15 | 1 |
| Royal Ostrich Farms | $33.75 | 1234923 | $3.00 | 6 | 1 |
| Minnesota Gold | $53.87 | 816122 | $1.00 | 25 | 2 |
| Georgia Peach | $2.35 | 387333 | $0.20 | 5 | 2 |
| Narembeen Emu | $12.34 | 45619 | $1.00 | 8 | 3 |
| Queensland Diamond | $6.73 | 89251 | $0.50 | 7 | 3 |
| Indooroopilly Ruby | $15.92 | 56147 | $0.50 | 20 | 3 |
| Bombay Duck | $25.55 | 167382 | $1.00 | 12 | 4 |

Nations

| ID | code | nationName | exchangeRate |
|---|---|---|---|
| 1 | UK | United Kingdom | 1.00 |
| 2 | USA | United States | 0.67 |
| 3 | AUS | Australia | 0.46 |
| 4 | IND | India | 0.02 |

 17

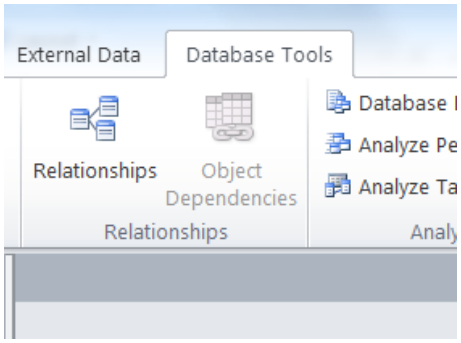
# Referential Integrity Constraint

- For every value of a foreign key there is a primary key with that value
  - E.g., for every value of **`nationID`** in Shares there is a value of **`ID`** in **`Nations)`**
- A foreign key can never be null
- A primary key must exist before the foreign key can be defined
  - Must create Nations before Shares

 18

# Relationships in Access

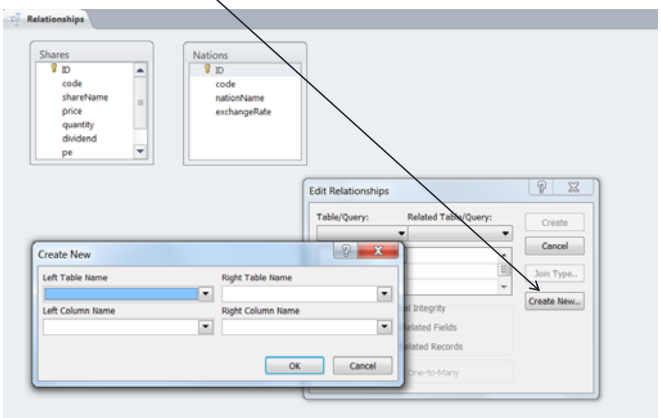
- To create a relationship in Access, first click on Relationships in Database Tools tab



 19

# Relationships in Access

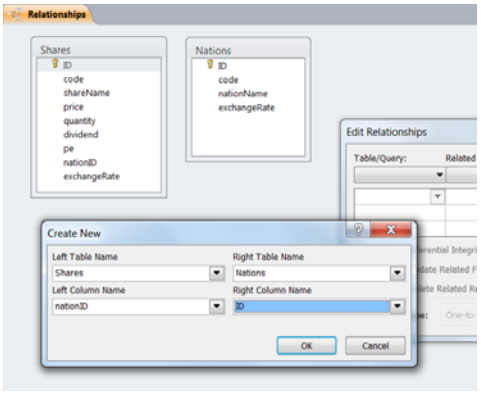
- Then click on Create New … to bring up the Create New dialog box




 20

# Relationships in Access

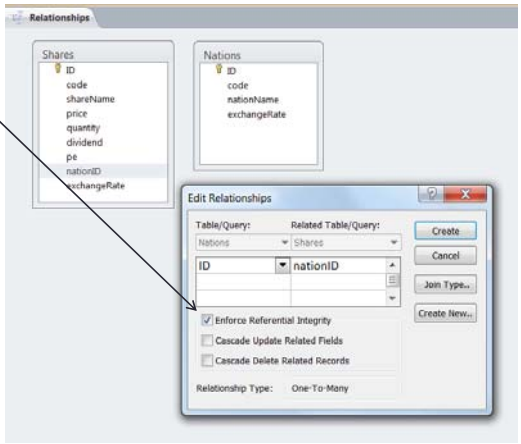
- For every value of a foreign key there is a primary key with that value



 21

# Relationships in Access

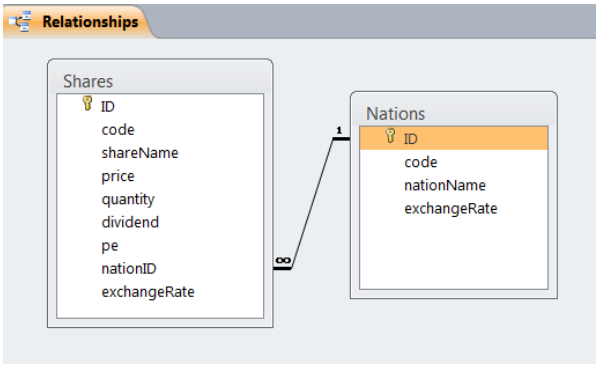
- Enforce referential integrity




 22

# Design View – Multiple Tables

- Access will show the relationships in Design View



 23

# Are We on Track?

- Create a new DB with two tables (Shares and Nations)
  - Create the Nations table with the following data
  - Add a foreign key to the Shares table using data in slide 11

NationShares | **Nations**

| ID | code | nationName | exchangeRate |
|---|---|---|---|
| 1 | UK | United Kingdom | 1.00 |
| 2 | USA | United States | 0.67 |
| 3 | AUS | Australia | 0.46 |
| 4 | IND | India | 0.02 |

 24

# Did You Meet the Objectives?

- Model a one-to-many relationship between two entities
- Model a one-to-many relationship using an automated tool
- Define a database with a on-to-many relationship