Agenda for 1/27/2025

- Answer questions about the course.
- Develop shared vocabulary and define terms related to database.
  - Review answers to week 1 exercise.
  - Do week 2 day 1 exercise.
  - Review answers to week 2 day 1 exercise.

Questions 1 & 2

What is computer technology-based stored data? Provide three examples of stored data.	
What is a database and how does it differ from stored data?	

Question 3

Business Process Application	Data Field	Primitive Data Type of Data Field

Describe a business process application that would use a computer-technology based database	• **A set of computer programs** • **Processes a transaction or a set of transactions to accomplish a business-related task.**

Questions 4 and 5

What is a database management system?	
List four capabilities/common features of a database management system (DBMS)	

Order ID	Order date	Customer Name	Customer Telephone	Customer email	Item ID	Item Description	Item Unit Cost	Item Unit Price	Item Qty Ordered
1234	01/19/2025	Joe's Diner	775-231-1234	joediner@gmail	A43	Tumbler, 12 oz 8 pack	9.20	22.95	16
1234	01/19/2025	Joe's Diner	775-231-1234	joediner@gmail	B97	Salt Shakers – 4 pack	2.25	8.95	10
1234	01/19/2025	Joe's Diner	775-231-1234	joediner@gmail	C71	Crestwood Blender	67.15	85.95	2
1234	01/19/2025	Joe's Diner	775-231-1234	joediner@gmail	A46	Tumbler, 6 oz 8 pack	7.60	21.95	12
3900	01/20/2025	Soup-to-Nuts	775-222-1201	stn@yahoo	C71	Crestwood Blender	67.15	85.95	2
3900	01/20/2025	Soup-to-Nuts	775-222-1201	stn@yahoo	A43	Tumbler, 12 oz 8 pack	9.20	22.95	8
3900	01/20/2025	Soup-to-Nuts	775-222-1201	stn@yahoo	A46	Tumbler, 6 oz 8 pack	7.60	21.95	4
4218	01/15/2025	Red Robin	831-451-8999	redrobin@rr	B97	Salt Shakers – 4 pack	2.25	4.95	1,200
4218	01/15/2025	Red Robin	831-451-8999	redrobin@rr	A43	Tumbler, 12 oz 8 pack	9.20	11.95	1,800
4218	01/15/2025	Red Robin	831-451-8999	redrobin@rr	A42	Tumbler, 16 oz 8 pack	10.15	12.95	1,650
4560	01/17/2025	Soup-to-Nuts	775-222-1201	stn@yahoo	B97	Salt Shakers – 4 pack	2.25	8.95	15

Questions 6, 7, 8, 9

What is the generic term for "CustomerName" or "ItemDescription" on the spreadsheet?	

How does the CustomerName differ from the words "Joe's Diner" in what we generically call them?	

Why are there multiple rows for a single order on the spreadsheet? For example, there are three rows on the spreadsheet for OrderID 3900 and four rows on the spreadsheet for OrderID 1234. Why?	

Keep on referring to that spreadsheet. Does the Item Unit Cost differ for the same item when it is on a different order? (yes or no)	
Refer to the spreadsheet. Does the Item Unit Price differ for the same item when it is on a different order? (yes or no)	

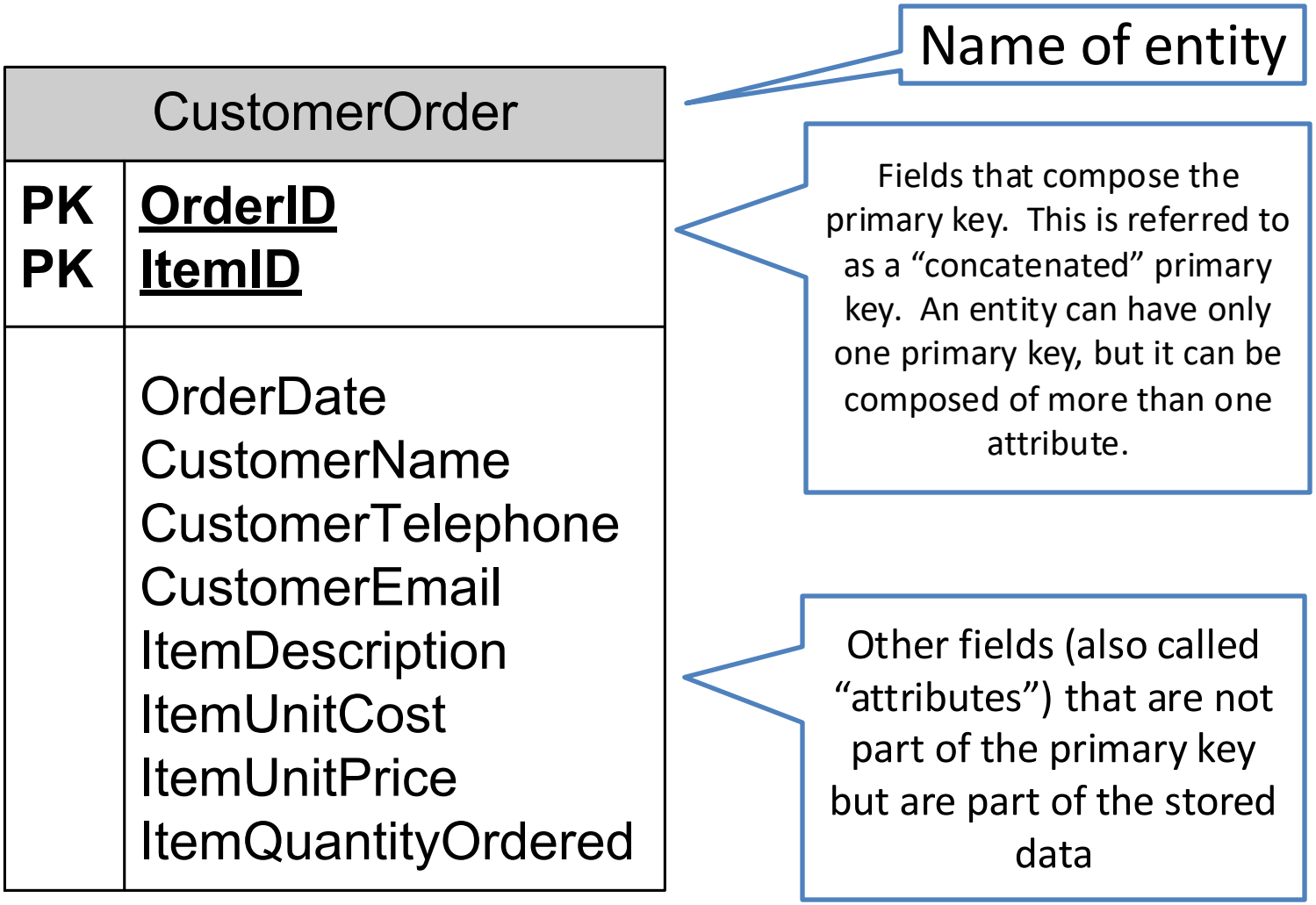
Question 10

I want to keep all the order data as shown in the table on pg. 3. Assume that I've decided to keep all this data in one big table as shown on pg. #3 of the exercise (slide #6). What are the benefits and drawbacks of that decision?

Benefits	Drawbacks

Order ID	Order date	Customer Name	Customer Telephone	Customer email	Item ID	Item Description	Item Unit Cost	Item Unit Price	Item Qty Ordered
1234	01/19/2025	Joe's Diner	775-231-1234	joediner@gmail	A43	Tumbler, 12 oz 8 pack	9.20	22.95	16
1234	01/19/2025	Joe's Diner	775-231-1234	joediner@gmail	B97	Salt Shakers – 4 pack	2.25	8.95	10
1234	01/19/2025	Joe's Diner	775-231-1234	joediner@gmail	C71	Crestwood Blender	67.15	85.95	2
1234	01/19/2025	Joe's Diner	775-231-1234	joediner@gmail	A46	Tumbler, 6 oz 8 pack	7.60	21.95	12
3900	01/20/2025	Soup-to-Nuts	775-222-1201	stn@yahoo	C71	Crestwood Blender	67.15	85.95	2
3900	01/20/2025	Soup-to-Nuts	775-222-1201	stn@yahoo	A43	Tumbler, 12 oz 8 pack	9.20	22.95	8
3900	01/20/2025	Soup-to-Nuts	775-222-1201	stn@yahoo	A46	Tumbler, 6 oz 8 pack	7.60	21.95	4
4218	01/15/2025	Red Robin	831-451-8999	redrobin@rr	B97	Salt Shakers – 4 pack	2.25	4.95	1,200
4218	01/15/2025	Red Robin	831-451-8999	redrobin@rr	A43	Tumbler, 12 oz 8 pack	9.20	11.95	1,800
4218	01/15/2025	Red Robin	831-451-8999	redrobin@rr	A42	Tumbler, 16 oz 8 pack	10.15	12.95	1,650
4560	01/17/2025	Soup-to-Nuts	775-222-1201	stn@yahoo	B97	Salt Shakers – 4 pack	2.25	8.95	15

IS professionals "model" data fields (not values) to simplify and understand its structure. A model of the data in the prior spreadsheet/table format would look like this:



Second Day Exercise

Assume that you decided to NOT store all the data in one big table on the last page of this exercise due to the reasons we discussed in the answer to the first day exercise. The data is repeated on the second page of this exercise.

Option 1: Design a database to store the data using an entity-relationship diagram. Draw it on the back of this page.

Option 2. You may not know how to do option 1 if you did not take IS201 or if you forgot how to do that from IS201. That's okay. Instead of using an entity-relationship diagram, think about creating tables that would store the data non-redundantly, but could be related to each other. There are 10 columns (fields) in the spreadsheet on the last page (OrderID, OrderDate, CustomerName, CustomerTelephone, CustomerEmail, ItemID, ItemDescription, ItemUnitCost, ItemUnitPrice, ItemQuantityOrdered) that need to be included in our database. We are NOT going to store that data all in one table. Some of the columns are text (character) data types which are a big problem when the data is stored redundantly so we need to solve that problem. Make sure that each row that stores a column that has a text data type (example is the ItemDescription) is stored uniquely. For example, we want the actual data value of "Crestwood Blender" to be stored in only one column of one row of one table. How many tables would you need to accomplish this goal? Try to draw those tables on the back of this page, and put data in the tables so that the data will be stored non-redundantly.

I am starting you off with an example of an Item table to store just the Item-related data. Notice that each item occupies only one row in the table. This table takes care of 3 of the required 10 **columns**. You have seven columns to find a home for! Make more tables!

ItemID	ItemDescription	ItemUnitCost
A43	Tumbler, 12 oz 8 pack	9.20
B97	Salt Shakers – 4 pack	2.25
C71	Crestwood Blender	67.15
A46	Tumbler, 6 oz 8 pack	7.60
A42	Tumbler, 16 oz 8 pack	10.15

Beginning list of the rules of a relational database

All data is stored in two-dimensional tables.

Each table must have a column (or set of columns that are concatenated) that serves as a primary key.

A primary key must have a unique data value for each row in the table.

The intersection of a row and column (a cell) can have only one data value.

A table can have (theoretically) an unlimited number of rows.

Relationships on an ERD

- Relationships are between two entities. Never more than two entities.
- Relationships are depicted visually with a line between the two entities.
- A relationship is always read in both directions – they are always bidirectional.
- Relationships are supported technically with a foreign key.
- We are using the crowsfoot notation to depict the cardinality of the relationships.
- There are three possibilities with this notation: zero, 1 or many.

IS475/675 Agenda for Week 2

Learn about

Learn about database management systems (DBMS)

- Know vocabulary
- Identify stored data and database
- Know features of a DBMS

Do

Do database design

- Practice creating data models
- Primarily from "draftsman" perspective – learning how to read the diagrams and create each part of the diagram.

Back to our initial scenario

Your client runs a company that sells restaurant supplies.

Your client sells supplies at a brick-and-mortar store and over the web.

Your client stores all the data on the next slide (the column headers) for each order taken.

Each line on the spreadsheet represents one item on one order. One item on one order is referred to as an "order line."

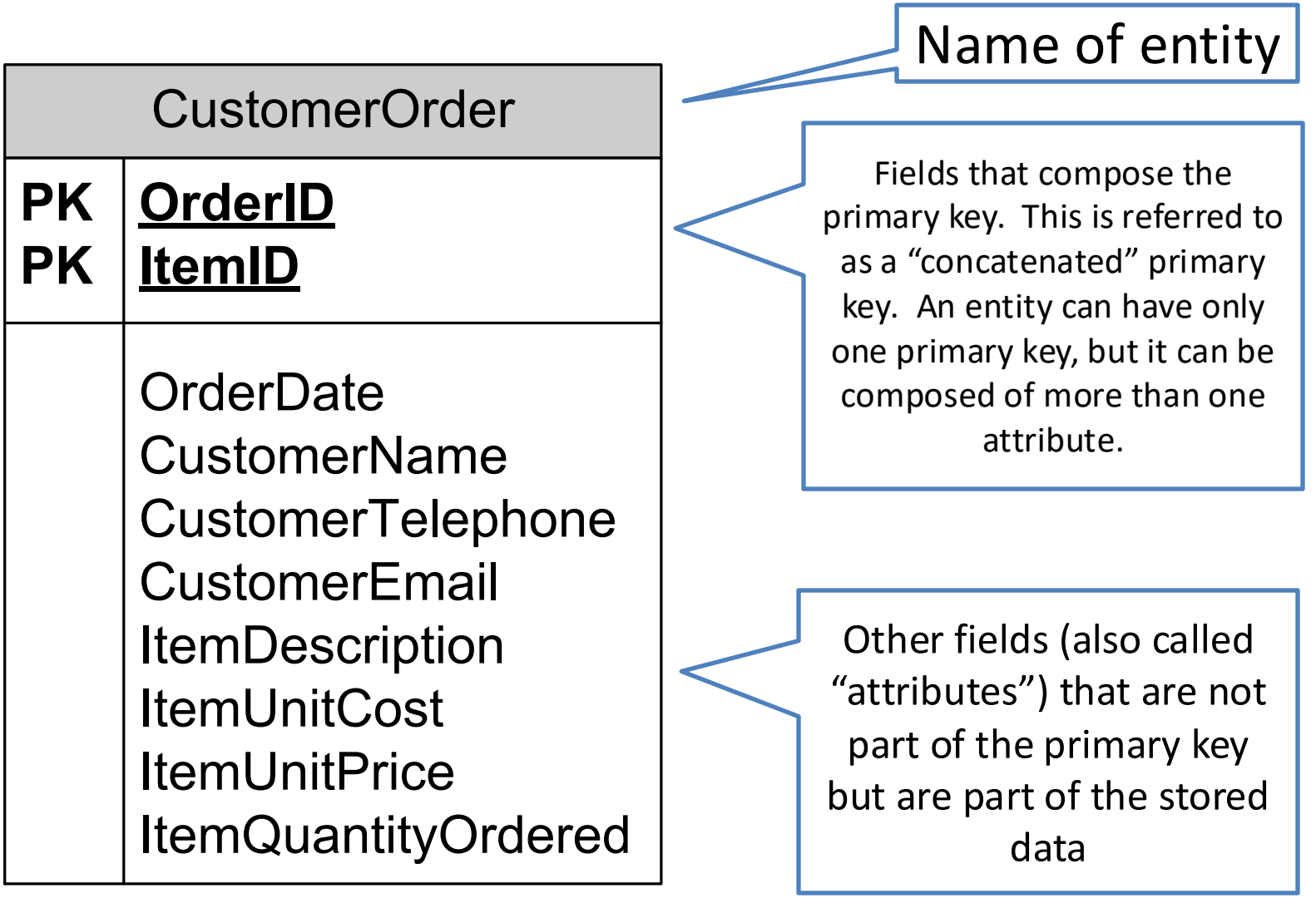
An order can have more than one order line.

An orderID uniquely identifies an order.

The concatenation of an orderID and an ItemID uniquely represents one item on one order.

Order ID	Order date	Customer Name	Customer Telephone	Customer email	Item ID	Item Description	Item Unit Cost	Item Unit Price	Item Qty Ordered
1234	01/19/2025	Joe's Diner	775-231-1234	joediner@gmail	A43	Tumbler, 12 oz 8 pack	9.20	22.95	16
1234	01/19/2025	Joe's Diner	775-231-1234	joediner@gmail	B97	Salt Shakers – 4 pack	2.25	8.95	10
1234	01/19/2025	Joe's Diner	775-231-1234	joediner@gmail	C71	Crestwood Blender	67.15	85.95	2
1234	01/19/2025	Joe's Diner	775-231-1234	joediner@gmail	A46	Tumbler, 6 oz 8 pack	7.60	21.95	12
3900	01/20/2025	Soup-to-Nuts	775-222-1201	stn@yahoo	C71	Crestwood Blender	67.15	85.95	2
3900	01/20/2025	Soup-to-Nuts	775-222-1201	stn@yahoo	A43	Tumbler, 12 oz 8 pack	9.20	22.95	8
3900	01/20/2025	Soup-to-Nuts	775-222-1201	stn@yahoo	A46	Tumbler, 6 oz 8 pack	7.60	21.95	4
4218	01/15/2025	Red Robin	831-451-8999	redrobin@rr	B97	Salt Shakers – 4 pack	2.25	4.95	1,200
4218	01/15/2025	Red Robin	831-451-8999	redrobin@rr	A43	Tumbler, 12 oz 8 pack	9.20	11.95	1,800
4218	01/15/2025	Red Robin	831-451-8999	redrobin@rr	A42	Tumbler, 16 oz 8 pack	10.15	12.95	1,650
4560	01/17/2025	Soup-to-Nuts	775-222-1201	stn@yahoo	B97	Salt Shakers – 4 pack	2.25	8.95	15

IS professionals "model" data fields (not values) to simplify and understand its structure. A model of the data in the prior spreadsheet/table format would look like this:



ItemID	ItemDescription	ItemUnitCost
A43	Tumbler, 12 oz 8 pack	9.20
B97	Salt Shakers – 4 pack	2.25
C71	Crestwood Blender	67.15
A46	Tumbler, 6 oz 8 pack	7.60
A42	Tumbler, 16 oz 8 pack	10.15

Item	
PK	**ItemID**
	ItemDescription ItemUnitCost

Name	Telephone	Email
Joe's Diner	775-231-1234	joediner@gmail
Red Robin	831-451-8999	redrobin@rr
Soup-to-Nuts	775-222-1201	stn@yahoo

Contains only those customers that are unique – no duplications! This is a sample data table.

CustomerID	Name	Telephone	Email
100	Joe's Diner	775-231-1234	joediner@gmail
200	Red Robin	831-451-8999	redrobin@rr
300	Soup-to-Nuts	775-222-1201	stn@yahoo

This is a surrogate (also called artificial) primary key.

Customer	
PK	**CustomerID**
	Name Telephone Email

This is how an IT professional models the entity

Vocabulary Words

- **Entity:** A person, place or thing about which data are to be stored. Also referred to in other contexts as a "file" or a "table".
- **Entity instance:** One recording of data values for attributes in an entity. Also referred to as a "row" or a "record".
- **Attribute:** A detailed characteristic of the data stored within an entity. Also referred to as a "field," "column," or "property".
- **Primary Key (PK):** An attribute or group of attributes that by its/their data value(s) uniquely identifies a "row" (an "instance")in an entity. A primary key that exists in the data is "natural." A primary key that must be created by the designer is called a "surrogate" or "artificial."

Strong entity

A strong entity is an entity that exists whether or not there is a relationship between it and another entity.

For example, in the order application you probably want to store data about a customer, regardless of whether that given customer currently has an order waiting to be filled.

What is another example of a strong entity?

CustomerID	Name	Telephone	Email
100	Joe's Diner	775-231-1234	joediner@gmail
200	Red Robin	831-451-8999	redrobin@rr
300	Soup-to-Nuts	775-222-1201	stn@yahoo

Customer	
PK	**CustomerID**
	Name Telephone Email

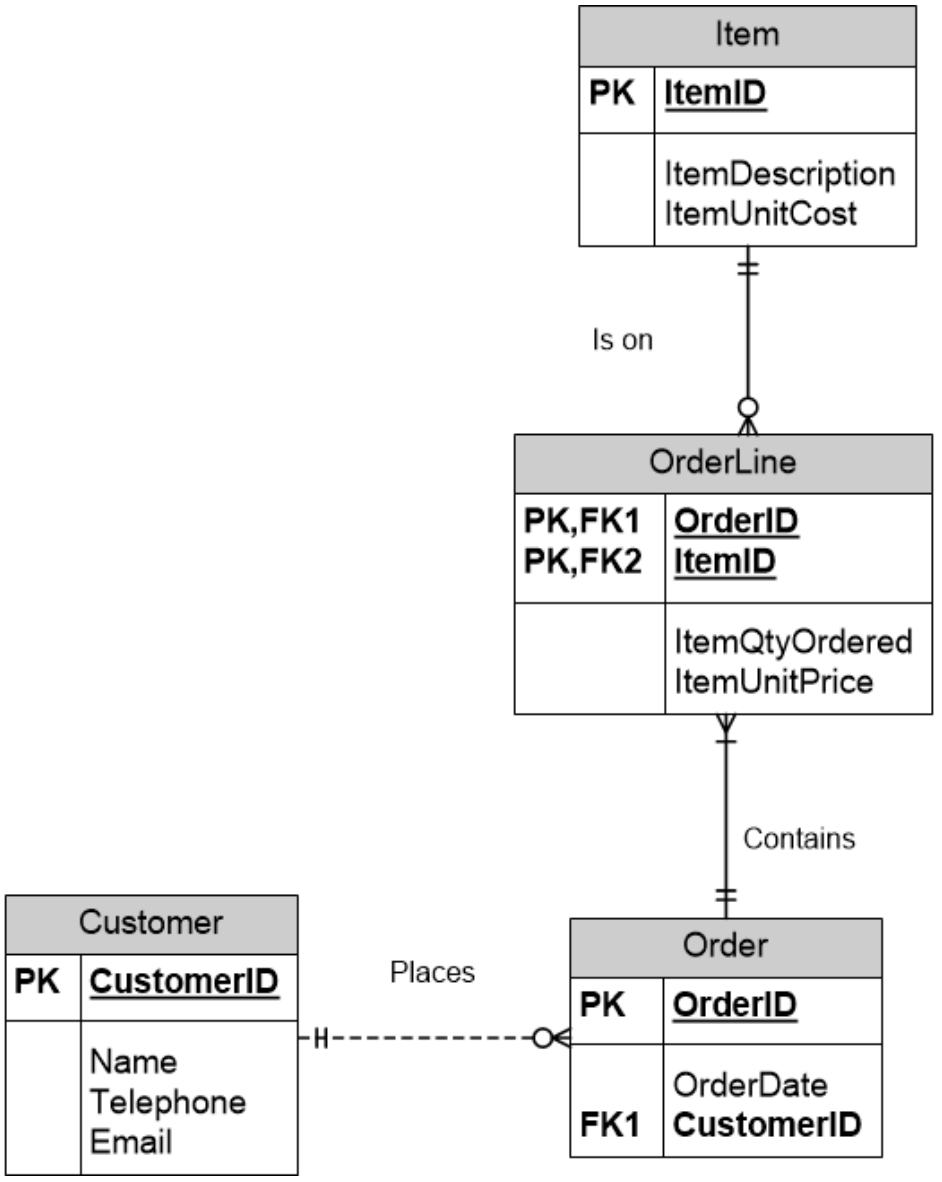
Item	
PK	**ItemID**
	ItemDescription ItemUnitCost

- OrderID
- OrderDate
- ItemUnitPrice
- ItemQuantityOrdered

ItemID	ItemDescription	ItemUnitCost
A43	Tumbler, 12 oz 8 pack	9.20
B97	Salt Shakers – 4 pack	2.25
C71	Crestwood Blender	67.15
A46	Tumbler, 6 oz 8 pack	7.60
A42	Tumbler, 16 oz 8 pack	10.15

What is an entity-relationship diagram (ERD)?

- A model of a database design.
- Depicts each entity and the attributes that will be stored in each entity.
- Shows which attribute(s) compose the primary key.
- Shows the **relationships between entities**. A database is a set of interrelated entities so an ERD must depict the relationships.
- Shows the foreign keys that support the relationships between entities.



Customer	
PK	**CustomerID**
	Name Telephone Email

Order	
PK	**OrderID**
	OrderDate ItemUnitPrice ItemQtyOrdered

Item	
PK	**ItemID**
	ItemDescription ItemUnitCost

Weak Entity

- An entity that relies on the existence of another entity.
  - Example is Order in our sample application.
  - A weak entity is usually a transaction in an application.

Relationships on an ERD

- Relationships are between two entities. Never more than two entities.
- Relationships are depicted visually with a line between the two entities.
- A relationship is always read in both directions – they are always bidirectional.
- Relationships are supported technically with a foreign key.
- We are using the crowsfoot notation to depict the cardinality of the relationships.
- There are three possibilities with this notation: zero, 1 or many.

A bit more about relationships

Entities do not usually exist in isolation.

A connecting line between two entities on an ERD represents a relationship.

A relationship is a natural business association existing between two or more entities.

A relationship supports and sometimes creates a business rule.

A verb phrase describes the relationship.

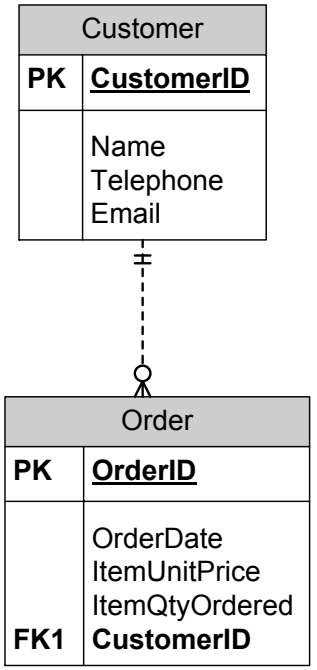
Cardinality of a relationship

"Cardinality" is a constraint on the number of entity instances that participate in a relationship.

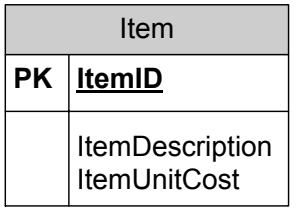
Cardinality describes the minimum and maximum number of instances that one entity has with another entity in a relationship.

Cardinality also describes whether the relationship is mandatory or optional.

We depict cardinality in this class using the Crowsfoot Notation.



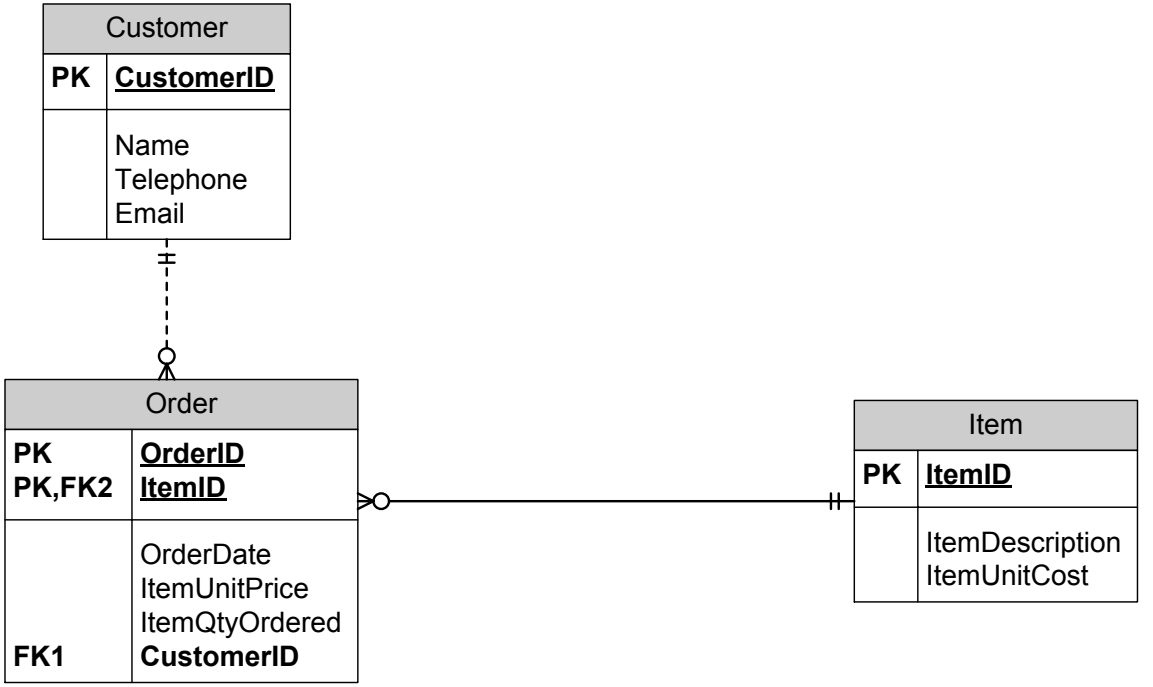
CustomerID	Name	Telephone	Email
100	Joe's Diner	775-231-1234	joediner@gmail
200	Red Robin	831-451-8999	redrobin@rr
300	Soup-to-Nuts	775-222-1201	stn@yahoo



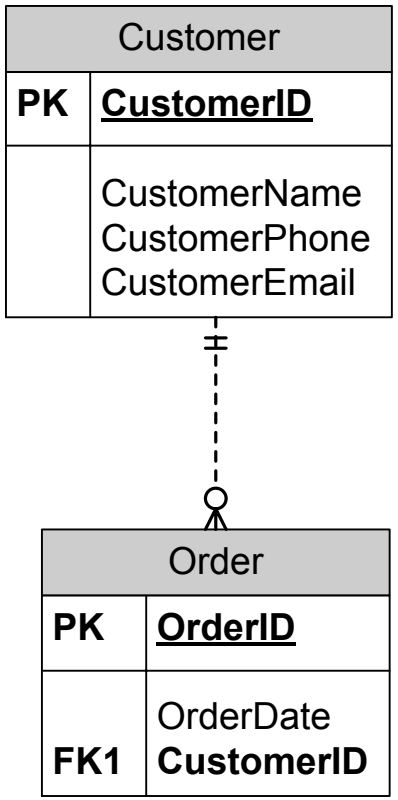
OrderID	OrderDate	ItemQty Ordered	ItemUnit Price	CustomerID
1234	01/19/2025	16	22.95	100
1234	01/19/2025	10	8.95	100
1234	01/19/2025	2	85.95	100
1234	01/19/2025	12	21.95	100
3900	01/20/2025	2	85.95	300
3900	01/20/2025	8	22.95	300
3900	01/20/2025	4	21.95	300
4218	01/15/2025	1200	4.95	200
4218	01/15/2025	1800	11.95	200
4218	01/15/2025	1650	12.95	200
4560	01/17/2025	15	8.95	300

Foreign Key

- A way of supporting the relationship between two tables.
- The primary key of one table is added to another table in order to link the two tables together.
- In a 1:M relationship, the primary key of the entity on the “1” side of the relationship is added to the entity on the “M” side of the relationship. Once added to the “M” side of the relationship, it is called a “foreign key” in that entity.



OrderID	ItemID	OrderDate	ItemQty Ordered	ItemUnit Price	CustomerID
1234	A43	01/19/2025	16	22.95	100
1234	B97	01/19/2025	10	8.95	100
1234	C71	01/19/2025	2	85.95	100
1234	A46	01/19/2025	12	21.95	100
3900	C71	01/20/2025	2	85.95	300
3900	A43	01/20/2025	8	22.95	300
3900	A46	01/20/2025	4	21.95	300
4218	B97	01/15/2025	1200	4.95	200
4218	A43	01/15/2025	1800	11.95	200
4218	A42	01/15/2025	1650	12.95	200
4560	B97	01/17/2025	15	8.95	300



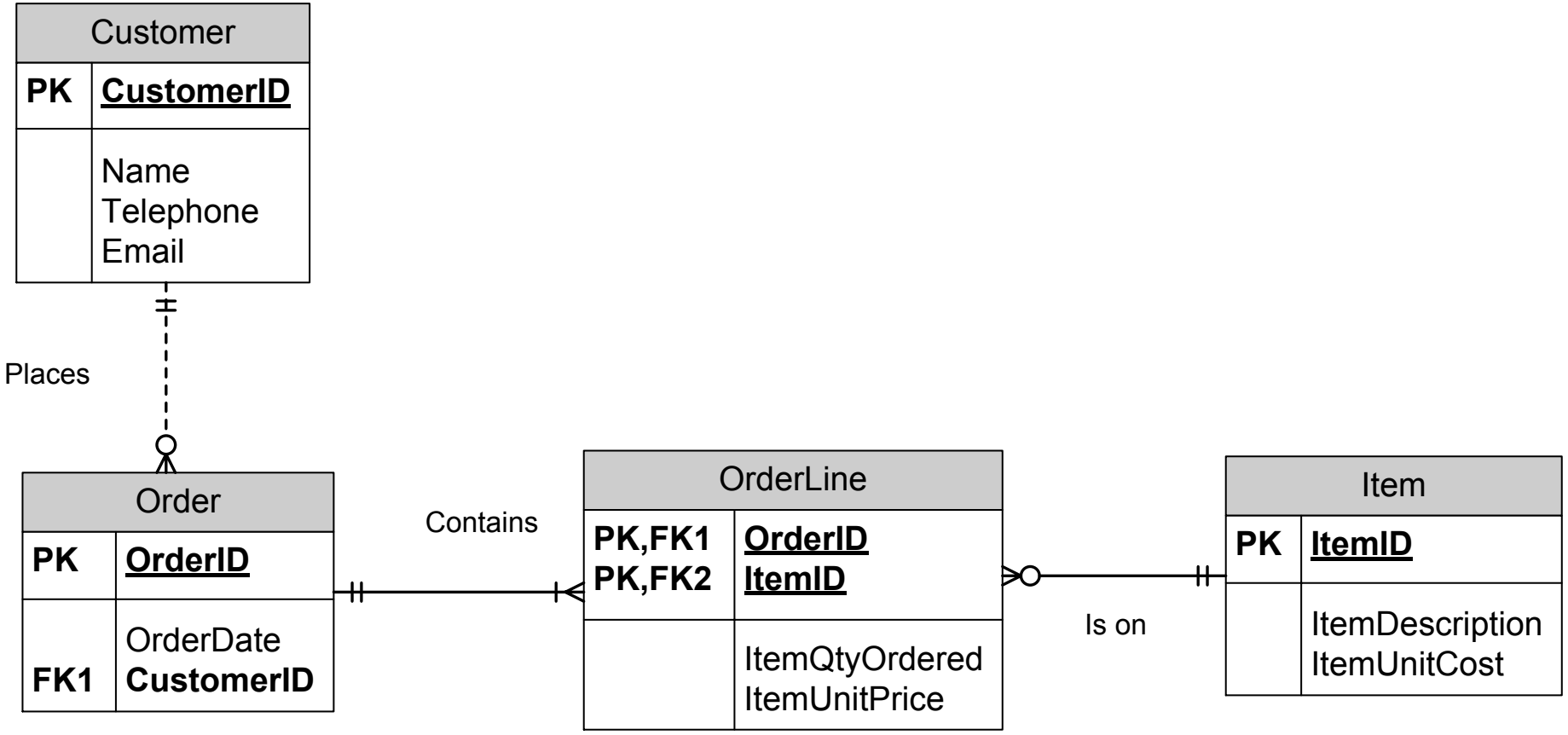
Item	
PK	**ItemID**
	ItemDescription ItemUnitCost

Associative Entity – sample data table

This is the foreign key

OrderID	OrderDate	CustomerID
1234	1/19/2025	100
3900	1/20/2025	300
4218	1/15/2025	200
4560	1/17/2025	300

This is the primary key


Customer
PK
CustomerID
Name
Telephone
Email
Places
Order
PK
OrderID
OrderDate
FK1
CustomerID
Contains
OrderLine
PK,FK1
PK,FK2
OrderID
ItemID
ItemQtyOrdered
ItemUnitPrice
Is on
Item
PK
ItemID
ItemDescription
ItemUnitCost

This is the foreign key for one relationship

This is the foreign key for the second relationship

Intersection Entity sample data table

OrderID	ItemID	Quantity	Price
1234	A43	16	22.95
1234	B97	10	8.95
1234	C71	2	85.95
1234	A46	12	21.95
3900	C71	2	85.95
3900	A43	8	22.95
3900	A46	4	21.95
4218	B97	1200	4.95
4218	A43	1800	12.95
4218	A46	1650	11.95
4560	B97	15	8.95

Together, these two fields (orderID and ItemID) create a concatenated, natural primary key

OrderID	OrderDate	CustomerID
1234	1/19/2025	100
3900	1/20/2025	300
4218	1/15/2025	200
4560	1/17/2025	300

CustomerID	Name	Telephone	Email
100	Joe's Diner	775-231-1234	joediner@gmail
200	Red Robin	831-451-8999	redrobin@rr
300	Soup-to-Nuts	775-222-1201	stn@yahoo

OrderID	ItemID	ItemQty Ordered	ItemUnit Price
1234	A43	16	22.95
1234	B97	10	8.95
1234	C71	2	85.95
1234	A46	12	21.95
3900	C71	2	85.95
3900	A43	8	22.95
3900	A46	4	21.95
4218	B97	1200	4.95
4218	A43	1800	11.95
4218	A42	1650	12.95
4560	B97	15	8.95

ItemID	ItemDescription	ItemUnit Cost
A43	Tumbler, 12 oz 8 pack	9.20
B97	Salt Shakers – 4 pack	2.25
C71	Crestwood Blender	67.15
A46	Tumbler, 6 oz 8 pack	7.60
A42	Tumbler, 16 oz 8 pack	10.15

Beginning list of the rules of a relational database

Customer	
PK	**CustomerID**
	Name Telephone Email

Order	
PK	**OrderID**
	OrderDate CustomerID

OrderLine	
PK PK	**OrderID** **ItemID**
	ItemQtyOrdered ItemUnitPrice

Item	
PK	**ItemID**
	ItemDescription ItemUnitCost

All data is stored in two-dimensional tables.

Each table must have a column (or set of columns that are concatenated) that serves as a primary key.

A primary key must have a unique data value for each row in the table.

The intersection of a row and column (a cell) can have only one data value.

A table can have (theoretically) an unlimited number of rows.

What is database design?

- Database design is the process of creating the structure or blueprint of stored data for an organization.
- The goals of database design are to produce a structure that:
  - Protects the integrity of the data;
  - Can be changed relatively easily; and
  - Stores all data required for organizational processing and decision making.

About Entities

Entities are nouns that describe the person, event, place, or thing about which we want to store data.

The name of an entity is singular, not plural.

An entity usually becomes a table in a database.

An entity instance is a single occurrence of an entity (think a row in a table, or a record in a file). An entity almost always has more than one entity instance.

About Attributes

- Attribute: a characteristic of an entity. Attributes are also called “fields” or “columns.”
- Primary Key: an attribute or group of attributes that contain/store a unique data value for each entity instance.
  - A “natural” primary key is an attribute or group of attributes that is part of the data set provided by the end user of the database.
  - A “surrogate” primary key is a single attribute created by a database designer when a natural primary key is not available in the existing data set. A surrogate should NEVER be concatenated.
  - A primary key is considered to be a “constraint” on a table because it limits the data that can be entered into the primary key (must be unique).

What does a database designer do?

- Divides the one large entity into a series of entities.
- The goal is to have one instance of one entity represent a unique value; data is not repeated in more than one entity and it is not repeated in more than one entity instance and it is not repeated in more than one column.
- For example, data for a given customer is placed in only one entity and a single customer is stored in only one row of that entity – it is not repeated in any other rows.
- As an example, the name of the customer would be in only one column of one row of one table (entity).

Week 2 Real Estate exercise

- The database should keep track of homes and owners. An owner can possess one or more homes. A home can be owned by one or more owners. We want to store the percent owned of a home by an owner.

Home ID	Address	City	State	Zip	#Bed rooms	#Bath rooms	Square Feet	Owner ID	Owner Name	Owner Phone Number	OwnerEmail	Percent Owned
100	123 Oak Street	Reno	NV	89502	3	2	1860	180	Martinson	775-782-1233	martin@gmail.com	85%
100	123 Oak Street	Reno	NV	89502	3	2	1860	213	Krandall	702-547-4402	Kone@hotmail.com	15%
200	762 Elm Street	Sparks	NV	89431	6	3.5	3540	165	Jacoby	775-583-4834	jboy@yahoo.com	100%
300	7812 BreathView Way	San Diego	CA	92124	3	1.5	1250	180	Martinson	775-782-1233	martin@gmail.com	25%
300	7812 BreathView Way	San Diego	CA	92124	3	1.5	1250	213	Krandall	702-547-4402	Kone@hotmail.com	15%
300	7812 BreathView Way	San Diego	CA	92124	3	1.5	1250	294	Salinas	971-887-4612	JuanSal@gmail.com	60%
400	6542 Grandstand Circle	Reno	NV	89508	2	1	980	213	Krandall	702-547-4402	Kone@hotmail.com	50%
400	6542 Grandstand Circle	Reno	NV	89508	2	1	980	400	Nguyen	560-432-3300	NgNg@smw.com	50%
500	12 Corriente	San Diego	CA	92127	3	2.5	2310	180	Martinson	775-782-1233	martin@gmail.com	100%

Let's start: divide the data into separate entities

Home	
PK	**HomeID**
	Address City State Zip NumberBedrooms NumberBathrooms SquareFeet

Owner	
PK	**OwnerID**
	Name PhoneNumber Email

Know what data will be stored in each entity with sample data tables

Home ID	Address	City	State	Zip	#Bed rooms	#Bath rooms	Square Feet
100	123 Oak Street	Reno	NV	89502	3	2	1860
200	762 Elm Street	Sparks	NV	89431	6	3.5	3540
300	7812 BreathView Way	San Diego	CA	92124	3	1.5	1250
400	6542 Grandstand Circle	Reno	NV	89508	2	1	980
500	12 Corriente	San Diego	CA	92127	3	2.5	2310

Owner ID	OwnerName	OwnerPhone Number	OwnerEmail
180	Martinson	775-782-1233	martin@gmail.com
213	Krandall	702-547-4402	Kone@hotmail.com
165	Jacoby	775-583-4834	jboy@yahoo.com
294	Salinas	971-887-4612	JuanSal@gmail.com
400	Nguyen	560-432-3300	NgNg@smw.com

Why can't we store data in a many-to-many relationship without data redundancy?

Because this is what happens if you store the percent owned in the Home table.

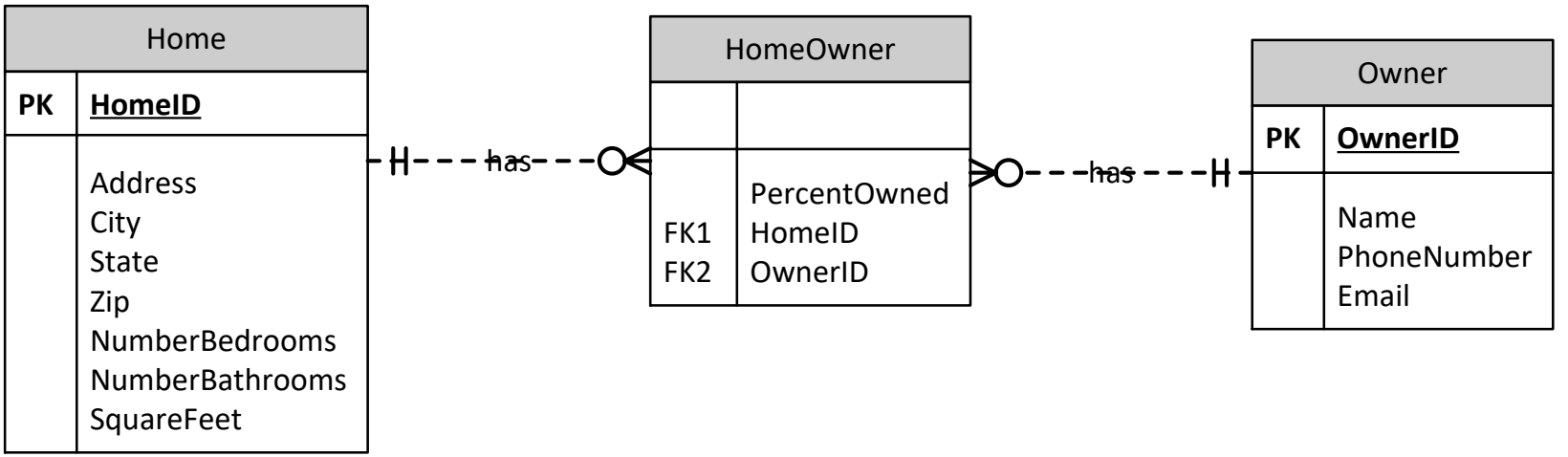
Home ID	Address	City	State	Zip	Number Bedrooms	Number Bathrooms	Square Feet	OwnerID	Percent Owned
100	123 Oak Street	Reno	NV	89502	3	2	1860	180	85
100	123 Oak Street	Reno	NV	89502	3	2	1860	213	15
200	762 Elm Street	Sparks	NV	89431	6	3.5	3540	165	100
300	7812 BreathView Way	San Diego	CA	92124	3	1.5	1250	180	25
300	7812 BreathView Way	San Diego	CA	92124	3	1.5	1250	213	15
300	7812 BreathView Way	San Diego	CA	92124	3	1.5	1250	294	60
400	6542 Grandstand Circle	Reno	NV	89508	2	1	980	213	50
400	6542 Grandstand Circle	Reno	NV	89508	2	1	980	400	50
500	12 Corriente	San Diego	CA	92127	3	2.5	2310	180	100

Why can't we store data in a many-to-many relationship without data redundancy?

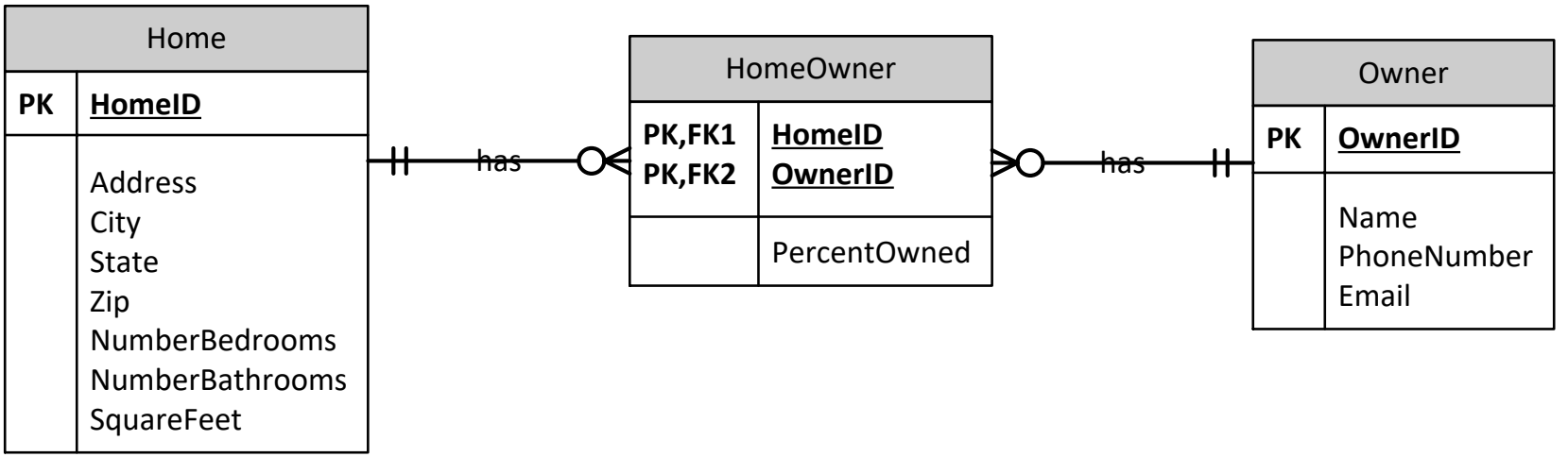
Because this is what happens if you store the percent owned in the Owner table.

Owner ID	OwnerName	OwnerPhone Number	OwnerEmail	HomeID	Percent Owned
180	Martinson	775-782-1233	martin@gmail.com	100	85
180	Martinson	775-782-1233	martin@gmail.com	300	25
180	Martinson	775-782-1233	martin@gmail.com	500	100
213	Krandall	702-547-4402	Kone@hotmail.com	100	15
213	Krandall	702-547-4402	Kone@hotmail.com	300	15
213	Krandall	702-547-4402	Kone@hotmail.com	400	50
165	Jacoby	775-583-4834	jboy@yahoo.com	200	100
294	Salinas	971-887-4612	JuanSal@gmail.com	300	60
400	Nguyen	560-432-3300	NgNg@smw.com	400	50

Create a new table to store the percent owned that intersects the two tables



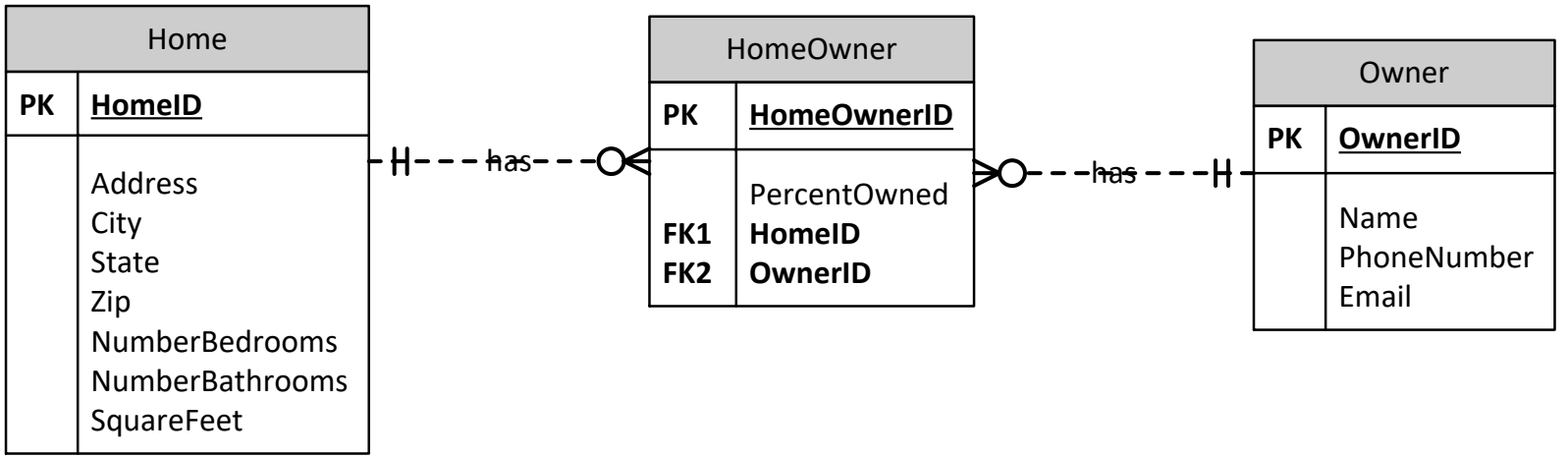
Make a natural primary key



Data in the intersection table - HomeOwner

HomeID	OwnerID	PercentageOwned
100	180	85%
100	213	15%
200	165	100%
300	180	25%
300	213	15%
300	294	60%
400	213	50%
400	400	50%
500	180	100%

Make a surrogate primary key



Three separate physical tables

Home ID	Address	City	State	Zip	Number Bedrooms	Number Bathrooms	Square Feet
100	123 Oak Street	Reno	NV	89502	3	2	1860
200	762 Elm Street	Sparks	NV	89431	6	3.5	3540
300	7812 BreathView Way	San Diego	CA	92124	3	1.5	1250
400	6542 Grandstand Circle	Reno	NV	89508	2	1	980
500	12 Corriente	San Diego	CA	92127	3	2.5	2310

Owner ID	OwnerName	OwnerPhone Number	OwnerEmail
180	Martinson	775-782-1233	martin@gmail.com
213	Krandall	702-547-4402	Kone@hotmail.com
165	Jacoby	775-583-4834	jboy@yahoo.com
294	Salinas	971-887-4612	JuanSal@gmail.com
400	Nguyen	560-432-3300	NgNg@smw.com

Home OwnerID	HomeID	OwnerID	Percent Owned
1	100	180	85%
2	100	213	15%
3	200	165	100%
4	300	180	25%
5	300	213	15%
6	300	294	60%
7	400	213	50%
8	400	400	50%
9	500	180	100%

Heuristics: Redundant Data

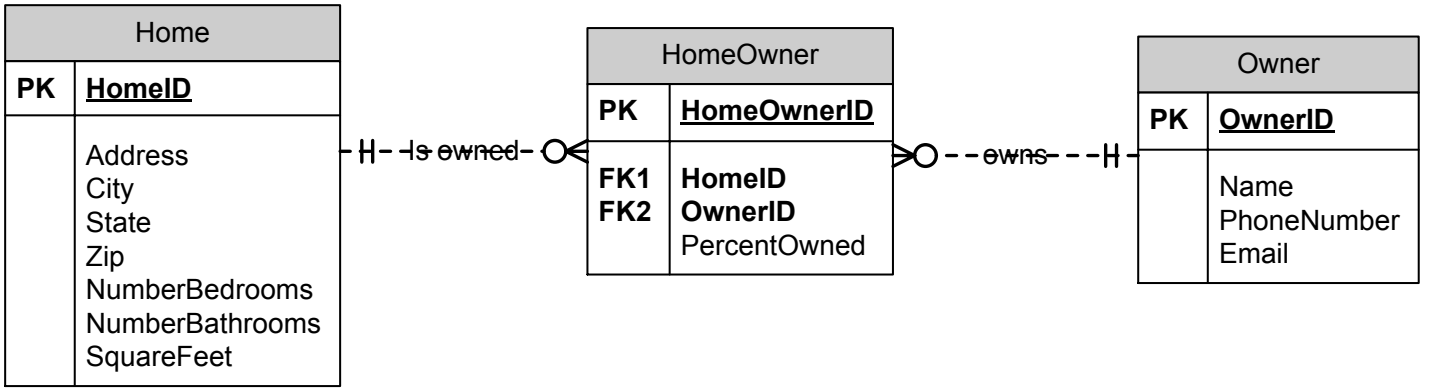
Avoid redundant data that is composed of long alphanumeric data types.

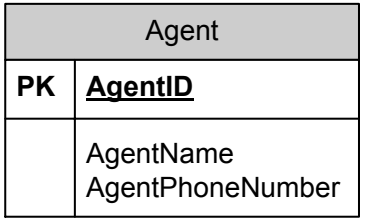
- Examples are names, addresses, comments, notes.
- Standardize any “descriptive” attributes such as categories or types.

Put sample data in a few rows of each entity so that you can determine whether or not the data will be redundant.

- If you don’t know what data will be stored, ask your client.

Step 2 – make an entity for the real estate agent

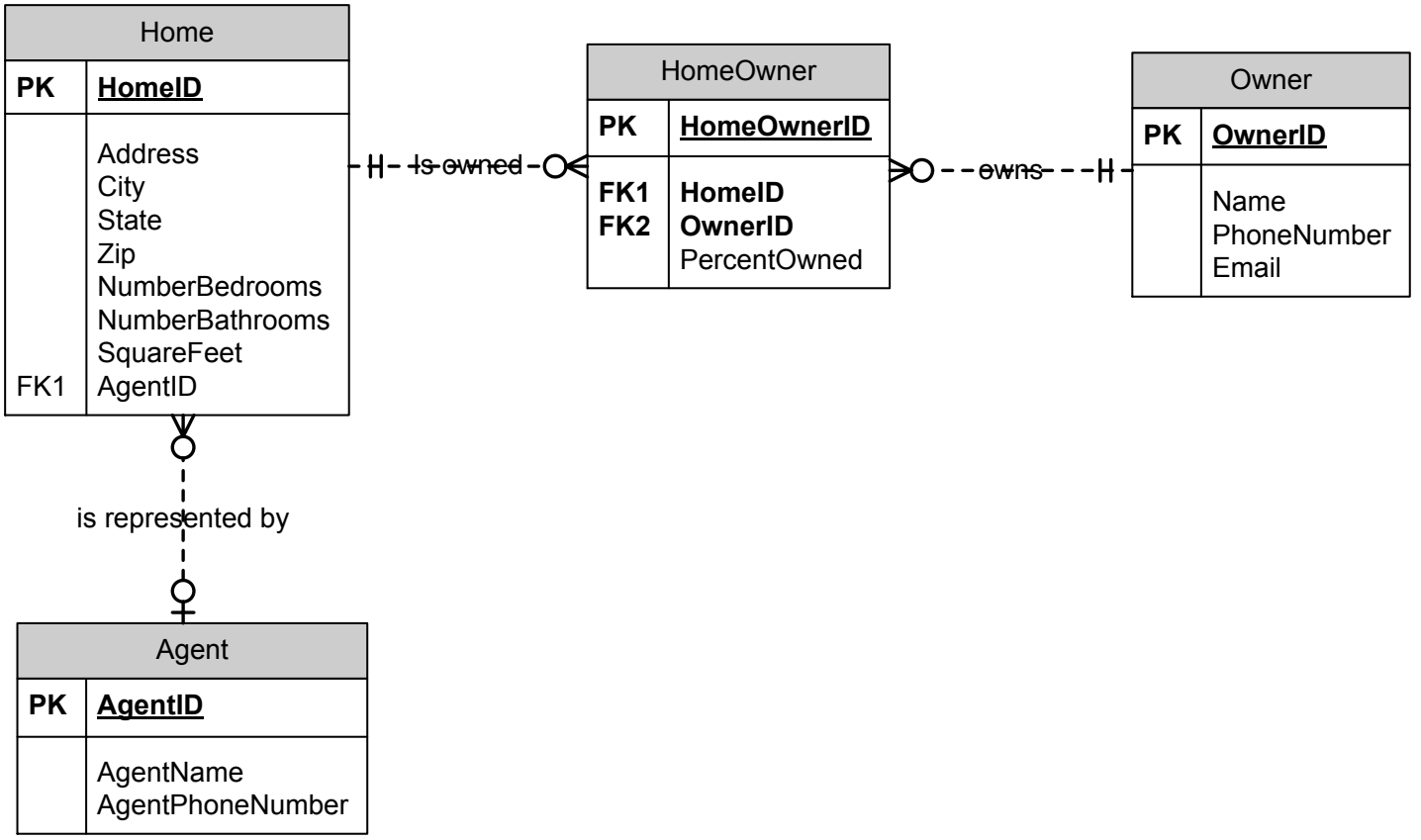




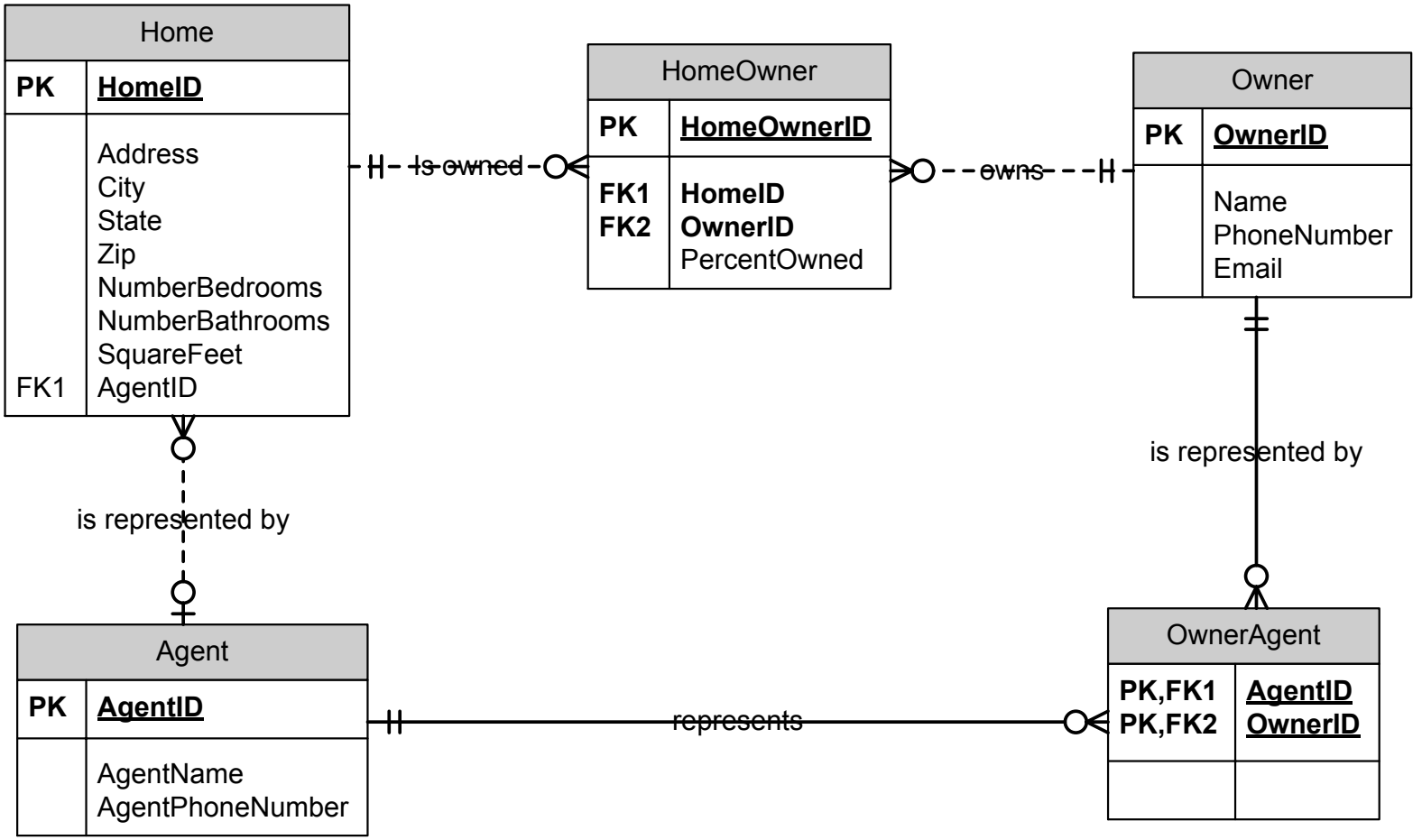
Deciding which business rules to capture…

- An agent could represent more than one owner, an owner could work with more than one agent.
- An agent could represent more than one home, a home has a single agent.
  - Are both relationships important? Do we need both?
  - Can each owner of a single home have a different agent? For example, if the people who own a home together aren't related, maybe each one wants their own agent.

Step 2 – first part of rule that a home is represented by one agent results in a 1:M relationship between Home and Agent



Step 2 – second part of the rule could result in a many-to-many relationship between owner and agent (assuming that an owner is also a "buyer"). This would require an intersection entity shown as OwnerAgent on the ERD.



Step 2 – second part of the rule could result in a many-to-many relationship between owner and agent (assuming that an owner is also a "buyer"). This would require an intersection entity shown as OwnerAgent on the ERD.