upGrad

Data Management and Relational Modelling

Course: Data Engineering - I

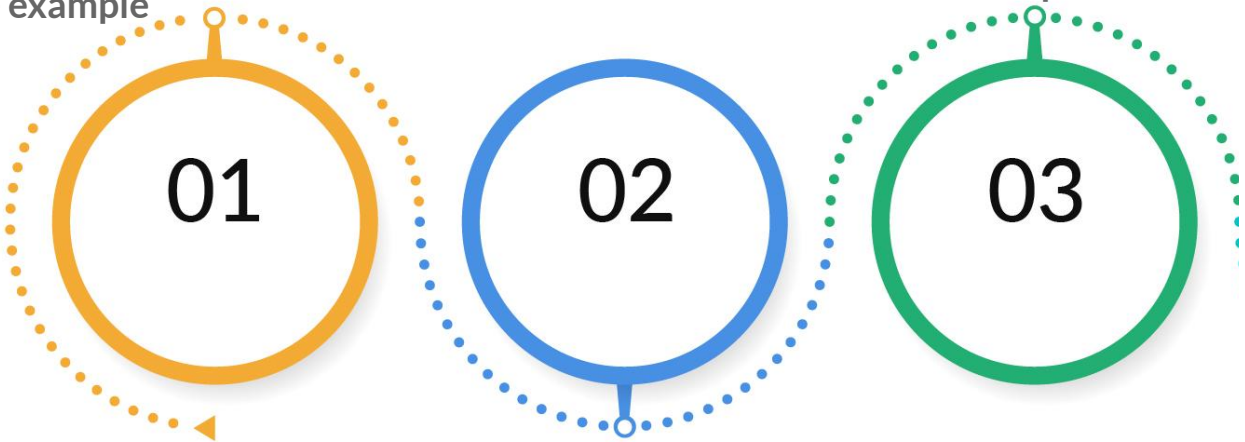
Lecture On: Data Management and Relational Modelling

Instructor: Vishwa Mohan

Session 5 | Use Case Example

Session Overview

Understanding the attributes of a use case example



Identifying any partial functional dependency and transitive dependencies.

Identifying the main entities

When is Data Normalization required?

Consider an Example

01

A food delivery company has provided the restaurant menu on its app and offers delivery services to its customers.

Once an order is received by the restaurant, a delivery employee is selected to deliver the order.

02

It has defined various attributes that are used by the company to store information.

We are given the attributes, and we have to design the logical layer of our database.

Consider All These Attributes

Attributes
Order ID
Order Date
Restaurant Name
Restaurant Address
Restaurant Phone Number
Restaurant Type

Attributes
Restaurant Category
Restaurant Membership
Customer Name
Customer Phone Number
Customer Address
Item Purchased

Cont'd...

Consider All These Attributes

Attributes
Item Quantity
Item Type
Item Category
Customer Membership
Coupon Applied
Price Paid

Attributes
Payment Mode
Delivery Employee Name
Vehicle Number
Pickup Time
Delivered Time
Total Delivery Time

Why Does This Data Need to Be Modelled?

Order ID	Order Date	Restaurant Name	Restaurant Address	Restaurant Phone Number	Restaurant Type	Restaurant Category	Restaurant Membership
10	20Jan2020	South indian Food	plot 5, mumbai	9768546320	South Indian	Veg	Plan Premium

Customer Name	Customer Phone Number	Customer Address	Customer Membership	Item Purchased	Item Quantity	Item Type	Item Category
Virat	9765438210	plot 10, mumbai	Plan premium	idli, dosa	5, 2	South Indian, South Indian	Veg, Veg

Coupon Applied	Price Paid	Payment Mode	Delivery Employee Name	Vehicle Number	Pickup Time	Delivered Time	Total Delivery Time
COUPON12	200	Cash	Rohit	VEH1234	5:30 pm	6:00 pm	30 mins

The Main Entities

Customer
Customer ID
Name
Phone Number
Customer Address
City
State
Customer Membership
Customer Membership Price

Restaurant
Restaurant ID
Name
Type
Category
Address
City
State

Delivery Employee
Delivery Employee ID
Name
Vehicle Number
Date of joining
Membership Plan
Membership Plan Details

The Item Entity

Weak Entity

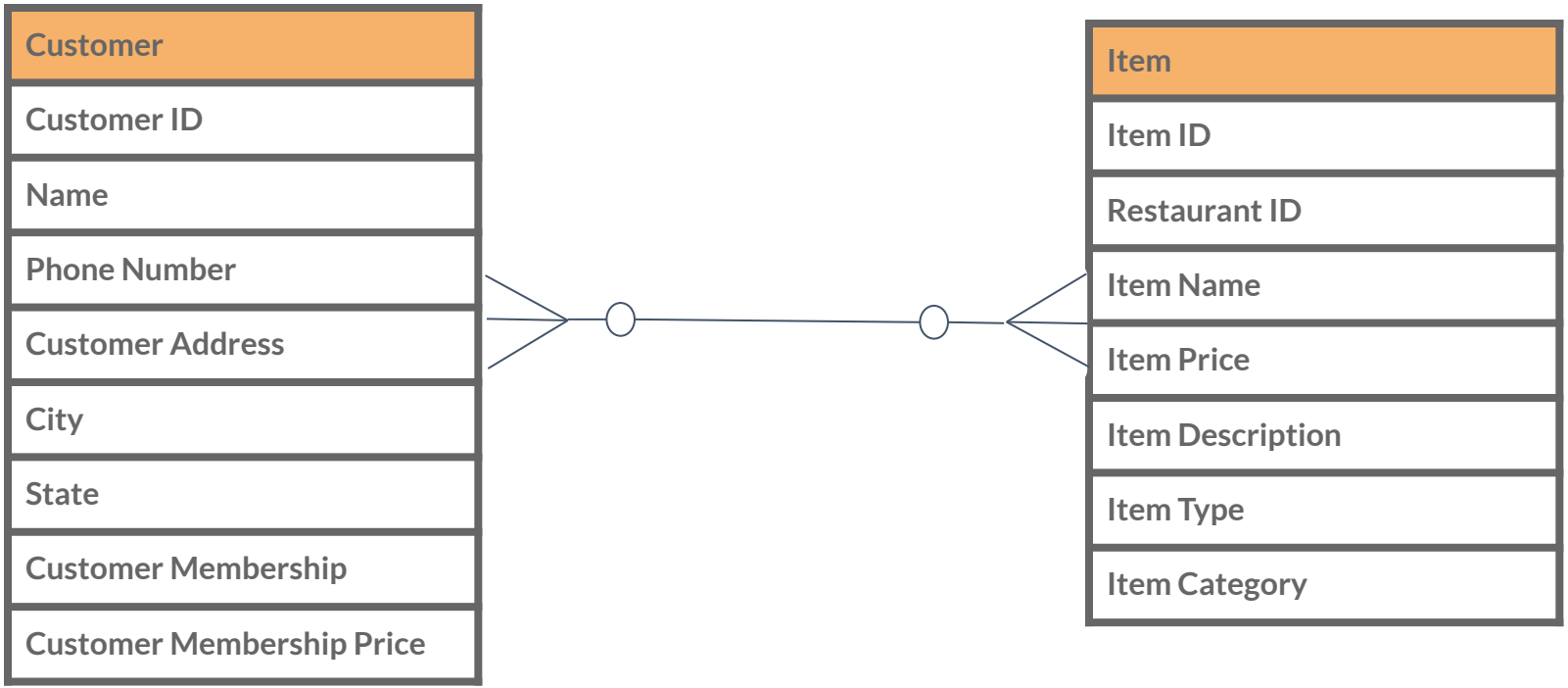
Item
Item ID
Restaurant ID
Item Name
Item Price
Item Description
Item Type
Item Category

Strong Entity

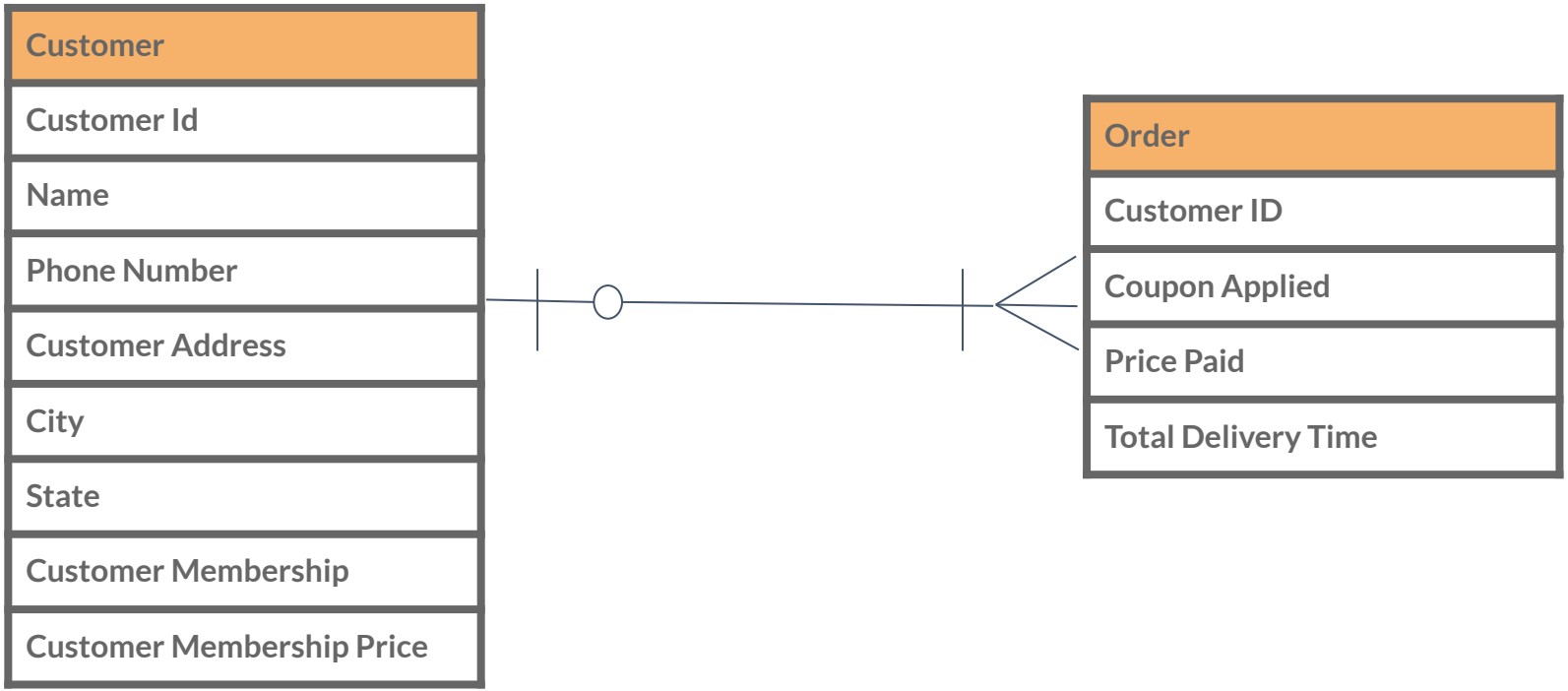
Restaurant
Restaurant ID
Name
Type
Category
Address
City
State

Customer and Item Entities

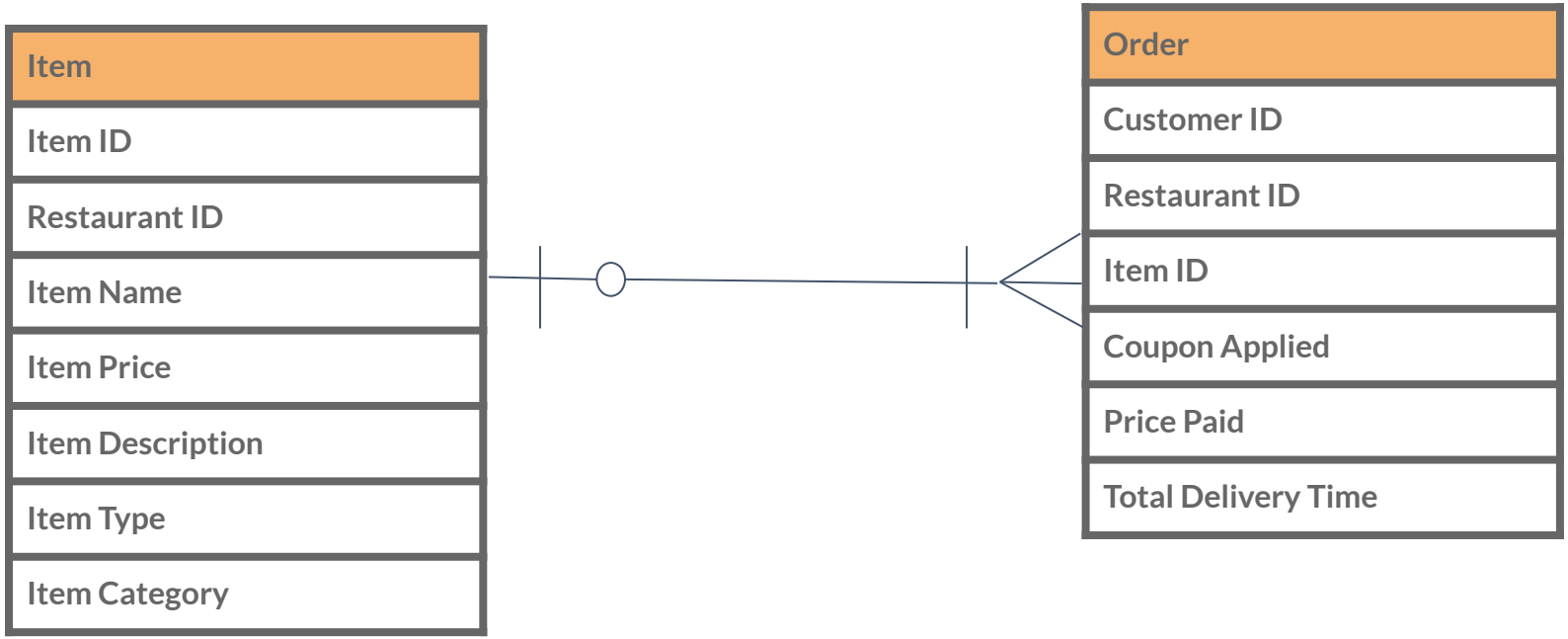
The relation between the customer and item tables is many-to-many.
To implement a many-to-many relation, there will be one more entity 'order' that has a many-to-one relation with customer table as well as order table.



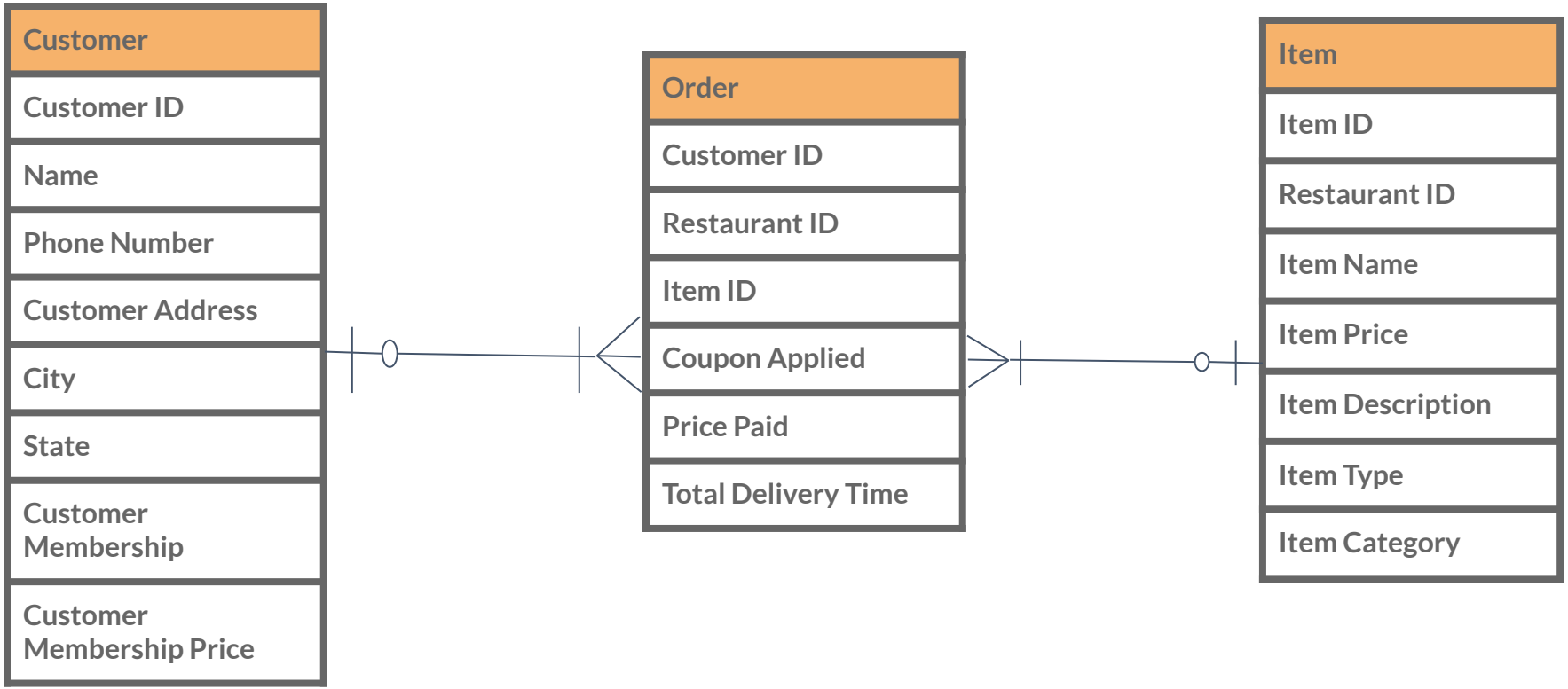
The Order Entity



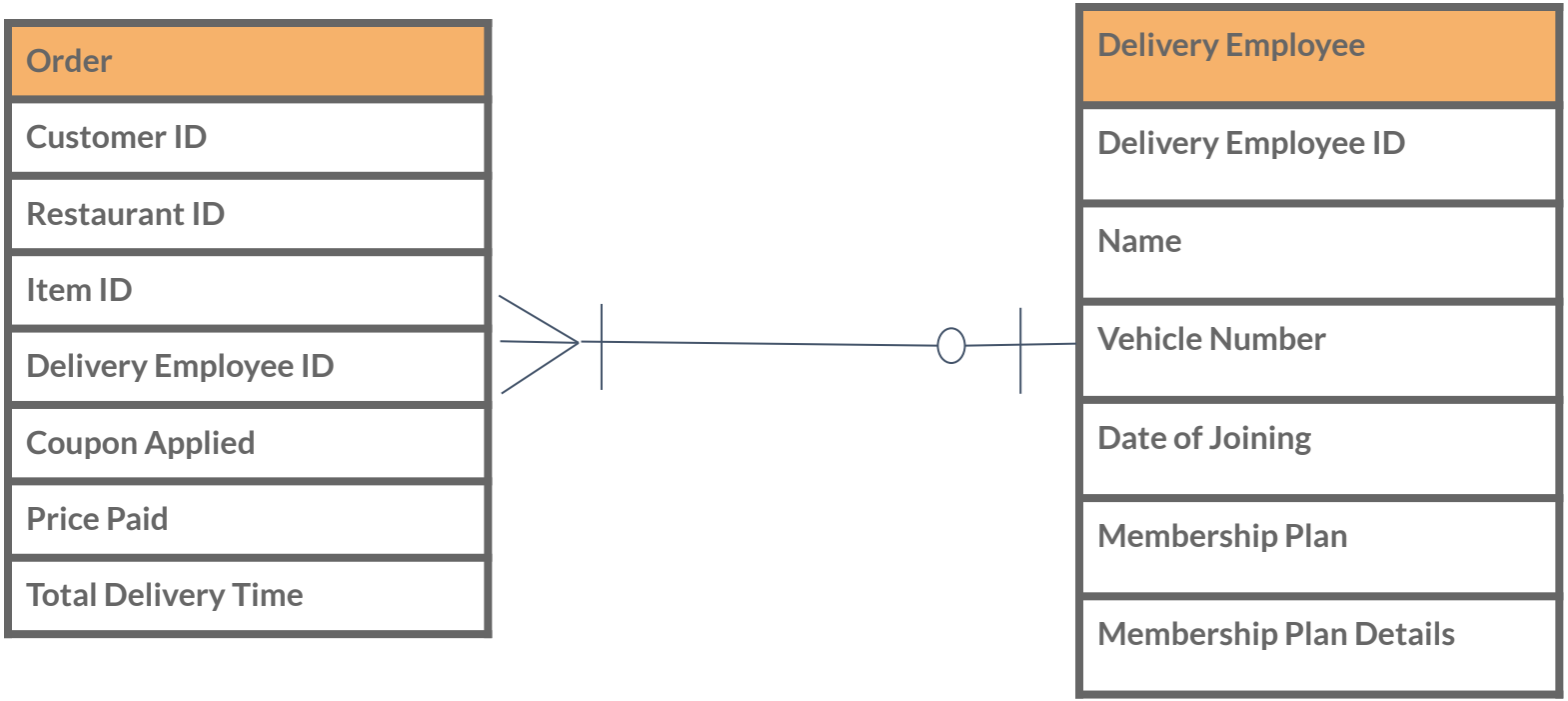
The Order Entity



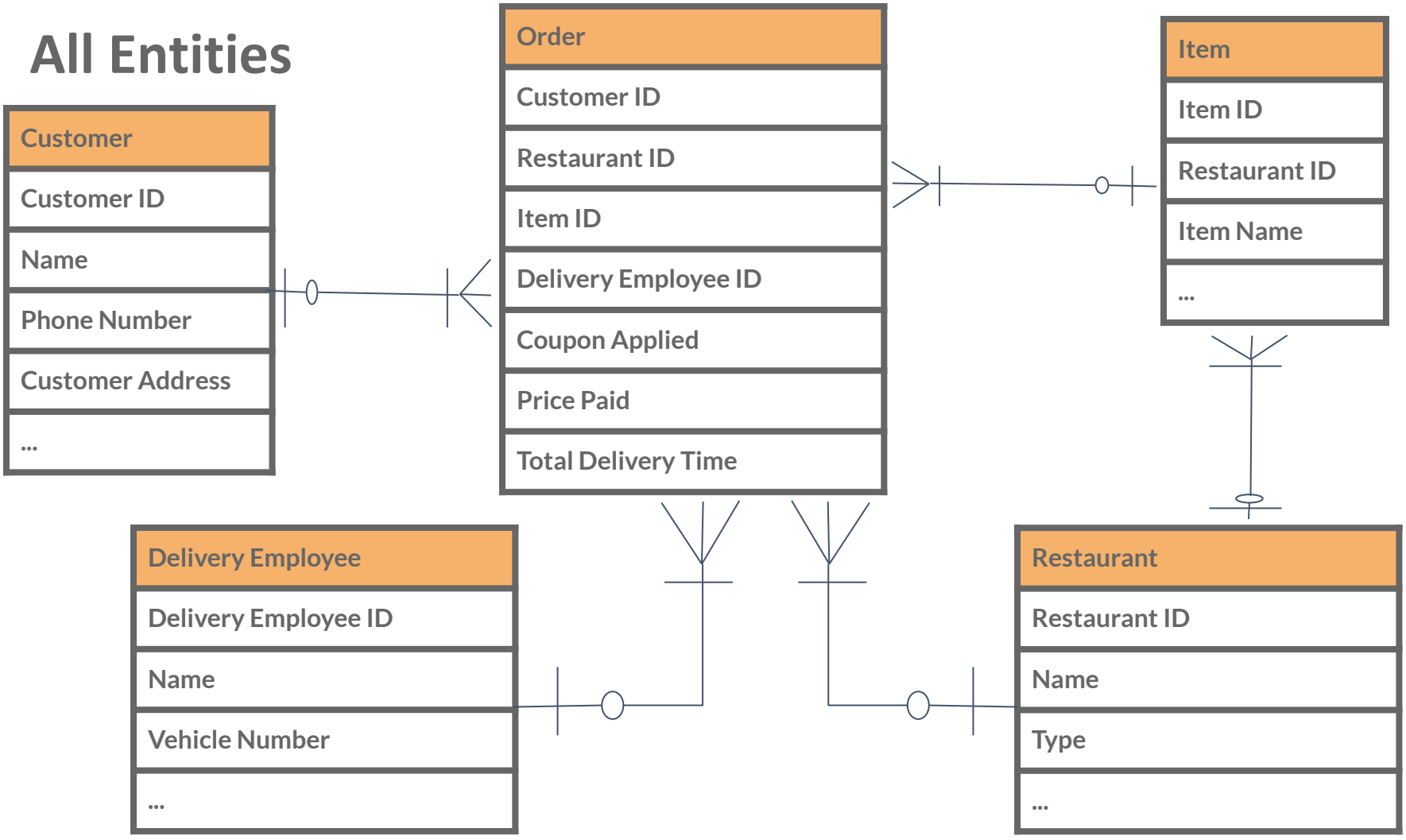
The Order Entity



The Order Entity



All Entities



Functional Dependencies

Order
Customer ID
Restaurant ID
Item ID
Delivery Employee ID
Coupon Applied
Price Paid
Total Delivery Time

There is no composite key in this table to uniquely identify every record.

If there is an attribute order ID for each order, there can be a composite key.

One order can have two items. This will cause the order ID to be the same for two rows. Thus, it cannot act as a primary key.

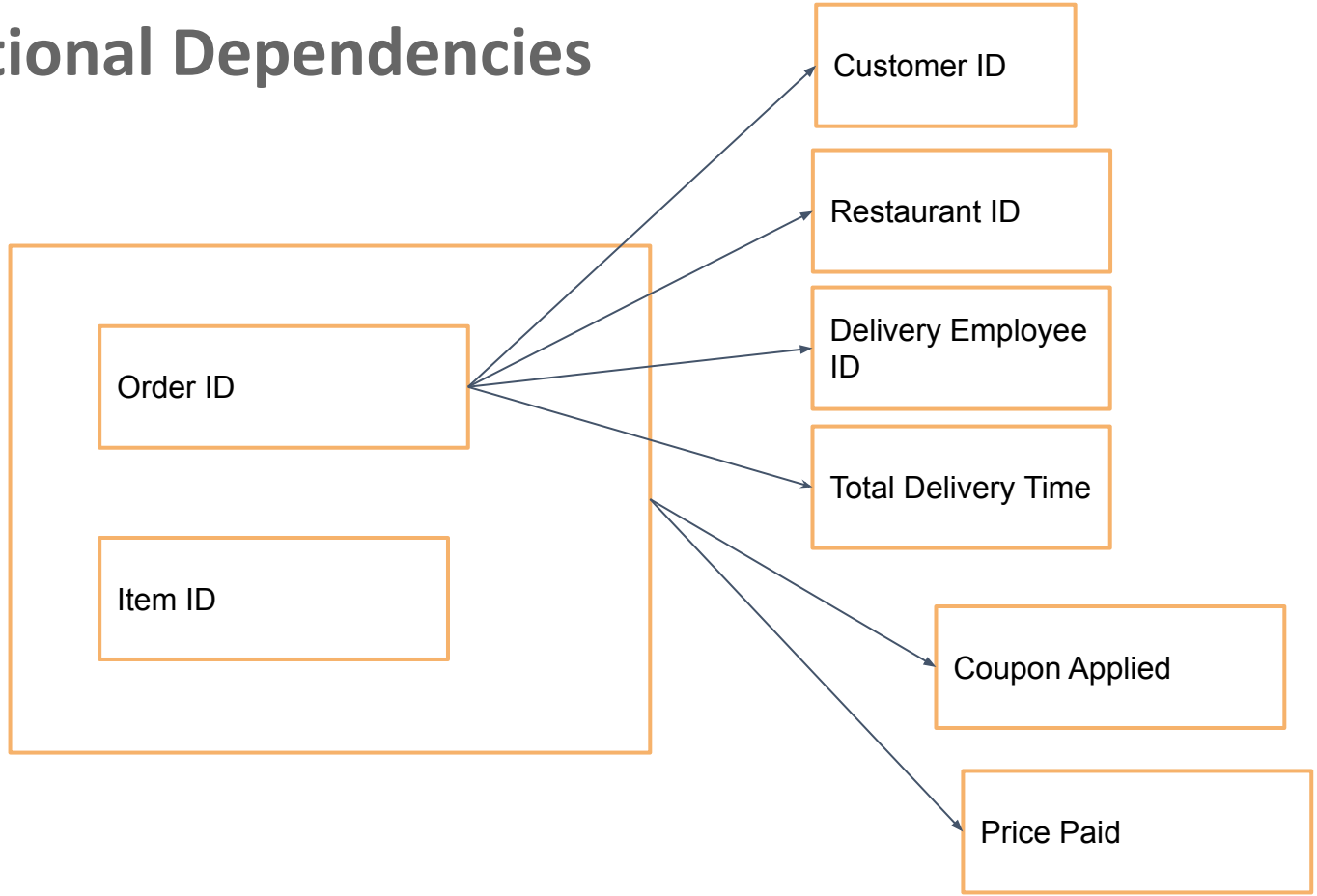
Order
Order ID
Customer ID
Restaurant ID
Item ID
Delivery Employee ID
Coupon Applied
Price Paid
Total Delivery Time

Functional Dependencies

Order ID	Customer ID	Restaurant ID	Item ID	Delivery Employee ID	Coupon Applied	Price Paid	Total Delivery Time
50	C101	R112	2, 3	D12	C100, C200	120, 130	50 mins

Order ID	Customer ID	Restaurant ID	Item ID	Delivery Employee ID	Coupon Applied	Price Paid	Total Delivery Time
50	C101	R112	2	D12	C100	120	50 mins
50	C101	R112	3	D12	C200	130	50 mins

Functional Dependencies



Functional Dependencies

Order
Order ID
Customer ID
Restaurant ID
Item ID
Delivery Employee ID
Coupon Applied
Price Paid
Total Delivery Time

Order
Order ID
Customer ID
Restaurant ID
Delivery Employee ID
Total Delivery Time

Order Item
Order ID
Item ID
Coupon Applied
Price Paid

The order table is separated into two different tables to remove any partial dependency.

Functional Dependencies

If there is a composite key in the table, partial dependencies may exist. If there is only one primary key, there cannot be any partial dependencies.

Customer
Customer ID
Name
Phone Number
Customer Address
City
State
Customer Membership
Customer Membership Price

Delivery Employee
Delivery Employee ID
Name
Vehicle Number
Date of Joining
Membership Plan
Membership Plan Details

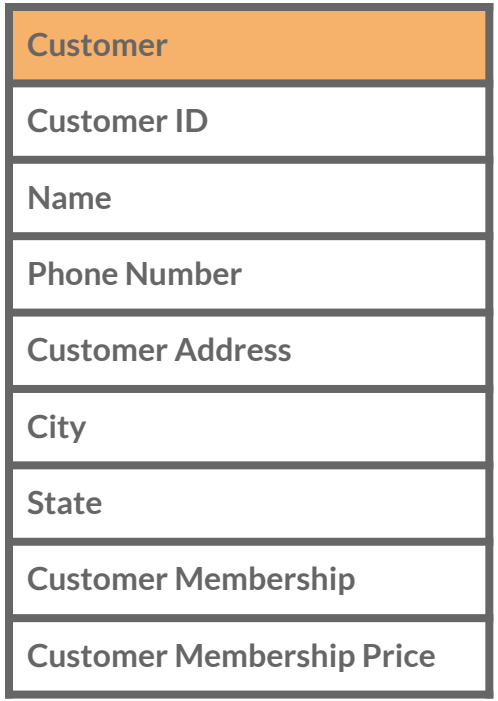
Restaurant
Restaurant ID
Name
Type
Category
Address
City
State

Functional Dependencies

Item entity is a weak entity. It depends on the primary key of restaurant entity to uniquely identify each record in the item table.

Restaurant ID is the foreign key to the restaurant table. Item Id and restaurant ID form the composite key for the table. All the non-prime attributes fully functionally dependent on prime attributes.

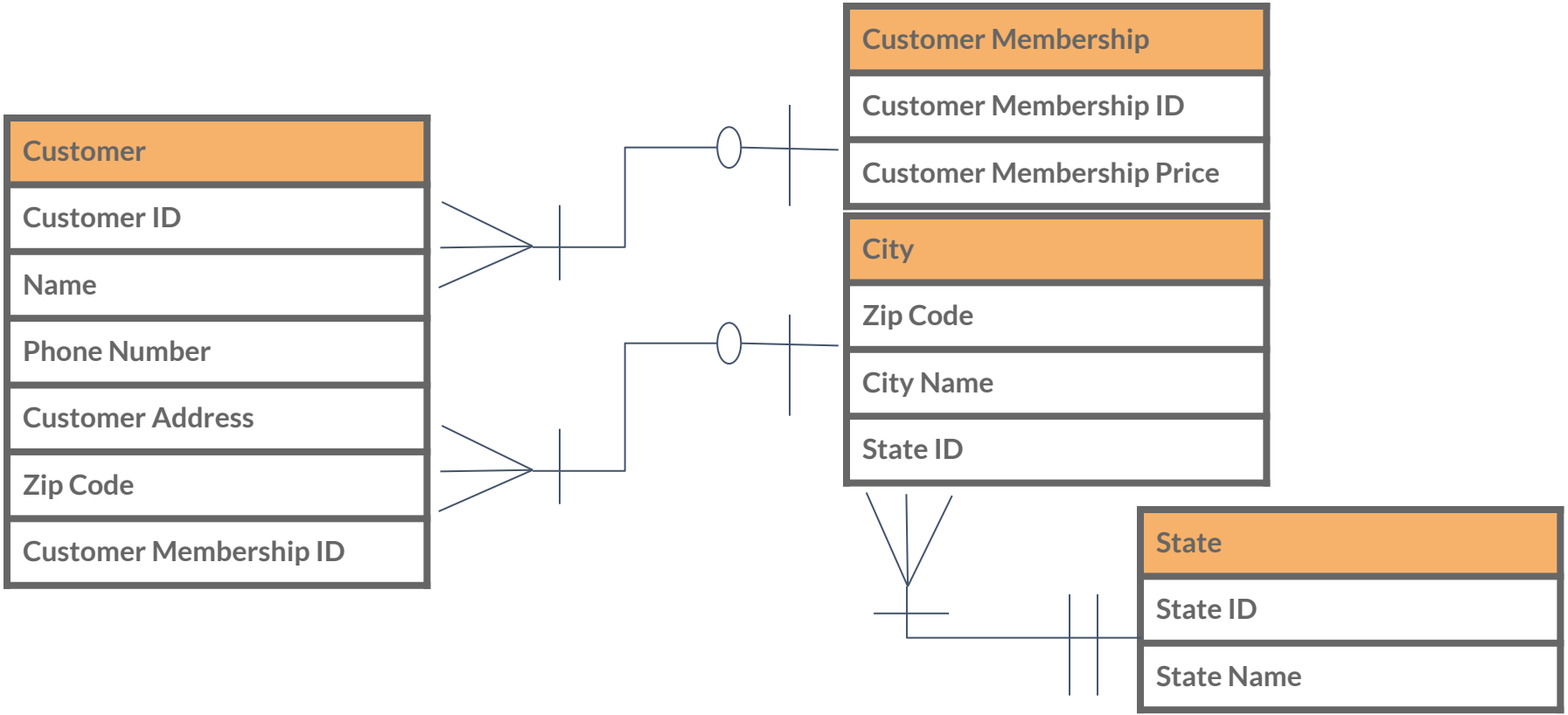
Transitive Dependencies in Customer Entity



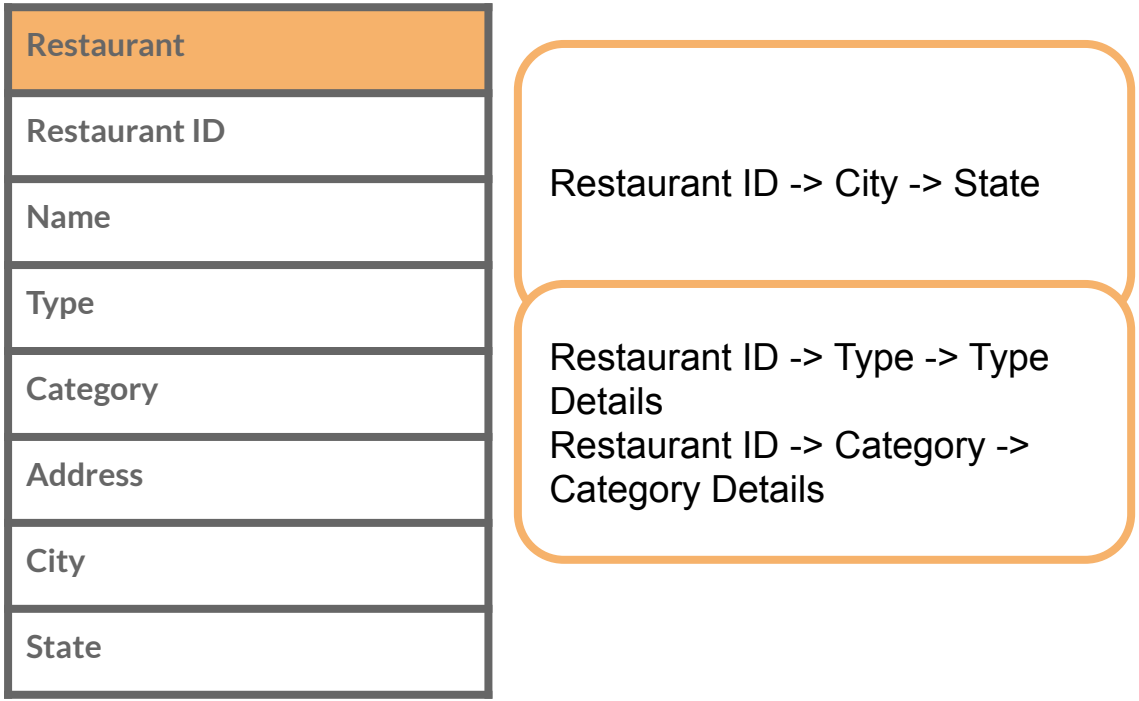
Customer ID -> Customer Membership -> Customer Membership Price

Customer ID -> City -> State

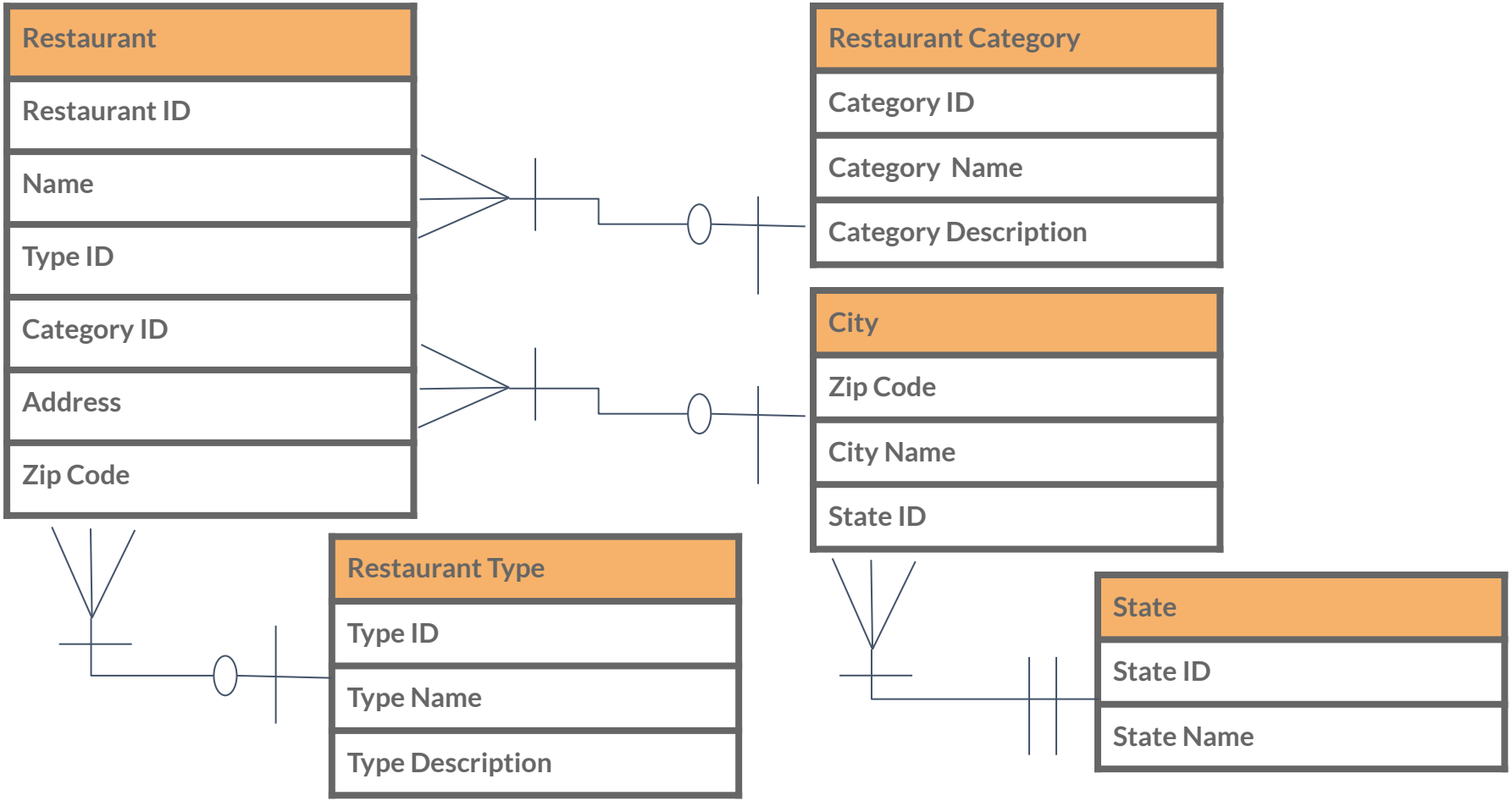
Transitive Dependencies in Customer Entity



Transitive Dependencies in Restaurant Entity



Transitive Dependencies in Restaurant Entity

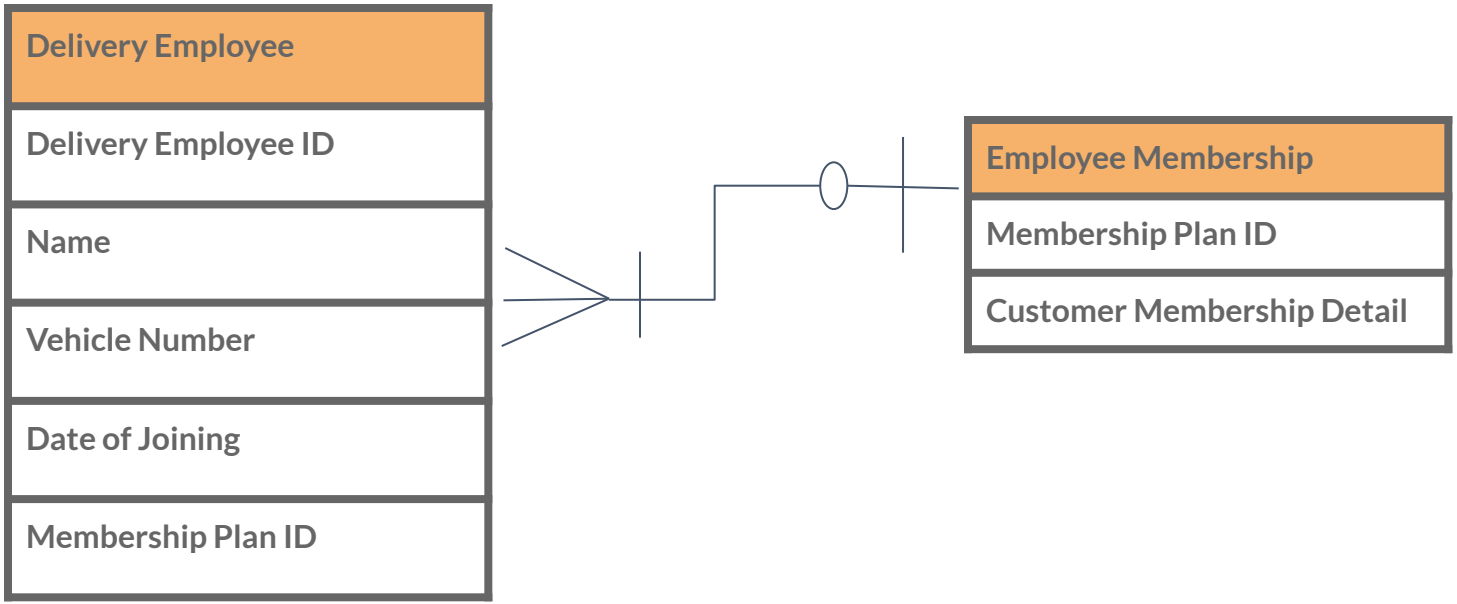


Transitive Dependencies in Delivery Employee Entity

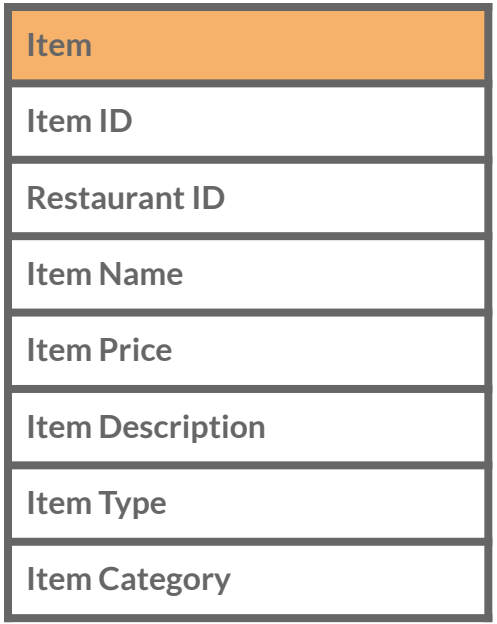
Delivery Employee
Delivery Employee ID
Name
Vehicle Number
Date of Joining
Membership Plan
Membership Plan Details

Delivery Employee ID -> Membership Plan
-> Membership Details

Transitive Dependencies in Delivery Employee Entity

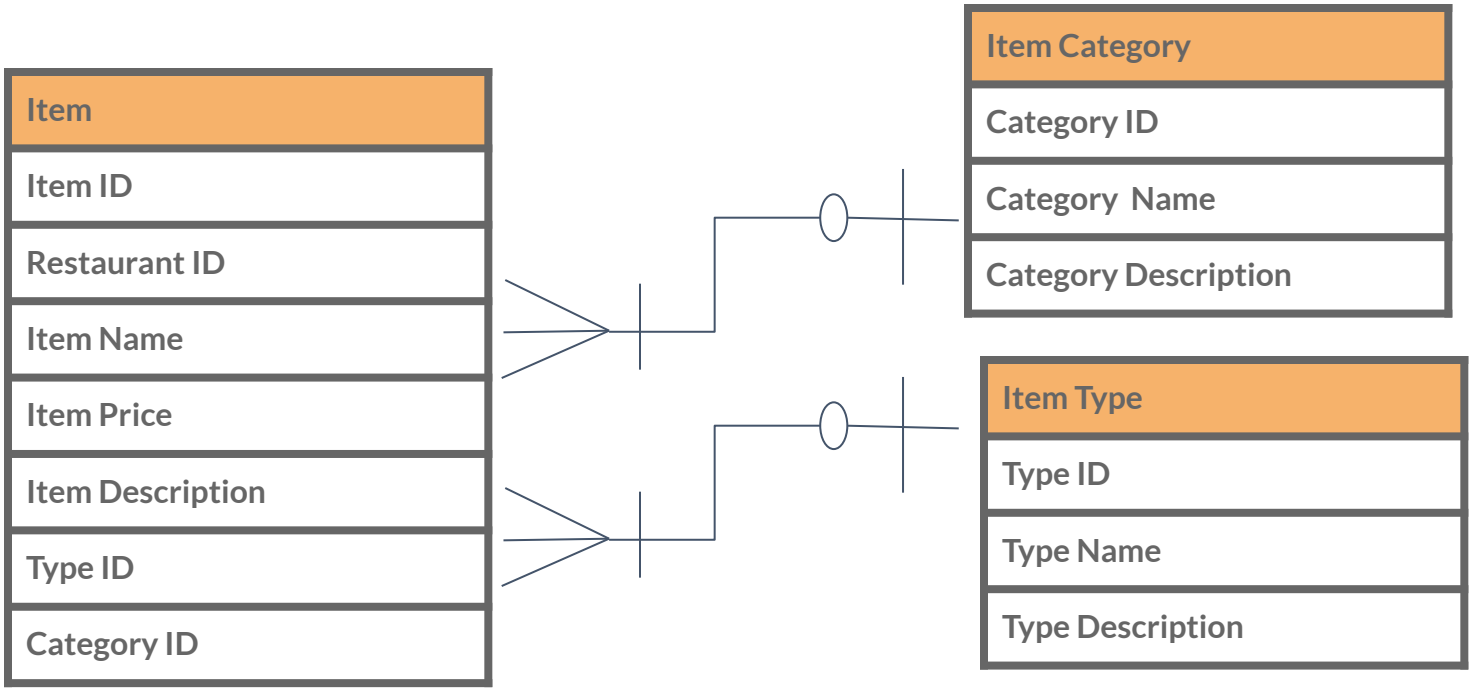


Transitive Dependencies in Item Entity

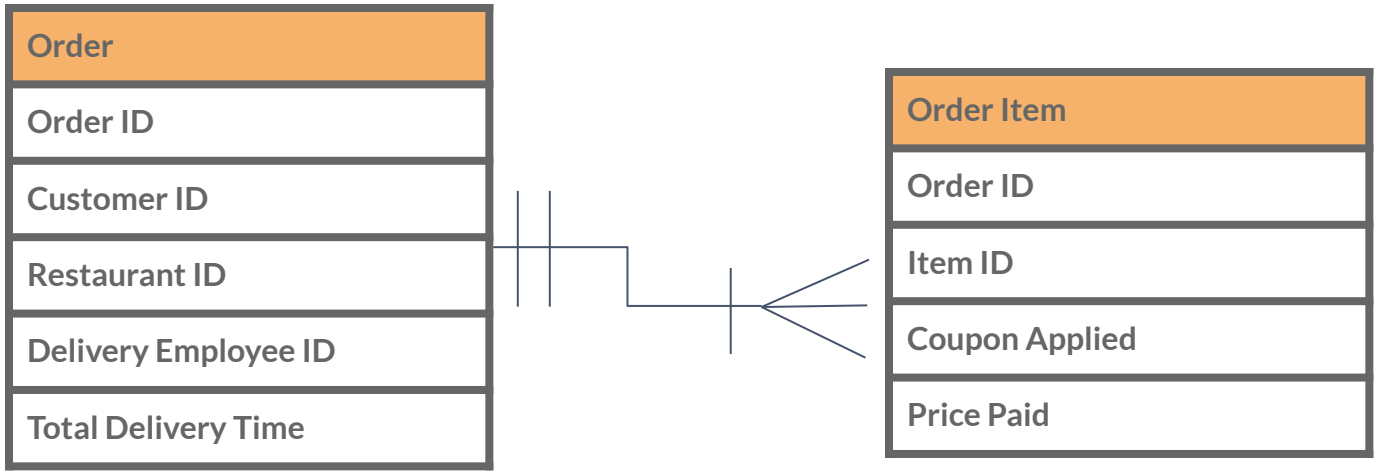


Item ID -> Type -> Type Details
Item ID -> Category -> Category Details

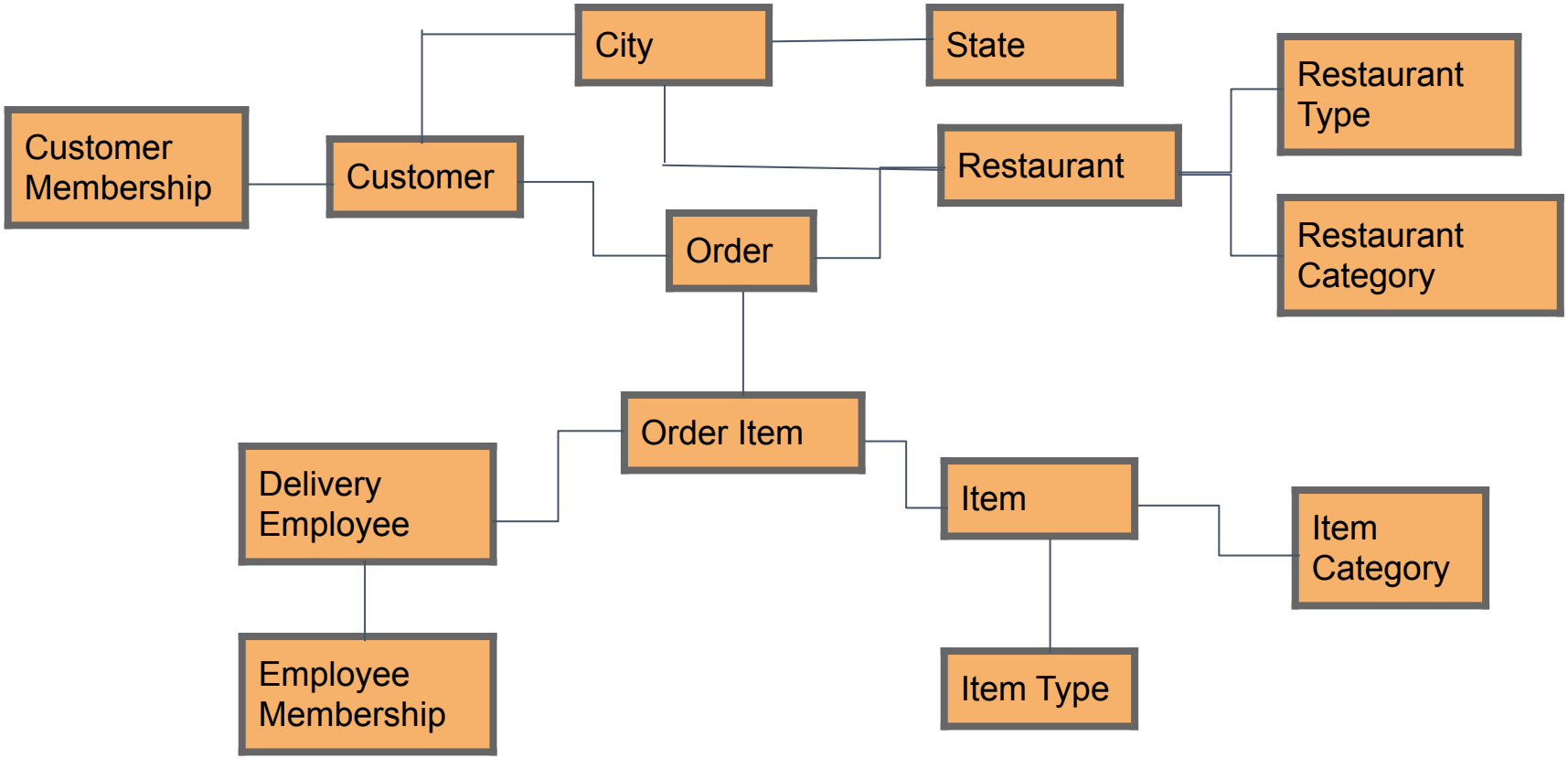
Transitive Dependencies in Item Entity



Transitive Dependencies in Order Entity



All Entities



Thank you