Object-Oriented Design

The Process For Designing An Object-Oriented Program

- An object-oriented program typically should model a real-world system. Thus, you should analyze the real-world system and then map it onto the object-oriented program.
- A class in an object-oriented program typically defines an object that corresponds with an object, or entity, in the real world.
- The process of object-oriented design.
    1. Identify the data attributes
    2. Subdivide each attribute into items smallest useful parts
    3. Identify the classes
    4. Identify the methods
    5. Refine the classes, attributes, and methods

Identifying The Data Attributes

- Depending on the nature of the system, you can identify data attributes in a variety of ways, including interviewing, analyzing existing systems, and evaluating comparable systems.

Subdividing The Data Attributes

- If a data attribute contains two or more parts, you should consider subdividing the attribute into multiple attributes. That way, you won't need to parse the attribute each time you use it.
- The extent to which you subdivide a data attribute depends on how it will be used. You should subdivide data attributes as much as possible so that your problem will be more flexible in case the requirement may be changed later.
- When you subdivide a data attribute, you can easily assemble it when necessary by concatenating the attributes.

Identifying The Classes

- After you identify and subdivide all of the data attributes for a program, you should group them by the classes with which they're associated.
- If a data attribute relates to more than one class, you can include it under all of the classes it relates to. Then, when you review class diagrams, or start to write the code for the classes, you may which attribute should be removed. Sometimes, this problem can be resolved by defining an abstract class.
- As you assign the attributes to classes, you should omit attributes that aren't needed and you should add any additional attributes that are needed to facilitate your class implementation later.

Identifying The Methods

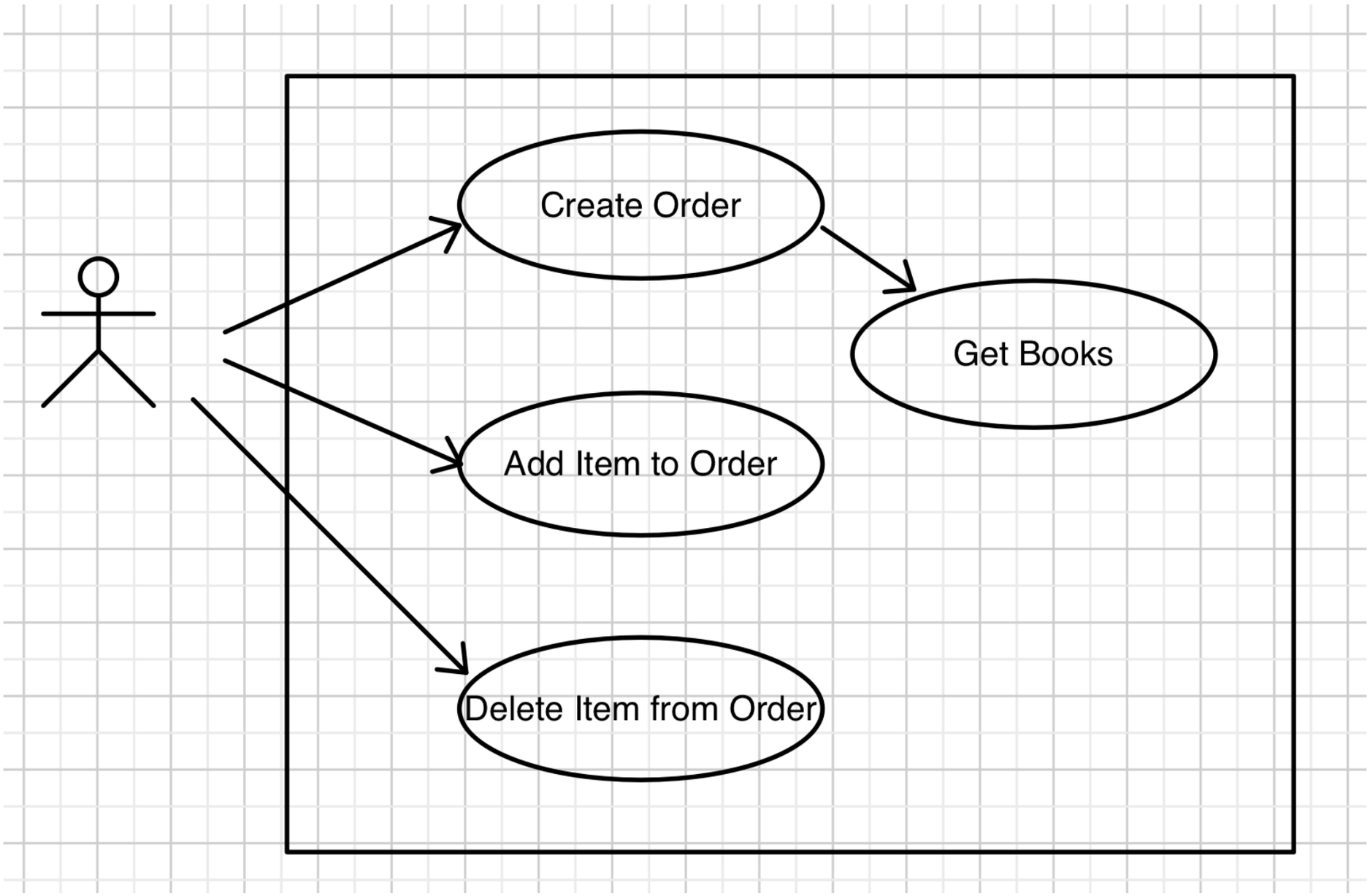
- After you identify the data attributes for an object, you can attempt to identify the methods that define the behavior of an object.
- If a data attribute such as “Order Total” can be calculated from other data attributes in the class, you can add a property or method that calculates the value for that attribute.
- As you work on this step, you can add methods that are necessary for the object to work correctly or convenience methods such as public properties, display and __str__ method that make it easier for other programmers to work with the object.
- Since methods perform actions, their names typically begin with a verb such as get, calculate, add, insert, delete, remove, and so on.
- To model the relationships between the classes in an object-oriented system, you can use UML (Unified Modeling Language) class diagrams.

The Three-Tier Architecture

- To simplify development and maintenance of large programs, many applications use a three-tier architecture to separate the application's user interface, business rules, and data processing. Each tier of the architecture may consist of one or more modules that contain classes or functions.
- The classes or functions in the presentation tier control the application's user interface. For a console application, the presentation tier typically uses a series of functions to provide console input and output. For a web application, the presentation tier typically consists of one class for each view for generating a web page,
- The classes or functions in the database tier handle all of the application's file or data processing.
- The classes in the business tier define the business objects and rules for the application. These classes act as an interface between the classes in the presentation and database tiers.

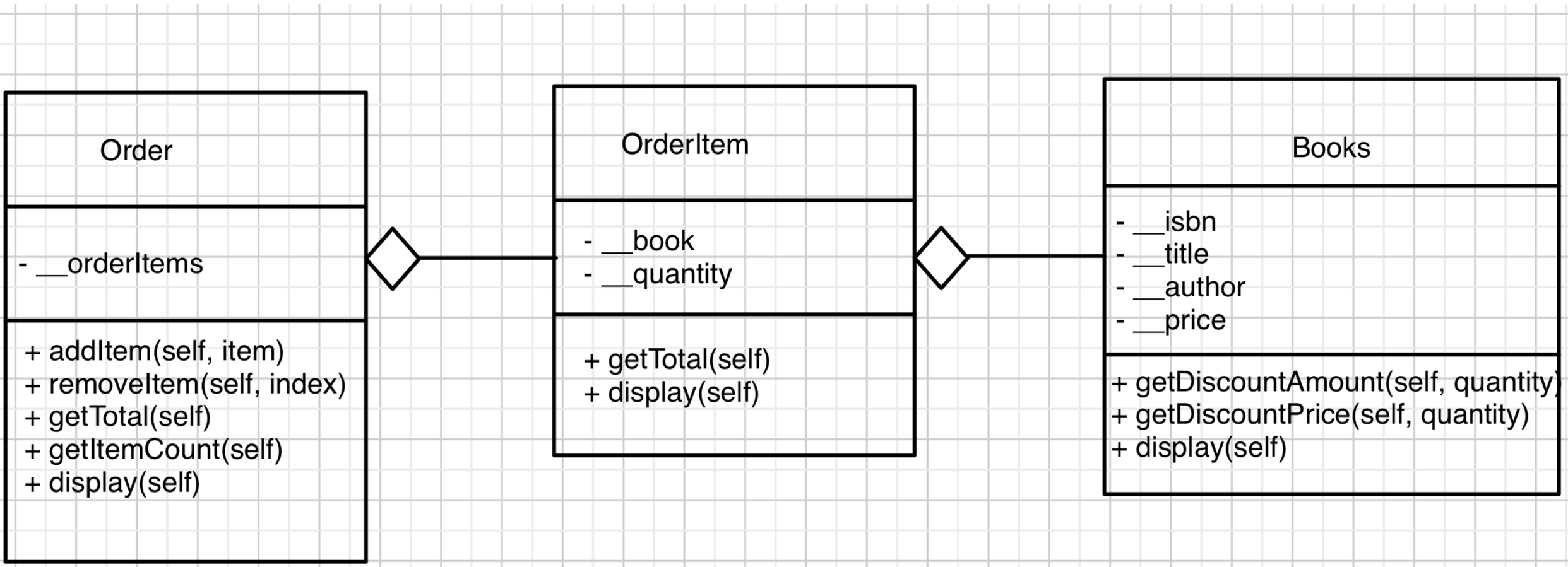
Use Cases - Order App

- This application allows the user to get a list of books, add books to an order, delete a book from the order, and see the order details. Each book has ISBN, title, author and price. And each has a list order items. And each order item contains a book and quantity. The book information is saved in a CSV file.



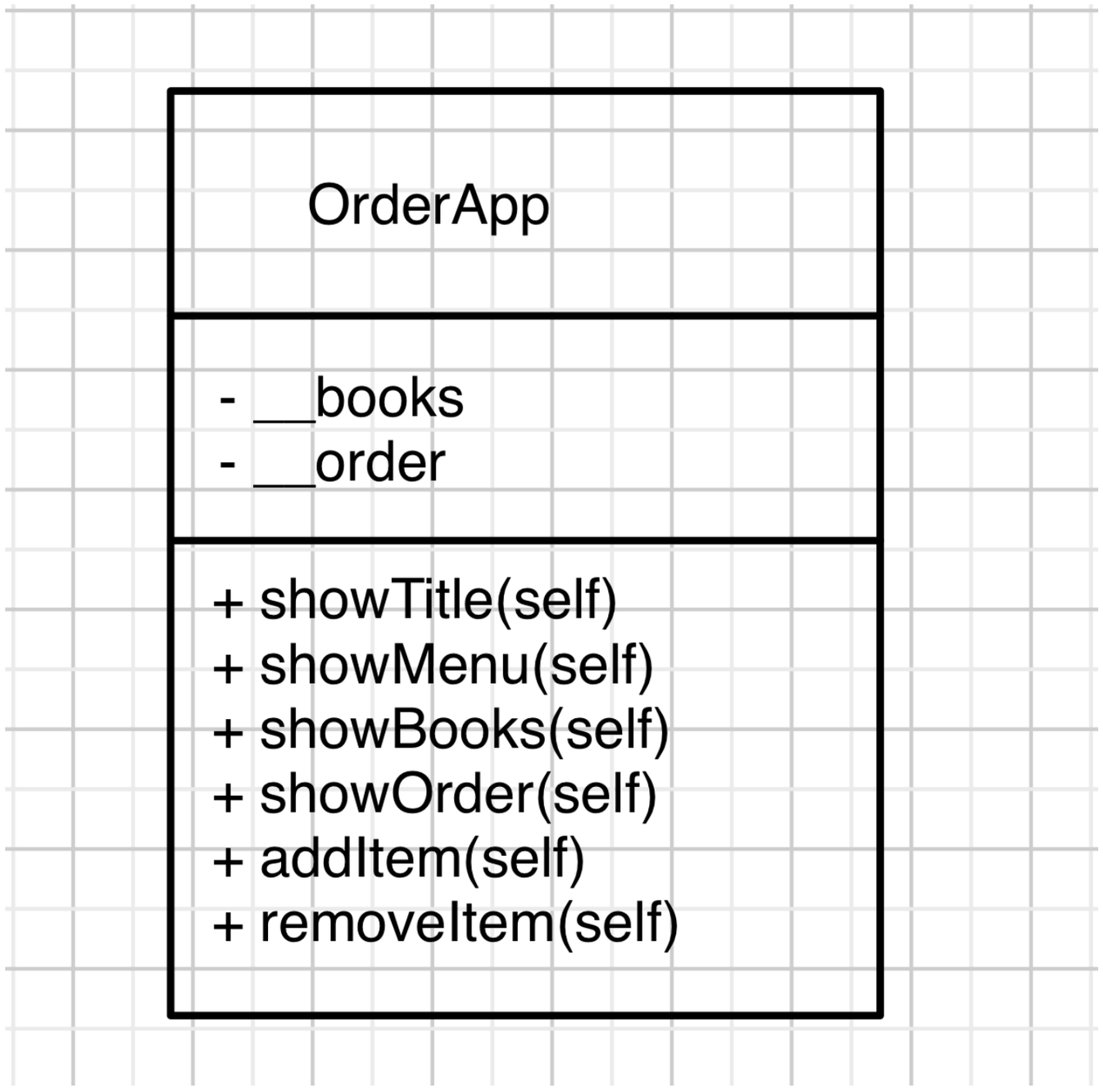
Class Diagrams - Order App

- These classes are defined in the business layer.



Class Diagrams - Order App

- The class is defined in the presentation layer.



The Three-Tier Architecture - Order App

- The orderapp has three tiers.