Data Structure and Algorithm

Laboratory Activity No. 1

Object-oriented Programming

|  |  |
| --- | --- |
| *Submitted by:* | *Instructor:* |
| Gabuyo, Ivan love D. | Engr. Maria Rizette H. Sayo |

07-26-2025

# Objectives

This laboratory activity aims to implement the principles and techniques in object-oriented programming specifically through:

* Identifying object-orientation design goals
* Identifying the relevance of design pattern to software development

# Methods

* Software Development
  + The design steps in object-oriented programming
  + Coding style and implementation using Python
  + Testing and Debugging
  + Reinforcement of below exercises
  1. Suppose you are on the design team for a new e-book reader. What are the primary classes and methods that the Python software for your reader will need? You should include an inheritance diagram for this code, but you do not need to write any actual code. Your software architecture should at least include ways for customers to buy new books, view their list of purchased books, and read their purchased books.
  2. Write a Python class, Polygons that has three instance variables of type str, int, and float, that respectively represent the name of the polygon, its number of sides, and its area. Your class must include a constructor method that initializes each variable to an appropriate value, and your class should include methods for setting the value of each type and retrieving the value of each type.

# Results

* 1. In this section, the diagram below shows the structures of the design for the e-book reader.

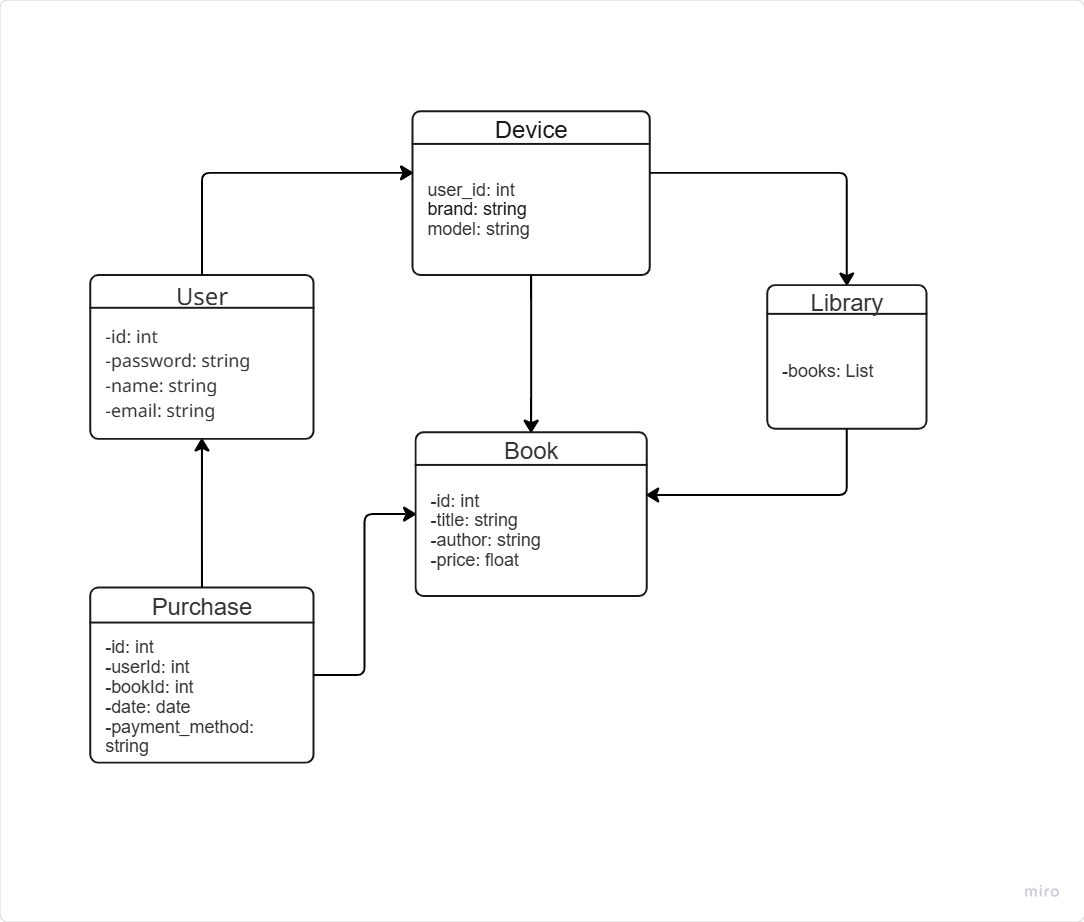


Figure 1 Diagram for e-book reader

The class diagram provides a structured overview of the eBook reader system, illustrating how its core components interact. The User**class** models individual users, enabling them to purchase and access books. The Book**class** encapsulates key details such as title, author, and price, serving as the foundation for the digital library. Transactions are managed by the Purchaseclass, which links users to books and records payment details, ensuring a seamless buying process.

The Libraryclass acts as a centralized repository, storing and organizing the books available to users. Meanwhile, the Shop(or eBook platform) functions as the primary interface, facilitating user interactions like browsing, purchasing, and reading.

By assigning each class a single, well-definedresponsibility, the system achieves modularity and maintainability. The relationships between classes such as a User making a Purchase of a Book clearly map the flow of data, from transaction initiation to content access. This design not only enhances organization but also simplifies future scalability, such as adding new features (e.g., subscriptions, reviews) without disrupting existing functionality.

* 1. The Python Class was created to represent the properties of a polygon. It includes the attributes for the polygon’s name, number of sides, and areas. The figures below show how the class is used to set and retrieve these values.

A computer screen shot of a program

AI-generated content may be incorrect.

Figure 2 Polygon Class

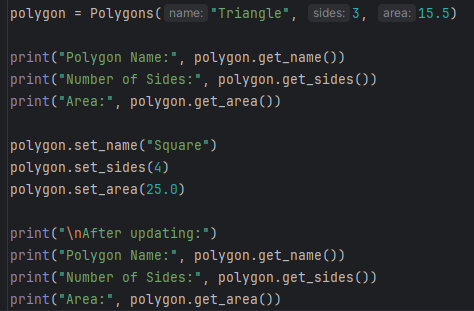


Figure 3 Example

In figure 3, the object polygon is created from the Polygons class with the values “Triangle” as the name, 3 as the number of sides, and 15.5 as the area. These values are passed to the constructor method \_\_in\_\_ and stored in the object.

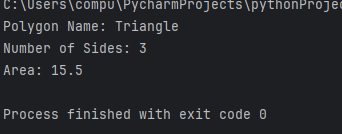


Figure 4 Result

In figure 4, the output indicated that the object was successfully created and the values were correctly stored and retrieved. It shows that the class is working as expected by allowing data to be accessed using a method.

# Conclusion

Through these exercises, we gained practical insights into software architecture, emphasizing how thoughtful design and clean coding practices contribute to scalable, maintainable programs. Future work could extend these systems with additional features, such as payment processing for the e-book reader or geometric calculations for the Polygons class, further solidifying these foundational skills.

**References**

[1] https://www.jetbrains.com/pycharm/download/?section=windows