UNIVERSITY OF CALOOCAN CITY
Caloocan, 1400 Metro Manila, Philippines

COLLEGE OF ENGINEERING
Computer Engineering
2nd Semester, School Year 2024-2025

Object-Oriented Programming

Laboratory Activity No. 1

Review of Technologies

Submitted by:
Monoy, Justin Rhey A.
1-18-2025/ BSCPE 1A

Submitted to
Engr. Maria Rizette H. Sayo
Instructor

Date Performed:
18-01-2025

Date Submitted
18-01-2025

I. Objectives

In this section, the goals in this laboratory are:

- To define the key terms in Object-oriented programming
- To be able to know the construction of OO concepts in relation to other types of programming such as procedural or functional programming

II. Methods

General Instruction:

A. Define and discuss the following Object-oriented programming concepts:

1. Classes

A blueprint for creating objects (a particular data structure), providing initial values for state (member variables or attributes), and implementations of behavior (member functions or methods).

Example: A class called "Person" may include fields like "name," "age," and "gender," as well as producers like "speak," "eat," and "sleep." Multiple person objects can be created from this class, each with its own unique values for the fields.

2. Objects

A component of a program that knows how to perform certain actions and how to interact with other elements of the program. Objects are the basic units of object-oriented programming. Objects can correspond to real-world objects or an abstract entity. When class is defined initially, the description is the only object that is defined.
Example: If we create an instance "myCar" based on the car class, it will have specific field values (e.g., color: blue, make: Toyota, model: Camry, year: 2022). This object can then invoke the producers defined in the class, such as "startEngine()" or "accelerate()".

3. Fields

Known as attributes or variables, are used to store data within an object or class. They represent the characteristics or properties associated with an object. Fields can have different data types, such as integers, strings, or custom objects.

4. Methods

According to Metcalf et al., 2023, method is a procedure associated with an object, defined within the object's type. It operates on the object's data and is invoked by sending a message to the object, encapsulating behavior and data together. Also, Methods are similar to function, but methods are also classified according to their purpose in the class design.

The following are the types of methods:

1. Instance Methods - to set or get details about instances (objects), and that is why they're known as instance methods. They are the most common type of methods used in a Python class.

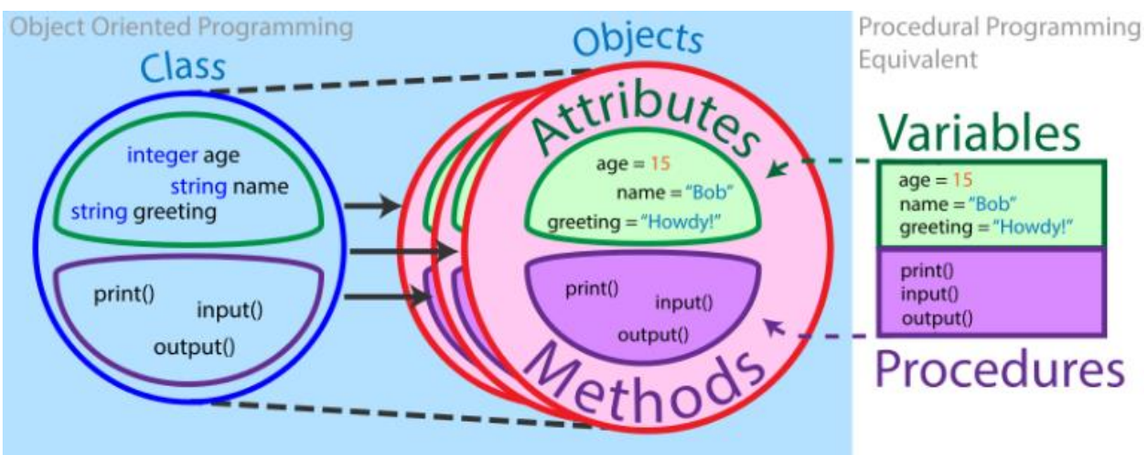
2. Class Methods – the purpose of this method is to set or get the details (status) of the class. To define the class method, the programmer should specify that it's a class method with the help of the @classmethod decorator. Class method have a default parameter - cls, which points to the class.

3. Static Methods – this method cannot access the class data. This means they do not need to access the class data. They are self-sufficient and can work on their own. Since they are not attached to any class attribute, they cannot get or set the instance state or class state.

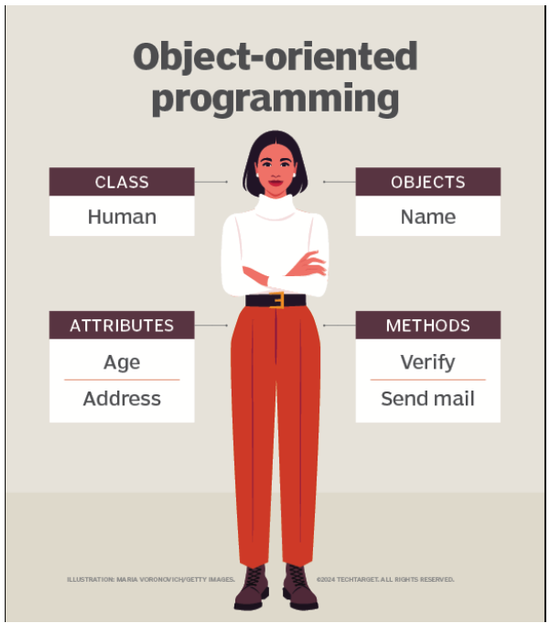
5. Properties – they are able to be looked at as functions, a property houses a function that can have its procedures or variables altered without directly going to the code and editing it. A property can be changed or updated based on user input which allows for a lot of user-interactive programs and applications.

III. Results

The image below shows the difference between classes and objects. Class shows the template or blueprint that defines attributes and methods. On the other hand, objects have their own data depending on what the programmer information is (e.g., age = 19, name = "Justin", and greeting = "welcome").



https://brilliant.org/wiki/object-oriented-programming



https://www.techtarget.com/searchapparchitecture/definition/object-oriented-programming-OOP

Object-oriented programming is like organizing ideas into categories. A class is a general blueprint, like "Human," that describes what the category includes. From this blueprint, you can create objects which are specific examples, such as a person with a particular name. Each object has attributes, which are details about it, like "Age" and "Address." It also has methods, which are actions it can perform, like "Verify" or "Send mail." This way of organizing makes it easier to understand and work with complex ideas in programming.

IV. Conclusion

Object-oriented programming also needs methods to work properly; if any of these methods are not present, the program has a possibility of having a problem or error. In order to develop something like software, the programmer should know the structure or the building blocks of object-oriented programming because it is very important.

Understanding the structure in object-oriented programming (OOP) is crucial as it enhances modularity, reusability, and maintainability of software. This knowledge minimizes errors and improves software quality, ultimately leading to more efficient development practices and better software products (Onu et al., 2024).

Reference

Book

[1] F. U. Onu, I. C. Oliver, Mamah Chukwurah Hyginus, Afolabi Idris Yinka, Nwodo John O, and Y. Sunday, “Demystifying The Object-Oriented Features Of Popular Object-Oriented Programming Languages.,” IOSR Journal of Computer Engineering, vol. 26, no. 5, pp. 26–32, Oct. 2024, doi: https://doi.org/10.9790/0661-2605012632.

[2] M. Metcalf, J. Reid, M. Cohen, and R. Bader, “Object-oriented programming,” Oxford University Press eBooks, pp. 294–324, Dec. 2023, doi: https://doi.org/10.1093/oso/9780198876571.003.0015.

Website

[1] A. Goodarzi, “Object oriented programming - part 2: Field, producer, class and object,” LinkedIn, https://www.linkedin.com/pulse/object-oriented-programming-part-2-field-producer-class-goodarzi-y0rif (accessed Jan. 18, 2025).

[2] A. S. Gillis and S. Lewis, “What is object-oriented programming (OOP)?: Definition from TechTarget,” Search App Architecture, https://www.techtarget.com/searchapparchitecture/definition/object-oriented-programming-OOP (accessed Jan. 18, 2025

[3] H. Singh, “Methods in python - A key concept of Object Oriented Programming,” Analytics Vidhya, https://www.analyticsvidhya.com/blog/2020/11/basic-concepts-object-oriented-programming-types-methods-python/ (accessed Jan. 18, 2025).

[4] J. P. Soren, “Method and attribute in OOP,” DEV Community, https://dev.to/jps27cse/method-and-attribute-in-oop-3kl (accessed Jan. 18, 2025).

[5] K. Moore, E. Njeru, and M. Pocevicius, “Classes (OOP) | Brilliant Math & Science Wiki,” Brilliant.org, 2016. https://brilliant.org/wiki/classes-oop/

[6] “Object-oriented programming: Brilliant math & science wiki,” Brilliant, https://brilliant.org/wiki/object-oriented-programming (accessed Jan. 18, 2025).

[7] “Object oriented programming/properties,” Wikibooks, open books for an open world, https://en.wikibooks.org/wiki/Object_Oriented_Programming/Properties (accessed Jan. 18, 2025).