What is object-oriented programming?

A computer programming model is a style of programming that emphasizes certain procedures while programming. One of the most influential and popular programming paradigms is Object-Oriented Programming. OOP is a model that privileges software design around data, or objects, instead of functions and logic. An object is defined as a construct that has unique attributes and behavior and these are different from other objects' properties.

The main focus of OOP is the objects that developers manipulate. The logic sequence of thinking is not as important as the object itself. Since the developers manipulate objects as we manipulate real-world objects, this paradigm is well-suited for programs that are large, complex and actively updated or maintained.

What is an object?

Objects can be classified depending of the needs of the program, we can establish different developing teams to be in charge of different types of objects. The organization of an program written in object-oriented language makes the method beneficial to collaborative development, where projects are divided into groups as stated before. Code reusability, scalability and efficiency are other advantages of OPP.

An object is a representation of something that we want to manipulate and often we can relate it to physical entities, such as a human being who is described by properties like name and address. Once an object is known, it is labeled with a class of objects that defines the kind of data it contains and any logic sequences that can manipulate it. Each distinct logic sequence is known as a method. Objects can communicate with well-defined interfaces called messages.

Structure of object-oriented programming

- Classes are user-defined data types that act as the blueprint for individual objects, attributes and methods.
- Objects are instances of a class created with specifically defined data. 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.
- Methods are functions that are defined inside a class that describe the behaviors of an object. Each method contained in class definitions starts with a reference to an instance object. Additionally, the subroutines contained in an object are called instance methods.
- Attributes are defined in the class template and represent the state of an object. Objects will have data stored in the attributes field. Class attributes belong to the class itself.

Principles of OPP

- Encapsulation. All important information is contained inside an object and only select information is exposed. The implementation and state of each object are privately held inside a defined class. And this information is inaccessible to other objects nor they are allowed to make changes. They are only able to call a list of public functions or methods. This characteristic of data hiding provides greater program security and avoids unintended data corruption.

- Abstraction. Objects only reveal internal mechanisms that are relevant for the use of other objects, hiding any unnecessary implementation code. The derived class can have its functionality extended. This concept can help developers more easily make additional changes or additions over time.

- Inheritance. Classes can reuse code from other classes. Relationships and subclasses between objects can be assigned, enabling developers to reuse common logic while still maintaining a unique hierarchy. This property of OOP forces a more thorough data analysis, reduces development time and ensures a higher level of accuracy.

- Polymorphism. Objects are designed to share behaviors and they can take on more than one form. The program will determine which meaning or usage is necessary for each execution of that object from a parent class, reducing the need to duplicate code. A child class is then created, which extends the functionality of the parent class. Polymorphism allows different types of objects to pass through the same interface.

Examples of Object-oriented programming languages

There are many programming languages that could be classified as OPP. Nevertheless, some programming languages pair with OOP better than others. For example, programming languages considered pure OOP languages treat everything as objects. 359898Other programming languages are designed primarily for OOP, but with some procedural processes included.

Pure OOP languages include Ruby, Scala, JADE, Emerald. Programming languages designed primarily for OOP include Java, Python and C++, and paired programming languages Visual Basic .NET, PHP and JavaScript.