

The Characteristics of an Object Orientated Paradigm

Object oriented programming

According to Vangie Beal (No Date) Object-oriented programming (OOP) refers to a paradigm of programming organized around objects rather than "actions" and data rather than logic. Object-oriented programming is based on the three concepts encapsulation, inheritance, and polymorphism.

Encapsulation

A fundamental concept in OOP refers to the wrapping up of data (Variables and functions) under a single unit (Tutorialspoint No Date). Thus creating self contained modules that bind processing functions to data. It ensures code modularity, which keeps routines separate and less prone to conflict with each other. This concept is often used to hide the internal representation, or state, of an object from the outside (Private, Public, Protected).

Class | Objects | Sub Object | Containers

Class can be defined as a blueprint pertaining to an object (Geeksforgeeks No Date), classes usually contain methods and variables of an object (pre-defined properties). Thus, it can be defined as a specific instance of a class (Wikipedia 2018). Whilst a Sub object is an object that sits inside another object in the same category (Wikipedia 2018). A container uses a dynamic structure and is class, data structure or abstract data type, whose instances are collections of other objects (Careerride No Date). In other words a container stores objects in an organized way that follows a specific set of rules pertaining to access.

Types Of Classes

Base & Derived Classes

Base is the parent class of a derived class, a class that facilitates the creation of other classes that can reuse the code implicitly inherited from the base class with the exception of constructors and destructors (Techopedia No Date).

Derived is a class created or derived from another existing class, conceived through the process of inheritance, which entails passing and using existing properties of that class/-object (Techopedia No Date).

Abstract & Concrete Classes

Abstract class is a template definition of methods and variables of a class that contains one or more abstracted methods. Similarly to derived classes, abstract essentially inherit properties of other classes, hence why abstracts methods contain no implementation (Quora No Date).

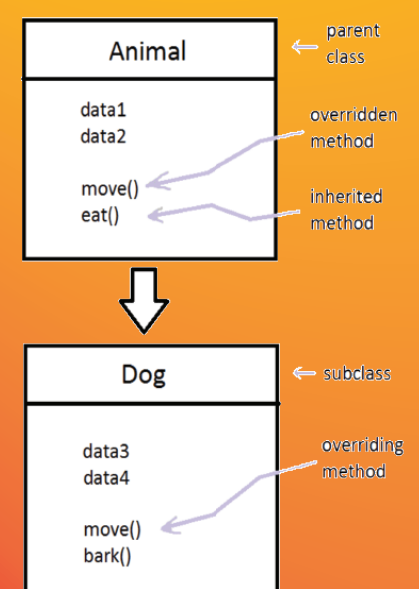
Concrete is a class that has an implementation for all of its methods that were inherited from abstract or implemented via interfaces (Stackoverflow 2017).

Object Oriented Interfaces

According to Akshay Raje (2016) an interface in OOP is essentially a programming structure (syntax) and paradigm or way to interact with methods, stating the enforcement of certain properties on an object (class) that includes that interface and what it could respond to. For instance a "box" is an object. For a given "box", interface declares the different "inputs" and corresponding "outputs" of that "box". Interface is a concept of abstraction and encapsulation.

Method Redefinition, Overriding & Overloading

As noted by Argha S (No Date), Method overloading in OPP is when there are two methods of the same class with the same name, but different parameter types. Naimesh (2006) insinuates that, Method overriding (also known as redefinition) is a procedure that allows a (child) subclass to provide specific implementation of a method that is provided by one of its super (parent) classes. In other words, is when a child class redefines a method inherited from its parent class (Wikipedia 2018).



Polymorphism

Refers to an object's ability to take multiple forms (Techopedia No Date), in addition to a programming language's ability to process objects differently depending on their data type or class (Webopedia No Date). More specifically, it is the ability to redefine methods for derived classes.

An instance of a derived class (creature) and base class (persona):

```
class persona {
public:
    void setname(string i) { name = i; };
    string getname() { return name; };

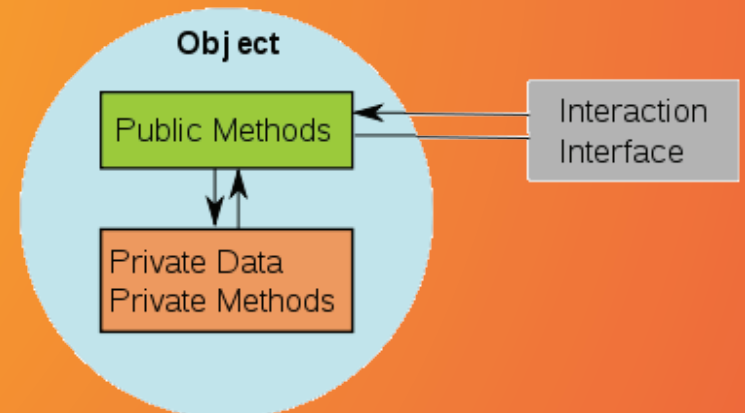
private:
    string name = "";
};

class creature : public stats, public persona {
public:
    int getWorldx() { return worldx; };

    int getWorldy() { return worldy; };

    int getDMG() { return dmgtaken; };

    bool isAlive() {
        bool alive = true;
        if (dmgtaken == maxHP) {
            alive = false;
        }
        return alive;
    }
};
```



OOP Interfaces (Wikimedia 2011)

Generics & Templates

Relates to writing "templates" of source codes that the compiler in the process of compilation transforms into source codes, rather than source code that is compiled as is (Stackoverflow 2010). An example of generic programming can be seen through container classes like arrays, lists or maps that contain a collection of other objects.

A template is the foundation of generic programming, essentially a blue print for creating a generic class or a function with code that can be modified to fit a specific situation (Tutorialspoint No Date).

Bibliography

- Akshay Raje (2016). How is the term "interface" defined in object-oriented programming (OOP)? Available at: <https://www.quora.com/How-is-the-term-interface-defined-in-object-oriented-programming-OOP> [Accessed: 02/12/2018]
- Argha S (No Date). What is method "Overloading" in object-oriented programming (OOP) ?. Available at: <https://www.mytutor.co.uk/answers/3994/A-Level/Computing/What-is-method-Overloading-in-object-oriented-programming-OOP/> [Accessed: 01/12/2018]
- Careerride (No Date). C++ Container class - Explain container class and its types in C++. Available at: <https://www.careerride.com/C++-container-class.aspx> [Accessed: 01/12/2018]
- Geeksforgeeks (No Date). Classes and Objects in Java. Available at: <https://www.geeksforgeeks.org/classes-objects-java/> [Accessed: 30/11/2018]
- Naimesh Patel (2008). ABAP Objects: Overriding (Redefinition). Available at: <http://zevolving.com/2008/09/abap-objects-overriding-redefinition/> [Accessed: 02/12/2018]
- Techopedia (No Date). Polymorphism. Available at: <https://www.techopedia.com/definition/28106/polymorphism-general-programming> [Accessed: 29/11/2018]
- Techopedia (No Date). Base Class. Available at: <https://www.techopedia.com/definition/26896/base-class> [Accessed: 30/11/2018]
- Techopedia (No Date). Derived Class. Available at: <https://www.techopedia.com/definition/3780/derived-class> [Accessed: 30/11/2018]
- Tutorialspoint (No Date). Java - Encapsulation. Available at: https://www.tutorialspoint.com/java/java_encapsulation.htm [Accessed: 30/11/2018]
- Tutorialspoint (No Date). C++ Templates. Available at: https://www.tutorialspoint.com/cplusplus/cpp_templates.htm [Accessed: 01/12/2018]
- Webopedia (No Date). OOP - Object Oriented Programming. Available at: https://www.webopedia.com/TERM/O/object_oriented_programming_OOP.html [Accessed: 29/11/2018]
- Webopedia (No Date). Polymorphism. Available at: <https://www.webopedia.com/TERM/P/polymorphism.html> [Accessed: 29/11/2018]
- Wikipedia (2018). Subobject. Available at: <https://en.wikipedia.org/wiki/Subobject> [Accessed: 30/11/2018]
- Wikipedia (2018). Object (computer science). Available at: [https://en.wikipedia.org/wiki/Object_\(computer_science\)](https://en.wikipedia.org/wiki/Object_(computer_science)) [Accessed: 30/11/2018]
- Wikipedia (2018). Method overriding. Available at: https://en.wikipedia.org/wiki/Method_overriding [Accessed: 1/12/2018]
- Wikimedia (2011). File:CPT-OOP-interfaces.svg. Available at: <https://commons.wikimedia.org/wiki/File:CPT-OOP-interfaces.svg> [Accessed: 02/12/2018]
- Stackoverflow (2010). What is the meaning of "generic programming" in c++. Available at: <https://stackoverflow.com/questions/3862378/what-is-the-meaning-of-generic-programming-in-c> [Accessed: 01/12/2018]
- Stackoverflow (2017). What is the Concrete class in java. Available at: <https://stackoverflow.com/questions/43224901/what-is-the-concrete-class-in-java> [Accessed: 01/12/2018]
- Quora (No Date). What is difference between concrete classes and abstract classes?. Available at: <https://www.quora.com/What-is-difference-between-concrete-classes-and-abstract-classes> [Accessed: 01/12/2018]