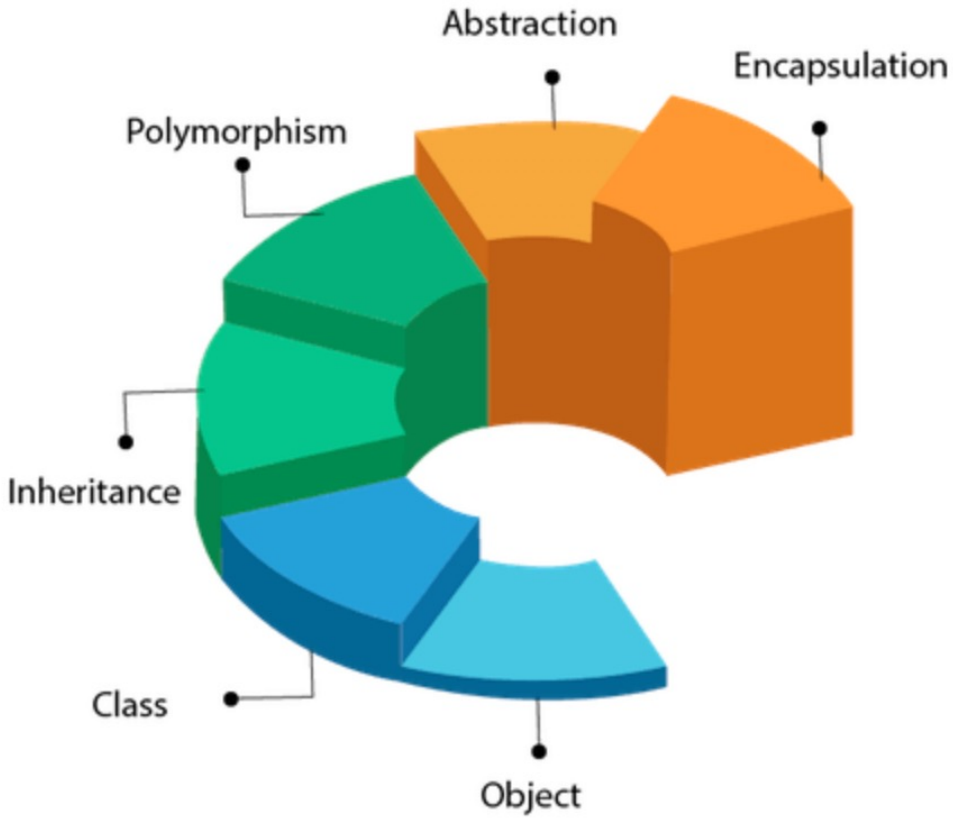
Object Oriented Programming

2022/11/15

Basic concepts

- Object
- Class
- Inheritance
- Polymorphism
- Abstraction
- Encapsulation



Object

- Any entity that has state and behaviour is known as an object. For example, a chair, pen, table, keyboard, bike, etc. It can be physical or logical.
- An Object can be defined as an instance of a class. An object contains an address and takes up some space in memory. Objects can communicate without knowing the details of each other's data or code. The only necessary thing is the type of message accepted and the type of response returned by the objects.

Class

- It is a logical entity.
- A class can also be defined as a blueprint from which you can create an individual object. Class doesn't consume any space.

Inheritance

- When one object acquires all the properties and behaviours of a parent object, it is known as inheritance. It provides code reusability. It is used to achieve runtime polymorphism.

```
//Inheritance
class Animal {

}

class Dog extends Animal {

}
```

Polymorphism

- If one task is performed in different ways, it is known as polymorphism. For example: to convince the customer differently, to draw something, for example, shape, triangle, rectangle, etc.
- In Java, we use method overloading and method overriding to achieve polymorphism.
- Another example can be to speak something; for example, a cat speaks meow, dog barks woof, etc.

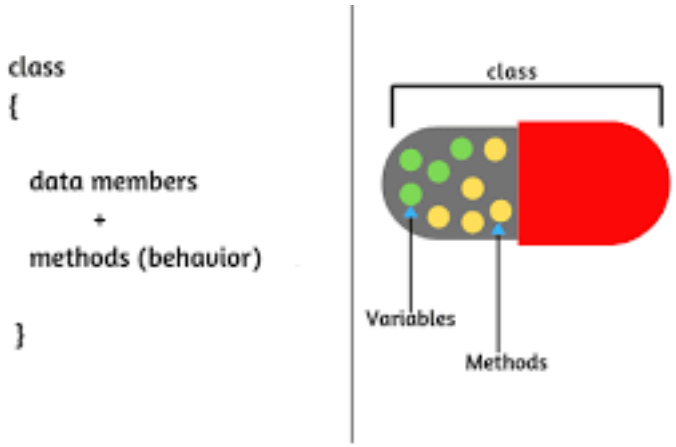
```java
//Polymorphism
class Person {
 void walk() {
  System.out.println("Can Run….");
 }
}
class Employee extends Person {
 void walk() {
  System.out.println("Running Fast…");
 }
 public static void main(String arg[]) {
  Person p = new Employee(); //upcasting
  p.walk();
 }
}
```

Abstraction

- Hiding internal details and showing functionality is known as abstraction. For example phone call, we don't know the internal processing.
- In Java, we use abstract class and interface to achieve abstraction.

Encapsulation

- Binding (or wrapping) code and data together into a single unit are known as encapsulation. For example, a capsule, it is wrapped with different medicines.
- A java class is the example of encapsulation. Java bean is the fully encapsulated class because all the data members are private here.




```java
//Encapsulation
public class Student {
 private String name;

 public String getName() {
  return name;
 }

 public void setName(String name) {
  this.name = name
 }

}

class Test {

 public static void main(String[] args) {
  Student s = new Student();
  s.setName("Isuru");
  System.out.println(s.getName());
 }

}
```

Thank you!