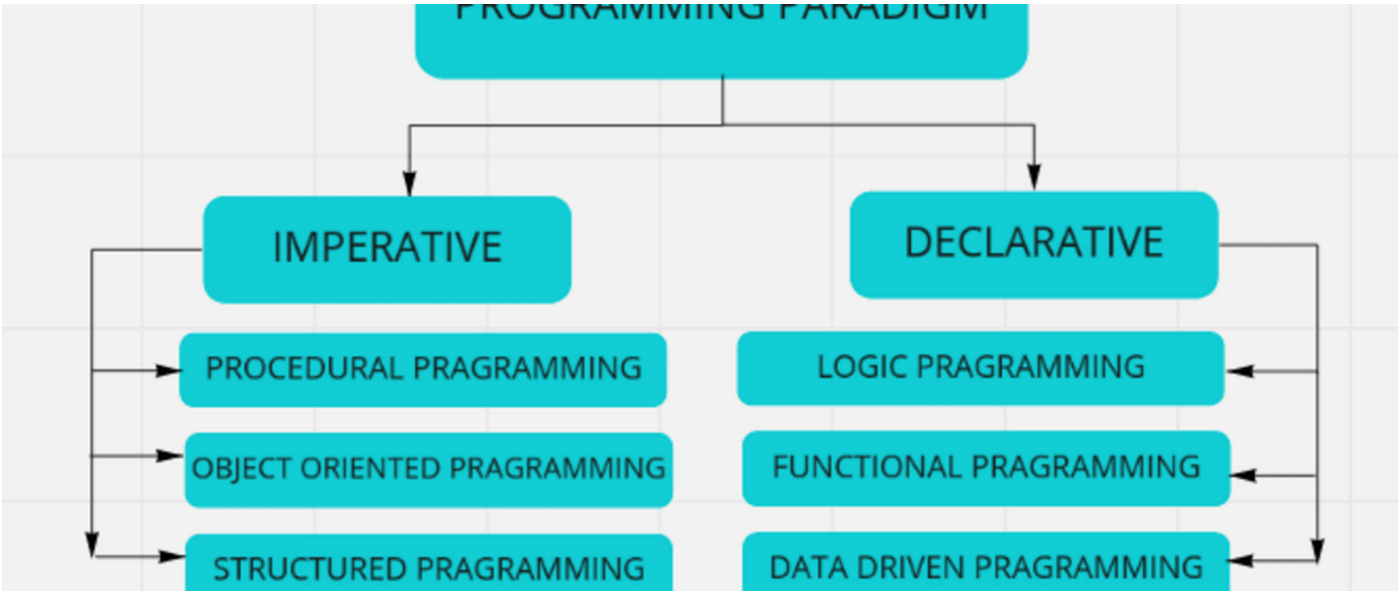
# OOPs1 - Introduction To OOPs

23 May 2024 20:58

Agenda :
1. Programming Paradigm
2. Object Oriented Programming
3. Abstraction

## 1. Programming Paradigm




**Procedural Programming** : We organise our code into a bunch of procedures. Each procedure may call other procedures internally.
Example : **C.**

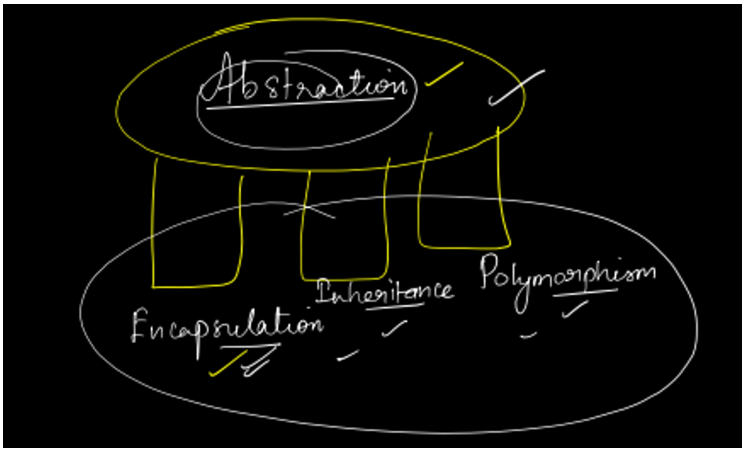
Cons of Procedural Programming
- Difficult to debug.
- Messy Code.
- Difficult to implement the complex system.

## 1. Object Oriented Programming

Object-oriented programming (OOP) is a **computer programming model that organizes software design around data, or objects, rather than** functions and logic.

**Principle Of Oops**
- Abstraction

**3 Pillars of Oops :**
- Inheritance
- Polymorphism
- Encapsulation




## 3. Abstraction

- Purpose is to represent a complex system which has various attributes which has some associated behaviour.
- Abstraction in Java refers to hiding the implementation details of a code and exposing only the necessary information to the user.
- It provides the ability to simplify complex systems by ignoring irrelevant details and reducing complexity.

### Encapsulation

**Encapsulation** is defined as the wrapping up of data under a single unit. It is the mechanism that binds together code and the data it manipulates. Another way to think about encapsulation is, that it is a protective shield that prevents the data from being accessed by the code outside this shield.

```
// Java Program to demonstrate
// Java Encapsulation

// fields to calculate area
class Area {
    int length;
    int breadth;

    // constructor to initialize values
    Area(int length, int breadth)
    {
        this.length = length;
        this.breadth = breadth;
    }

    // method to calculate area
    public void getArea()
    {
        int area = length * breadth;
        System.out.println("Area: " + area);
    }
}

class Main {
    public static void main(String[] args)
    {

        Area rectangle = new Area(2, 16);
        rectangle.getArea();
    }
}
```