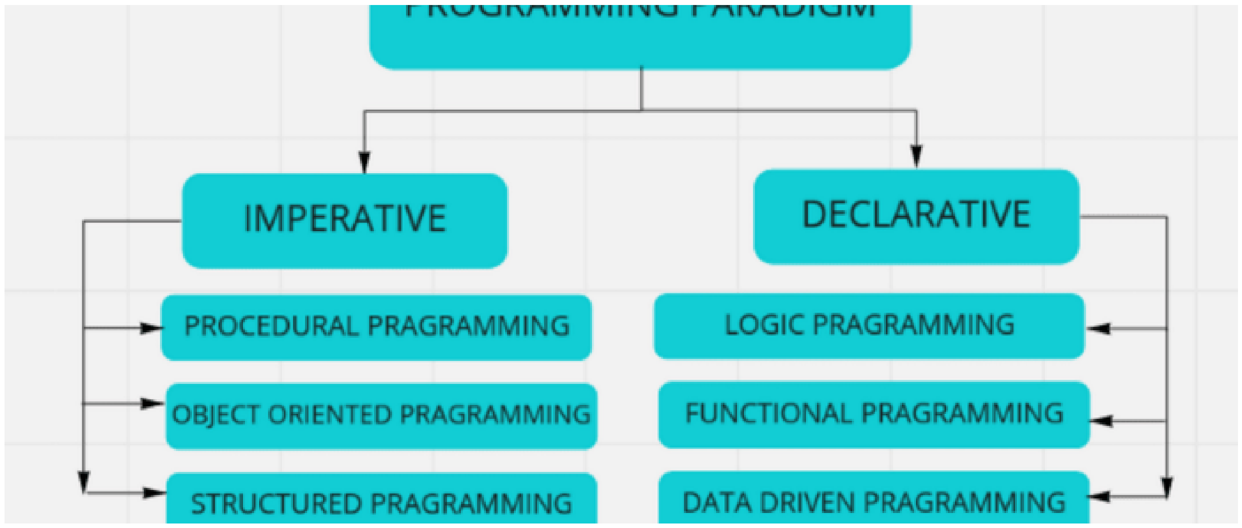
OOPs1 - Introduction To OOPs

1. Programming Paradigm



Procedural Programming :

- We organize 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.

2. 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.

Class

- The building block of C++ that leads to Object-Oriented programming is a Class.
- It is a user-defined data type, which holds its own data members and member functions, which can be accessed and used by creating an instance of that class.
- Data members are the data variables and member functions are the functions used to manipulate these variables together these data members and member functions define the properties and behavior of the objects in a Class.

Object

- An Object is an identifiable entity with some characteristics and behavior.
- An Object is an instance of a Class. (Pointer instance and refere)
- When a class is defined, no memory is allocated but when it is instantiated (i.e. an object is created) memory is allocated.

Ways To Create Object:

```
#include <iostream>
using namespace std;
// person is a class
class person {
    char name[20];
    int id;
public:
    void getdetails() {}
};

int main()
{
   1. person p1;                              // p1 is a stack allocation
   2. Person* p1 = new person();              // pointer reference heap
   3. std::unique_ptr<MyClass> myObjectPtr =
      std::make_unique<MyClass>(42,"Hello, World!"); // Using unique_ptr
    return 0;
}
```

3. How OOPs Achieved ?

- OOPs is achieved using Abstraction, which runs using three pillars Inheritance Polymorphism and Encapsulation.

Principle Of Oops

  - Abstraction

3 Pillars of Oops :

  - Inheritance
  - Polymorphism
  - Encapsulation

Abstraction

- Purpose is to represent a complex system which has various attributes which have some associated behavior.
- 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.
- Abstraction In java can be achieved in C++ using 2 Ways:
  - Abstraction Using Classes {Following OOPs Pillars}
  - Using Header Files