Unit 5 : Object Oriented Programming

Syllabus :

Programming Paradigms :

- ✓ Monolithic
- ✓ Procedural
- ✓ Structured
- ✓ Object oriented

Features of Object oriented programming :

- ✓ Classes, Objects, Methods and Message passing
- ✓ Inheritance, Polymorphism, Containership
- ✓ Reusability, Delegation, Data abstraction and encapsulation.

Unit 5 : Object Oriented Programming

Syllabus :

Classes and Objects :

- ✓ Classes and Objects
- ✓ Class method and self object
- ✓ Class variables and Object variables
- ✓ Public and private members
- ✓ Class methods

Introduction to Programming Paradigm

- Programming can be expressed in various ways.
- Over the years, people have improved programming in various ways.
- Paradigm means “type” or “ideal example”.
- Majorly there are four types of programming.
    1. **Monolithic programming**
    2. **Procedural programming**
    3. **Structured programming (Paradigm)**
    4. **Object Oriented Programming**

Introduction to Programming Paradigm

1. Monolithic programming

- **Here complete program is written as a sequence.**
- **There are no modules or functions used.**

Advantage :

It is simplest way of programming.

Disadvantage

Its problem is there can be lot of repetition of code when same operation has to be done many times.

Introduction to Programming Paradigm

2. Procedural programming

- Here program or main function is divided in procedures or functions.
- Function is a small unit of programming logic.Thus main task is composed of several procedures and functions

Advantages :

- Same function can be used multiple times in one program whenever the operation is needed.
- It reduces redundancy(repetition of syntax to reduce errors) in program.
- It makes program readable.

Introduction to Programming Paradigm

Disadvantages

- **Code size is increased with this approach.**
- **Data can be modified by any procedure.**
- **Data is not protected properly.**
- **This approach makes program un-structured and poor in readability.**

Introduction to Programming Paradigm

3. Structured programming (Paradigm)

- **Here program is written in a structured format compulsorily.**
- **It's making extensive use of the structured control flow constructs of selection (if/then/else) and repetition (while and for), block structures, and subroutines.**
- **Now most of the procedural programming languages support structured programming paradigm.**

Advantages

- **It is readable program.**
- **It is easy to find errors or especially logical errors.**

Disadvantage

- **Data can be modified by any procedure or part of the program.**

Introduction to Programming Paradigm

4. Object Oriented Programming (OOP)

- Here data is major focus.
- All program involve creation of class and objects
- Data and functions are put together to avoid misuse or un-intentional modifications in data.
- Here “Class” puts together data and functions.
- Only functions belonging to that class can modify the data.
- Also OOP supports features like inheritance, abstraction, etc.

Introduction to Programming Paradigm

Advantages

- **OOP features enable re-use of programs very easy.**
- **Data is protected here.**

Disadvantage

- **Dependence of a class on other class may lead to inefficient programs.**

Features of Object Oriented Programming

1. Class

- A class is a blue print to create objects or instances.
- It is model which is used to generate exact same objects or instances.
- In a class variable and functions are bind together.
- Thus any object of a class is having its variables and its methods or functions together.

Advantages :

- Class allows creating user defined data structure.
- Putting variables and functions together make data protection easy.
- It also helps in simulating real world scenario.

Features of Object Oriented Programming

```
# Sample Class
class Sample:
    """ This is a Sample class """
    def instance_method(self):# method with first argument 'self' is instance method
        print("In instance method")
    def class_method():       # method without self parameter is class method
        print("In class method")


s = Sample()
s.instance_method()          #instance method is accessed using object
Sample.class_method()        #class method is accessed using class name
```

Output :

```
In instance method
In class method
```

Features of Object Oriented Programming

2. Objects

- Object is a instance of a class.
- It contains all variables and methods together.
- For example, class is like plan of a building and object is real building.
- One plan can be used to generate multiple buildings.
- In following code three objects of Sample class are created with different values.

Features of Object Oriented Programming

2. Objects

```
# instantiation :: creating instance or instances of class
object_s_1 = Sample(10) # Object of class Sample
object_s_2 = Sample(20) # Object of class Sample
object_s_3 = Sample(30) # Object of class Sample
```