DATA SCIENCE COURSE TUTORIAL # 34

3.21 Introduction to Object Oriented Programming (OOP)

What is OOP

Object Oriented Programming (OOP) is a programming style that focuses on creating classes and objects. It helps organize code in a structured way that is easy to understand, reuse, and maintain. OOP is used in Python to model real world entities using objects that have properties and behaviors.

What is a Class

A class is a blueprint used to create objects. It defines how an object will behave and what data it will store. A class contains attributes that hold data and methods that define actions.

Example:

```
class Student:
    pass
```

This example creates an empty class named Student.

What is an Object

An object is an instance of a class. When you create an object, Python allocates memory for data defined in the class.

Example:

```
s1 = Student()
s2 = Student()
```

Here, s1 and s2 are two objects created from the Student class.

Creating a Class with Attributes and Methods

A class can contain data attributes and methods to perform actions.

Example:

```
class Student:
    name = "Ali"
    age = 20
    def show(self):
        print("Name:", self.name)
        print("Age:", self.age)
```

Creating Objects from a Class

Objects are created by calling the class name like a function.

Example:

```
s1 = Student()
s1.show()
```

Output:

```
Name: Ali
Age: 20
```

The init Function

The **init** function is a special method in a class. It runs automatically whenever an object is created. It is used to initialize object attributes.

Example:

```
class Student:
    def __init__(self, name, age):
        self.name = name
        self.age = age

    def show(self):
        print("Name:", self.name)
        print("Age:", self.age)
```

Creating Objects with init Function

Now objects must be created with arguments that match the **init** parameters.

Example:

```python
s1 = Student("Ali", 22)
s2 = Student("Aisha", 20)

s1.show()
s2.show()
```

Output:

```
Name: Ali
Age: 22
Name: Aisha
Age: 20
```

Why Use init Function

The **init** method helps in the following.

- Automatically sets values when object is created.
- Makes code cleaner and organized.
- Allows each object to have different data.

Summary of Basic OOP Concepts

- A class is a blueprint used to create objects.
- An object is an instance of a class.
- Methods are functions inside a class.
- The **init** function initializes object attributes.
- Each object can have different values for attributes initialized inside **init**.

Common Use Cases of Classes and Objects

- Creating models for students, employees, cars or products.
- Building structured programs.
- Managing large codebases in data science applications.
- Preparing reusable components for projects.