



Lecture 08

Nested Class

Mr. Mubashir Ali

Lecturer (Dept. of Software Engineering)

Office Address: 41-C, 2nd Floor

mubashirali@lgu.edu.pk

Lahore Garrison University, Lahore

Previous Lectures

- **Types of Methods**
- **Instance Methods**
 - **Accessor Method**
 - **Mutator Method**
- **Class Method**
- **Static Method**

Outline

- **Passing class members between classes**
- **Nested Class**

```

1  #Passing member of one Class to another Class.
2  class Student:
3      def __init__(self, nm, rn):
4          self.name=nm
5          self.rollno=rn
6      def show_student(self):
7          print("The name of Student is:", self.name , "and the Rollno is:", self.rollno)
8
9  class Department:
10     @staticmethod
11     def show(s):
12         print("Student Name:", s.name)
13         print("Student RollNo:", s.rollno)
14         s.show_student()
15
16  st1=Student('Adil', 100)
17  Department.show(st1)

```

Nested Class

A class within a class is called as nested class or nesting of a class.

```
class OuterClassName:
```

```
    def __init__(self):
```

```
        self.variable_name = value
```

```
        self.innerClassObjectName = self.InnerClassName( )
```

← Inner Class Object

```
    def method_name(self):
```

```
        method body
```

```
class InnerClassName:
```

```
    def __init__(self):
```

```
        self.variable_name = value
```

```
    def method_name(self):
```

```
        method body
```

```

1  class Student:
2      def __init__(self, nm, rn):
3          self.name=nm
4          self.rollno=rn
5          self.bk=self.Book()
6      def show_student(self):
7          print("The name of Student is:", self.name , "and the Rollno is:", self.rollno)
8      class Book:
9          def __init__(self):
10             self.name="OOP with Python"
11             self.author="Tony Gaddis"
12             self.price="$40"
13             self.pages=900
14         def show_book(self):
15             print("Book Name:", self.name, "Author:", self.author, "Price:", self.price, "Book Pages:", self.pages)
16     st1=Student('Adil',100)
17     print(st1.name)
18     st1.show_student()
19     print(st1.bk.name)
20     st1.bk.show_book()
21     bobject=st1.bk
22     print(bobject.name)
23     bobject.show_book()

```

Summary

- ✓ **Passing class members between classes**
- ✓ **Nested Class**

References

- Chapter 10, Starting out with Python by Tony Gaddis
- <https://docs.python.org/3/contents.html>