

Class Methods and members:

- Properties defined at class level are shared by all instances.
- Double-underscore properties are hidden from other classes.
- Use `@classmethod` before defining a method to create a class method.
- Use `@staticmethod` before defining a method to create a class method.

Understanding Inheritance:

class Publication:

```
def __init__(self, title, price):  
    self.title = title  
    self.price = price
```

class Book(Publication):

```
//This Book class inherits from super class Publication  
def __init__(self, title, author, pages, price):  
    super().__init__(title, price)  
    //inherits __init__ from super class  
    self.author = author  
    self.pages = pages
```

#Note : Python also allows multiple inheritance.

Class C(B, A):

```
// class C inherits from B and A superclasses.  
def __init__(self):  
    super().__init__()
```

Abstract Base classes and Interfaces:

We need to add the following before starting of our python script to use Abstract base classes:

```
from abc import ABC, abstractmethod
```

//Check out interface.py file in Coders paradise python Level 0 to get this clear.

Compositions :

Linking two separate classes together with some common attribute or method.