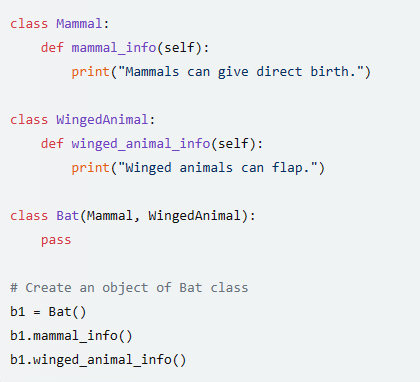
**Q1. What is the meaning of multiple inheritance?**

Ans: - Multiple inheritance in Python is a feature where a class can be derived from more than one superclass. Here is an example:



**Q2. What is the concept of delegation?**

Ans: - Delegation in Python refers to the process of passing a method call from one object (the delegator) to another object (the delegate) that performs the actual behavior of the method. Here is an example:

A screen shot of a computer program

Description automatically generated

**Q3. What is the concept of composition?**

Ans: - Composition in Python is an object-oriented design concept that models a “has-a” relationship. In composition, a class (known as the composite) contains an object of another class (known as the component). Here is an example:

A computer code with black text

Description automatically generated

In this example, the Composite class contains an object of the Component class.

**Q4. What are bound methods and how do we use them?**

Ans: - Bound methods in Python are methods that are dependent on the instance of the class as the first argument. They allow a class’s function to be called on an instance, instead of alone. Here is an example: -

A screenshot of a computer program

Description automatically generated

**Q5. What is the purpose of pseudoprivate attributes?**

Ans: - Pseudoprivate attributes in Python are used to “mangle” attribute names and they become less likely to conflict with attributes in subclasses. They are denoted by a double underscore prefix (and at most a single underscore suffix). For example, \_\_X in a class becomes \_classname\_\_X automatically. This feature is mostly intended to avoid namespace collisions in instances, not to restrict access to names in general. Here is an example:

A screen shot of a computer program

Description automatically generated

In this example, \_\_X is a pseudoprivate attribute. It is accessed within the class using self.\_\_X, but outside the class, it would be accessed using the mangled name