

**Q1. What is the difference between `__getattr__` and `__getattribute__`?**

**Ans:** The `__getattr__` method is run for fetches of undefined attributes only (i.e., those not present on an instance and not inherited from any of its classes). By contrast, the `__getattribute__` method is called for every attribute fetch, whether the attribute is defined or not. Because of this, code inside a `__getattr__` can freely fetch other attributes if they are defined, whereas `__getattribute__` must use special code for all such attribute fetches to avoid looping or extra calls.

**Q2. What is the difference between properties and descriptors?**

**Ans:** Properties serve a specific role, while descriptors are more general. Properties define get, set, and delete functions for a specific attribute; descriptors provide a class with methods for these actions, too, but they provide extra flexibility to support more arbitrary actions. In fact, properties are really a simple way to create a specific kind of descriptor, one that runs functions on attribute accesses. A property is created with a built-in function, and a descriptor is coded with a class; thus, descriptors can leverage all the usual OOP features of classes, such as inheritance. Moreover, in addition to the instance's state information, descriptors have local state of their own, so they can sometimes avoid name collisions in the instance.

**Q3. What are the key differences in functionality between `__getattr__` and `__getattribute__`, as well as properties and descriptors?**

**Ans:** The `__getattr__` and `__getattribute__` methods are more generic: they can be used to catch arbitrarily many attributes. In contrast, each property or descriptor provides access interception for only one specific attribute. We can't catch every attribute fetch with a single property or descriptor. On the other hand, properties and descriptors handle both attribute fetch and assignment by design: `__getattr__` and `__getattribute__` handle fetches only; to intercept assignments as well, `__setattr__` must also be coded. The implementation is also different: `__getattr__` and `__getattribute__` are operator overloading methods, whereas properties and descriptors are objects manually assigned to class attributes. Unlike the others, properties and descriptors can also sometimes avoid extra calls on assignment to unmanaged names, and show up in dir results automatically