

Descriptors in Python

Descriptors were introduced to Python way back in version 2.2. They provide the developer with the ability to add managed attributes to objects. The methods needed to create a descriptor are `__get__`, `__set__` and `__delete__`. If you define any of these methods, then you have created a descriptor.

The idea behind the descriptor is to get, set or delete attributes from your object's dictionary. When you access a class attribute, this starts the lookup chain. Should the looked up value be an object with one of our descriptor methods defined, then the descriptor method will be invoked.

Descriptors power a lot of the magic of Python's internals. They are what make properties, methods and even the **super** function work. They are also used to implement the new style classes that were also introduced in Python 2.2.