Q1. What is the difference between \_\_getattr\_\_ and \_\_getattribute\_\_?

\_\_getattribute\_\_Called unconditionally to implement attribute accesses for instances of the class. If the class also defines \_\_getattr\_\_() , the latter will not be called unless \_\_getattribute\_\_() either calls it explicitly or raises an AttributeError .

Q2. What is the difference between properties and descriptors?

the difference between attributes and properties in Python is that attributes are simply data members of an object, while properties are methods that are accessed like attributes but actually perform some computation when called.

Q3. What are the key differences in functionality between \_\_getattr\_\_ and \_\_getattribute\_\_, as well as properties and descriptors?

The main difference between \_\_getattr\_\_ and \_\_getattribute\_\_ is that if the attribute was not found by the usual way then \_\_getattr\_\_ is used. Whereas the \_\_getattribute\_\_ is used before looking at the actual attributes on the object.