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

Ans=>

\_\_getattr\_\_

Called when an attribute lookup has not found the attribute in the usual places

\_\_getattribute\_\_

Called unconditionally to implement attribute accesses for instances of the class

Q2. What is the difference between properties and descriptors?

Ans=> A descriptor is a mechanism behind properties, methods, static methods, class methods, and super()

It encapsulates instance attributes and provides a property

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

Ans=>A key difference between \_\_getattr\_\_ and \_\_getattribute\_\_ is that \_\_getattr\_\_ is only invoked if the attribute wasn't found the usual ways. It's good for implementing a fallback for missing attributes, and is probably the one of two you want.

descriptors are a low-level mechanism that lets you hook into an object's attributes being accessed. Properties are a high-level application of this; that is, properties are implemented using descriptors