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

Ans. \_\_getattr\_\_ gets called if there is no attribute in the instance.It’s invoked **last**, if Python can’t find that attribute. (lowest priority).

\_\_getattribute\_\_ gets called **all the times**, whether there is the attribute or not. It’s invoked fir**st.** (highest priority) So, even if there is the attribute in the instance, Python calls it first, with the attribute as an argument.

Q2. What is the difference between properties and descriptors?

Ans. Descriptors are Python objects that implement a method of the **descriptor protocol**, which gives you the ability to create objects that have special behavior when they’re accessed as attributes of other objects. It implements through \_\_get\_\_ , \_\_set\_\_ and \_\_del\_\_.

Property is the Pythonic way to avoid formal getter and setter methods in your code. This function allows you to turn class attributes into properties or managed attributes. Since property is a built-in function, you can use it without importing anything.

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

Ans.\_\_getattr\_\_ gets called if there is no attribute in the instance. It’s invoked **last**, if Python can’t find that attribute. (lowest priority).

\_\_getattribute\_\_ gets called **all the times**, whether there is the attribute or not. It’s invoked fir**st.** (highest priority) So, even if there is the attribute in the instance, Python calls it first, with the attribute as an argument.

Descriptors are Python objects that implement a method of the **descriptor protocol**, which gives you the ability to create objects that have special behavior when they’re accessed as attributes of other objects. It implements through \_\_get\_\_ , \_\_set\_\_ and \_\_del\_\_.

Property is the Pythonic way to avoid formal getter and setter methods in your code. This function allows you to turn class attributes into properties or managed attributes. Since property is a built-in function, you can use it without importing anything.