**Assignment\_1**

Q1. What is the purpose of Python's OOP?

1. In Python, object-oriented Programming (OOPs) is a programming paradigm that uses objects and classes in programming. It aims to implement real-world entities like inheritance…

Q2. Where does an inheritance search look for an attribute?

1. on the object The inheritance search first looks for the attribute in the object itself. If it is not found, it looks in the object’s parent class, then the parent class’s parent, and so on,

Q3. How do you distinguish between a class object and an instance object?

1. A class is a blueprint which you use to create objects. An object is an instance of a class - it's a concrete 'thing' that you made using a specific class. So, 'object' and 'instance' are the same thing, but the word 'instance' indicates the relationship of an object to its class.

Q4. What makes the first argument in a class’s method function special?

1. According to my understanding, the first argument passed in a class method is the class itself where that class method is defined. So for example, consider the following code: class A (object): \_\_x=10 @classmethod def clam (cls,\*args): print (cls.\_\_x) class B (A): \_\_x=50

Q5. What is the purpose of the \_\_init\_\_ method?

1. When you create a new object of a class, Python automatically calls the \_\_init\_\_ () method to initialize the object’s attributes. Unlike regular methods, the \_\_init\_\_ () method has two underscores (\_\_) on each side. Therefore, the \_\_init\_\_ () is often called dunder init. The name comes abbreviation of the double underscores init.

Q6. What is the process for creating a class instance?

1. A class is a user-defined blueprint or prototype from which objects are created. Classes provide a means of bundling data and functionality together. Creating a new class creates a new type of object, allowing new instances of that type to be made. Each class instance can have attributes attached to it for maintaining its state. Class instances can also have methods (defined by its class) for modifying its state.

Q7. What is the process for creating a class?

1. The first step in creating your own class is identifying what state you want to capture to describe objects. Let’s start with an example that doesn’t use any classes or objects: Code Editor ? This sketch uses four variables to store the state of a circle that bounces around the window.

Q8. How would you define the superclasses of a class?

1. In Python, a superclass or base class is defined just like a regular class using the class keyword. The superclass contains attributes and methods that are common to a group of related classes: The Vehicle class defined above contains the common make, model, and fuel attributes, along with a drive () method.