1. Explain the difference between a **class** and an **object**.
    1. A class contains member functions, variables, etc. An object is a specific instance of a class. The class is the objects definition.
2. What is the primary difference between a **member variable** and a regular variable?
    1. A member variable is associated with an object while regular variables are static (ie. Don't have to be part of an initialized object)
3. What is the primary difference between a **member function** and a regular function? Why do member functions require the `self` argument?
    1. Member functions are part of an object. The self argument is required so the function associated with the correct object is called. Regular functions aren't associated with objects.
4. What relationship is expressed by inheritance? How is this different from composition?
    1. Inheritance allows a derived class to utilize functions of its base class. Additionally, it allows new functions to be declared or redefined. Classes always inherit from, but are not made up of, base classes. Composition is what a class is made up of. It doesn't inherit the same things as a base class.
5. What is the purpose of the constructor?
    1. It initializes variables or calls certain functions when an object is created.
6. What is the special function name used when defining a constructor in Python? How many times is an object's constructor called during its lifetime?
    1. __init__ is used for a constructor. It is only called when an object is created.