Week 1 – In your Interface

1. Add the \_\_contains\_\_ protocol and show whether or not 'Tim' and 'Sam' are part of our team.
   1. Answer: Refer to the “Week 1 - In your Interface.py” file
2. Add the \_\_iter\_\_ protocol and show how you can print each member of the classmates object.
   1. Answer: Refer to the “Week 1 - In your Interface.py” file
3. Determine if the class classmates implements the \_\_len\_\_ method.
   1. Answer: This question is somewhat difficult to answer, mainly due to its wording. Technically, the Teams class (not the “classmates” class) does indeed properly incorporate the \_\_len\_\_ protocol in its overall structure. However, the \_\_len\_\_ method itself is considered by some to be an informal interface rather than a strict implementation, so it might be better to use a phrase other than “implements”.
4. Explain the difference between interfaces and implementation.
   1. Answer: The key difference between interfaces and implementation lies in how the user interacts with each one. An interface defines what an object can do but doesn’t actually do it; it describes how users of a class can interact with said class. An implementation, on the other hand, carries out the operation declared in the interface outside the user’s direct involvement.
5. Using both visual and written descriptions, think through the interface-implementation of a large-scale storage system. In many systems today, we have the ability to store information from a single application to a variety of storage devices - local storage (hard drive, USB), the cloud and/or some new medium in the future. How would you design an interface structure such that all of the possible implementations could store data effectively?
   1. Answer: In designing such a system, I would probably utilize a layered architectural framework similar to what is seen in most databases (refer to the diagram below for an illustrated example). As such, I would initialize the process by creating a base interface/presentation layer that allows users to create and manipulate files as needed. The data from those files would then be pulled to other layers via implementation methods for storage across a multitude of devices.

