Given an base class Pet.

Pet
-name: String -breed:String -weight:double
+__init__(name: String, breed:String, weight:double) +set_name(name:String) +get_name():name +get_weight():double +display() +__str__() +sound() +eat(food:String)

And two child class Dog and Cat that extends Pet

Dog
-microchipped:boolean
+__init__ (name: String, breed:String, weight:double, microchipped:boolean) +sound() +eat() +display() +__str__()

Cat
+__init__ (name: String, breed:String, weight:double)
+sound() +display() +__str__()

a. Implement the above three classes.
   - Sound => Pet has no sound, Cat meow and Dog bark.
   - Eat => Pet and Cat eat whatever is feed, Dog eat only cookies

b. Create a few Cat and Dog objects and put into a List.

c. Sound and feed the five pets in the List using a for loop

d. Change the Pet to an Abstract class *sound()* to abstract method()
   - Why do we make Pet "Abstract"?

e. Implement a Pet_App that allow user to
   Choose a Pet (Dog or Cat)
   Enter the Pet details
   The user then continuously choose to
   - Sound their Pet
   - Feed their Pet
   - Display the Pet

   Until the user choose to quit.