Class Description:

The class represents a dog shelter with different types of rescued dogs. These dogs are available for adoption for customers. The breed, name, age, hunger level, and sleep level of the dog is stored in a file because it is pre existing information about the dog which the customer cannot change. The purpose of the Dog_shelter class is for the customer to be able to pick a breed they want to adopt. The customer is also able to get the dog's attributes to understand the dog better. Lastly the customer can pick a location and donation amount if they want to donate to the shelter.

Variable Description:

Non class Variables:

1. dog_list: Stores the key and value pairs of the dogs in a dictionary. Where the breed of the dog is the key and the attributes of the dog are the values.
2. line: Each line of the text file is stored here as the loop iterates
3. dog_name: Each element in the line is stored here
4. filename: Stores the loaded file

Class Variables:

1. self.breed: Sets the breed variable as self.breed. Stores the breed customer picks as the Key in the dictionary.
2. self.name: Stores the name value pair for the breed picked by the customer
3. self.age: Stores the age value pair for the breed picked by the customer.
4. self.hunger.level: Stores the hunger level value pair for the breed picked by the customer.
5. self.sleep.level: Stores the sleep level value pair for the breed picked by the customer.
6. self.__location: Stores the choice of shelter location chosen by the customer.
7. self.__donation: Stores the choice of donation amount chosen by the customer.
8. answer: Concatenates the breed, a string and the value pair to be returned

Methods Description:

Non class Method:

1. load_dog_list: Reads through the file that contains the dog attributes. Then assigns the key and value pairs from the file. Takes in the filename argument which is already predefined.

Class Methods:

1. __init__: Allows the class to initialize the object's state. It takes in three arguments; self, dog_list, and breed. The self variable allows other methods to access variables that belong to the class. The dog_list argument refers to the dictionary created in the load_dog_list function which is taken as a parameter. Lastly the breed variable is the

input of the customer. The dog_list and breed don't have to be taken in as arguments in the other methods of the class as they are initialized with self, making them accessible to the other methods .

2. get_name: Initializes a self.name variable which is set to the name value pair of the key assigned by the customer. The method then enters a if statement and checks to see if the breed key exists in the dictionary. If it exists, then the answer variable is returned, which prints out the name of the dog. If the breed does not exist, then an error statement is returned.
3. get_age: Initializes a self.age variable which is set to the age value pair of the key assigned by the customer. The method then enters a if statement and checks to see if the breed key exists in the dictionary. If it exists, then the answer variable is returned, which prints out the age of the dog. If the breed does not exist, then an error statement is returned.
4. get_hunger_level: Initializes a self.hunger.level variable which is set to the hunger level value pair of the key assigned by the customer. The method then enters a if statement and checks to see if the breed key exists in the dictionary. If it exists, then the answer variable is returned, which prints out the hunger level of the dog. If the breed does not exist, then an error statement is returned.
5. get_sleep_level: Initializes a self.sleep.level variable which is set to the sleep level value pair of the key assigned by the customer. The method then enters a if statement and checks to see if the breed key exists in the dictionary. If it exists, then the answer variable is returned, which prints out the sleep level of the dog. If the breed does not exist, then an error statement is returned.
6. set_location: Initializes a self.__location variable which is set to the location chosen by the customer. Then the self.__location is returned.
7. set_donation: Initializes a self.__donation variable which is set to the location chosen by the customer. Then the self.__donation is returned.

Demo Program Description:

The purpose of the demo program is to test all the methods that belong to the class. The demo program is run in the main method. First an object from the class Dog_shelter is created. Then using the object "x", the methods are called using the dot operator. Functions that require arguments, are given the needed arguments.

Demo Program Instructions:

The demo program can be run by clicking on the play icon. You can change the breed argument to another type that is available at the shelter to see how the code changes. The location and donation amounts are also changeable to a location and amount you prefer. Once you are satisfied with the given arguments, run the program. The dog's attributes are then printed out in the terminal when the program is run.