# Attributes

**An attribute is a characteristic of an object.**

**For example, we can create a class called Dog. An attribute of a dog may be its breed or its name.**

**The syntax for creating an attribute is:**

```
self.attribute = something
```

**Let's get a better understanding of attributes through an example.**

In [1]:

```
class Dog:
    def __init__(self,breed):
        self.breed = breed

sam = Dog(breed='Lab')
frank = Dog(breed='Huskie')
```

**Lets break down what we have above.The special method**

```
__init__()
```

**is called automatically right after the object has been created:**

```
def __init__(self, breed):
```

**Each attribute in a class definition begins with a reference to the instance object. It is by convention named self. The breed is the argument. The value is passed during the class instantiation.**

```
 self.breed = breed
```

**Now we have created two instances of the Dog class. With two breed types, we can then access these attributes like this:**

In [2]:

```
sam.breed
```

Out[2]:

```
'Lab'
```

In [3]:

```
frank.breed
```

Out[3]:

```
'Huskie'
```

**Note how we don't have any parentheses after breed; this is because it is an attribute and doesn't take any arguments.**

**In Python there are also *class object attributes*. These Class Object Attributes are the same for any instance of the class. For example, we could create the attribute *species* for the Dog class. Dogs, regardless of their breed, name, or other attributes, will always be mammals. We apply this logic in the following manner:**

In [4]:

```
class Dog:
    # Class Object Attribute
    species = 'mammal'

    def __init__(self,breed,name):
        self.breed = breed
        self.name = name
```

In [5]:

```
#Now create instances of the Dog class With the breed and name of the dog.
bruno = Dog('Lab','Sam')
```

In [6]:

```
bruno.breed
```

Out[6]:

```
'Lab'
```

**Note that the Class Object Attribute is defined outside of any methods in the class. Also by convention, we place them first before the init.**

In [7]:

```
bruno.species
```

Out[7]:

```
'mammal'
```

In [ ]:

```

```