**Explanation**

**Class Pet:**

This serves as the base class for all pets in the application.

It contains properties such as Name and Age to represent the basic attributes of a pet.

The **DisplayInfo()** method is responsible for displaying the pet's information.

**Class Dog (Derived from Pet):**

Extends the Pet class to add specific features related to dogs.

Introduces an additional property Breed to represent the breed of the dog.

Utilizes method overloading with the DisplayInfo(bool includeBreed) method. This enables flexibility in displaying the dog's information, allowing the user to choose whether to include the breed or not.

**Class PetSeller:**

Represents a pet seller in the application.

Encapsulates the seller's name and the pet they are selling.

It has a method SellPet() that facilitates the selling process by displaying the pet's information.

**Method Overloading:**

Demonstrated in the Dog class with the DisplayInfo(bool includeBreed) method.

Allows the DisplayInfo method to be invoked with or without the includeBreed parameter, providing flexibility in how the dog's information is displayed.

**Creating Instances:**

Instantiates objects of the Pet and Dog classes to represent specific pets in the application.

Sets the properties of these objects to define their characteristics.

**Displaying Pet Information:**

Calls the DisplayInfo() method on the pet objects to print their details to the console.

In the case of Dog, the DisplayInfo(bool includeBreed) method is invoked to demonstrate method overloading.

**Selling a Pet:**

Creates an instance of the PetSeller class, representing a seller offering a specific pet for sale.

Calls the SellPet() method of the PetSeller object to initiate the selling process, which displays the pet's information to the user.

Overall, this program demonstrates a simple scenario of managing pets, sellers, and the selling process while showcasing various object-oriented programming concepts such as inheritance, encapsulation, method overloading, and polymorphism.