6. Turn your attention to the **Main**( ) method in the default **Program class** and create an instance of the **Cat class**

   ```csharp
   Cat tigger = new Cat();
   ```

7. Now, retrieve all (default) properties of the new object

   ```csharp
   string tagT = String.Format("{0} is a {1} year old {2} cat",
   tigger.getName(),
   tigger.getAge(),
   tigger.getColor());
   ```

8. Display all properties and call the miscellaneous method

   ```csharp
   Console.WriteLine( tagT + tigger.cry() );
   ```

9. Now, create another instance of the **Cat class** and set each property with new values

   ```csharp
   Cat smokey = new Cat();
   smokey.setName("Smokey");
   smokey.setAge(2);
   smokey.setColor("Gray");
   ```

10. Next, retrieve all (adjusted) properties of this new object

    ```csharp
    string tagS = String.Format("{0} is a {1} year old {2} cat",
    smokey.getName(),
    smokey.getAge(),
    smokey.getColor());
    ```

11. Display all properties and call the miscellaneous method

    ```csharp
    Console.WriteLine( tagS + smokey.cry() );
    Console.ReadKey();
    ```

12. Press **Start** or **F5** to run the application and see the properties of each object instance and method called

Object instances cannot be created from **static classes**, but you can supply a constructor method in **static classes**.

You can also specify parameters to a constructor method in order to allow argument values to be passed in when a **new** instance object is created.