**Object oriented programing**

1. *A style of programming*
2. **Goals** 
   1. Break up a program into logical pieces
   2. Make a problem easier to communicate about
   3. Make it easier to reuse code
3. **Key concept #1: objects** 
   1. *An object is an encapsulated chunk of code that can have* 
      1. **Properties (usually nouns or adjectives associated with a thing)** 
         1. Cars color is blue
         2. Dog has a name
         3. Ticket costs $74
      2. **Methods (usually verbs – what a thing does or how it behaves)** 
         1. Car accelerates
         2. **Dog barks**
4. **Key concept #2: classes vs instances** 
   1. A class is a category of objects e.g. Dog (general category)
   2. An “instance” is a specific manifestation of a class e.g. spot (specific dog)
   3. Be careful not to mix these up
   4. A class is the code that contains the definition of a category of objects-its where the available properties and methods are specified
   5. The terms instantiate and instantiation refer to the process of creating an object from a class ex: spot is an instantiation of dog
   6. Classes are also freq referred to as types
5. **Terminology** 
   1. Most programming languages have a handful of basic built in types
      1. **Integers**
         1. Int, int8, int32
      2. **Decimals** 
         1. Float, double
      3. **Text** 
         1. String, str, char
      4. **Boolean** 
         1. True/false values
      5. **Null**
         1. Undefined, void
   2. These basic types are classed scalar types
   3. Classes are referred to as object types
      1. Ex: spot is an object of type dog
6. Graphical user interface, website

   Description automatically generated
7. **Syntax** 
   1. The actual code used to express classes and objects is different in different programming languages
   2. There is no substitute for just learning it
   3. **Common syntax** 
      1. Spot = new dog(name = “spot”)