

Objects III

- You should have a dog class that supports energy. Playing fetch decreases a dog's energy, sleeping increases it.
- If you didn't finish this from last time, get the code from the website (day 29). **You should add the sleep(int hours) method.**
- **Warmup:** Add a method chase(dog & otherDog) to your class. This method will let your dog chase another dog. Inside the method, print a message with both dogs' names. This method should decrease both dogs' energies.
- **Hint:** Inside a class, you have access to your own object's private variables, plus private variables of other objects that are passed in as arguments!

Constructors and destructors

- A ***constructor*** is a method that is run automatically when an object is created.
- A ***destructor*** is a method that is run automatically when an object is "destroyed."
 - For objects on the stack, destroyed == goes out of scope.
 - For objects on the heap, destroyed == is deleted.

Constructors

- Constructors are commonly used to initialize the fields (variables) in a class to appropriate values.
- Without constructors, the user would have to set all the fields in a class by hand after each object creation.
- The name of a constructor is always the same name as the class itself.

Dog default constructor

- What are appropriate values to initialize each field to in our dogs?

Dog constructors

- Classes can have multiple constructors.
- The default constructor never takes any arguments, but other constructors can.
- These arguments are typically used to set the fields of the class.