



Project 15: Animals

Using Derived Classes and Inheritance



Project 15: Animals

- In this project we will create a base class to represent animals of various kinds.
 - Class Animal
- Then we will create derived classes to represent specific kinds of animals.
 - Dog
 - Cat
 - Bird



Class Animal

- Every animal has
 - Kind of animal (Dog, Cat, Bird)
 - Name
 - Owner's name
 - Date of Birth
- Create a class, Animal, to hold information common to all animals.
 - Will be a base class for derived classes representing specific kinds of animals.



Class Animal

- Define instance variables for the information that is common to all animals:
 - `private String kind_of_animal;`
 - `private String name;`
 - `private String owner_name;`
 - `private Calendar date_of_birth;`
- Provide a constructor that has parameters of the same name and type:

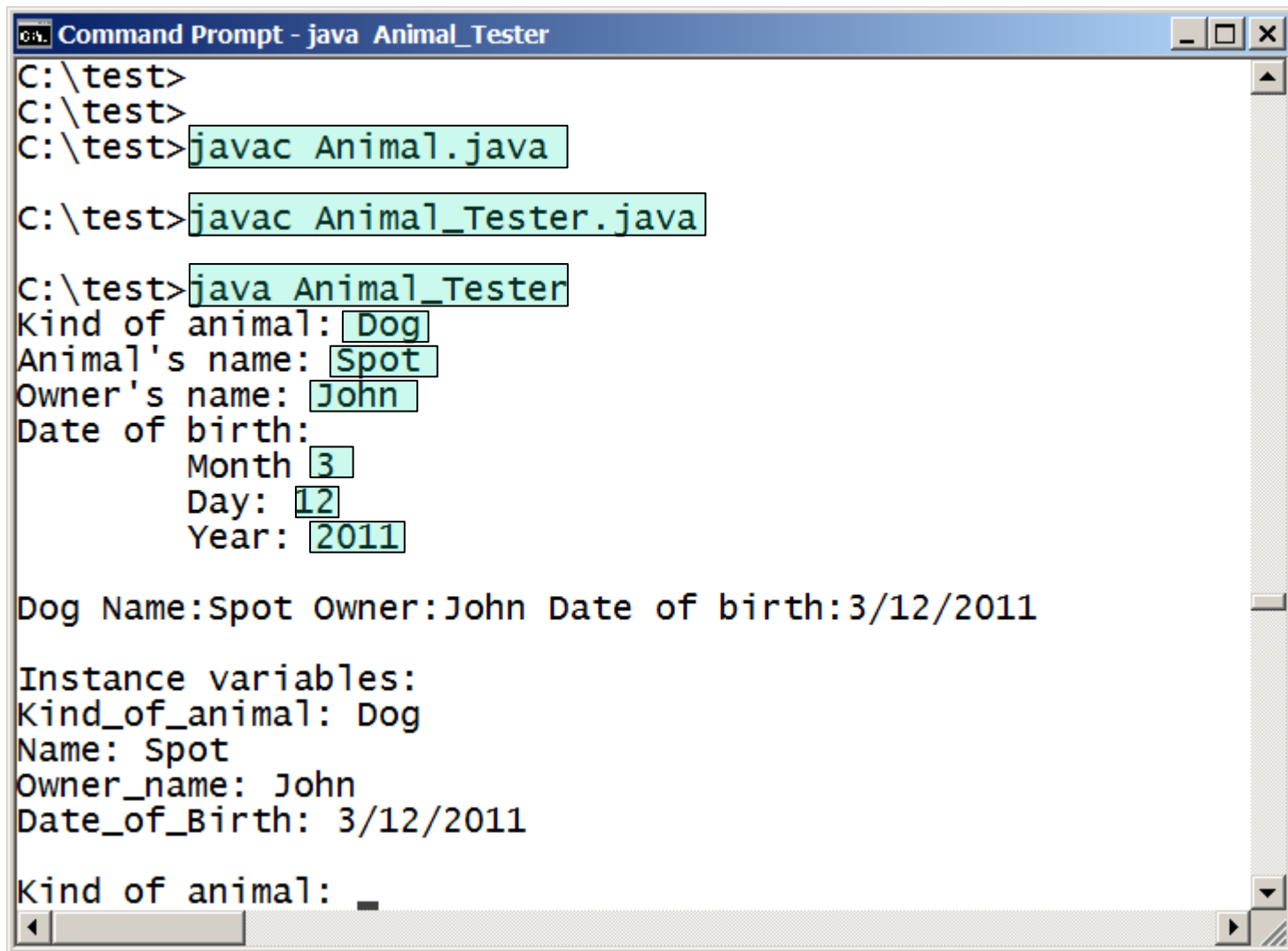
```
public Animal(String kind_of_animal, String name,  
              String owner_name, Calendar dob)
```



Class Animal

- Provide accessor methods for all instance variables.
 - Same name as the instance variable with first letter capitalized.
- Provide a toString method.
 - See test run.
- Download program Animal_Tester from the class web site, and use it to test your Animal class.
 - http://www.csee.usf.edu/~turnerr/Programming_Concepts/Downloads/Project_15_Animals/

Test Run



```
Command Prompt - java Animal_Tester
C:\test>
C:\test>
C:\test>javac Animal.java
C:\test>javac Animal_Tester.java
C:\test>java Animal_Tester
Kind of animal: Dog
Animal's name: Spot
Owner's name: John
Date of birth:
    Month 3
    Day: 12
    Year: 2011

Dog Name:Spot Owner:John Date of birth:3/12/2011

Instance variables:
Kind_of_animal: Dog
Name: Spot
Owner_name: John
Date_of_Birth: 3/12/2011

Kind of animal: _
```



Class Dog

- Create class Dog as a derived class from class Animal.
- Let class Dog have instance variables
 - `private String breed;`
 - `private String favorite_treat;`



Class Dog

- The constructor for class Dog should have parameters for breed and favorite treat, in addition to the parameters for class Animal
 - Except it doesn't need kind_of_animal, because we know that will be Dog.

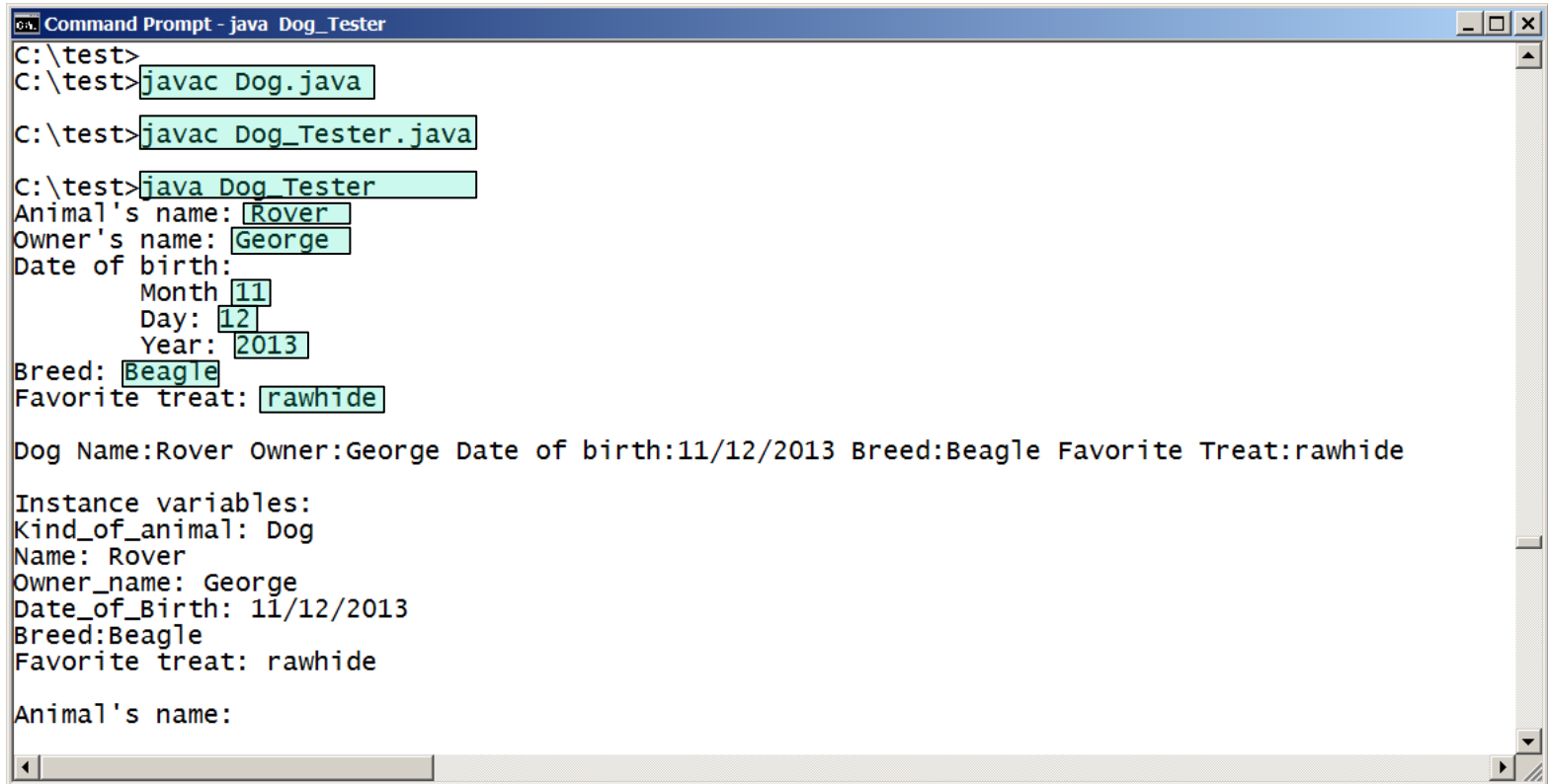
```
public Dog(String name,  
            String owner_name, Calendar dob,  
            String breed, String favorite_treat)
```




Class Dog

- Provide accessor methods for all instance variables specific to class Dog.
- Provide a toString method.
 - See test run.
- Download program Dog_Tester from the class web site, and use it to test your Dog class.
 - http://www.csee.usf.edu/~turnerr/Programming_Concepts/Downloads/Project_15_Animals/

Test Run



```
Command Prompt - java Dog_Tester
C:\test>
C:\test>javac Dog.java
C:\test>javac Dog_Tester.java
C:\test>java Dog_Tester
Animal's name: Rover
Owner's name: George
Date of birth:
    Month: 11
    Day: 12
    Year: 2013
Breed: Beagle
Favorite treat: rawhide

Dog Name:Rover Owner:George Date of birth:11/12/2013 Breed:Beagle Favorite Treat:rawhide

Instance variables:
Kind_of_animal: Dog
Name: Rover
Owner_name: George
Date_of_Birth: 11/12/2013
Breed:Beagle
Favorite treat: rawhide

Animal's name:
```



Class Cat

- Add class Cat, derived from class Animal.
- Cats have the following attributes in addition to those of class Animal:
 - Breed
 - Preferred Catfood



Class Cat

- The constructor for class Cat should have parameters for breed and preferred catfood, in addition to the parameters for class Animal
 - Except it doesn't need kind_of_animal, because we know that will be Cat.

```
public Cat(String name,  
           String owner_name, Calendar dob,  
           String breed, String preferred_catfood)
```



Class Cat

- Provide accessor methods for all instance variables specific to class Cat.
- Provide a toString method.
 - See test run.
- Download Cat_Tester.java from the class web site, and use it to test your Cat class.
 - http://www.csee.usf.edu/~turnerr/Programming_Concepts/Downloads/Project_15_Animals/

Test Run

```
Command Prompt - java Cat_Tester
C:\test>
C:\test>
C:\test> javac Cat.java
C:\test> javac Cat_Tester.java
C:\test> java Cat_Tester
Animal's name: Fuzzy
Owner's name: Betty
Date of birth:
    Month 4
    Day: 1
    Year: 2000
Breed: Persian
Preferred catfood: Friskies

Cat Name:Fuzzy Owner:Betty Date of birth:4/1/2000 Breed:Persian Preferred cat food:Friskies

Instance variables:
Kind_of_animal: Cat
Name: Fuzzy
Owner_name: Betty
Date_of_Birth: 4/1/2000
Breed:Persian
Preferred_catfood: Friskies

Animal's name:
```



Class Bird

- Create class Bird as a derived class from class Animal.
- Let class Bird have instance variables
 - `private String species;`
 - `private double wingspan;`



Class Bird

- The constructor for class Bird should have parameters for the instance variables that are specific to class Bird, in addition to the parameters for class Animal
 - Except it doesn't need `kind_of_animal`, because we know that will be Bird.

```
public Bird(String name,  
             String owner_name, Calendar dob,  
             String species, double wingspan)
```




Class Bird

- Provide accessor methods for all instance variables specific to class Bird.
- Provide a toString method.
 - See test run.
- Download program Bird_Tester from the class web site, and use it to test your Bird class.
 - http://www.csee.usf.edu/~turnerr/Programming_Concepts/Downloads/Project_15_Animals/

Test Run

```
Command Prompt - java Bird_Tester
C:\test>
C:\test>javac Bird.java
C:\test>javac Bird_Tester.java
C:\test>java Bird_Tester
Animal's name: Tweety Pie
Owner's name: Sherry
Date of birth:
    Month 4
    Day: 4
    Year: 2008
Species: Parakeet
Wingspan: 6.6

Bird Name:Tweety Pie Owner:Sherry Date of birth:4/4/2008 Species:Parakeet wingspan:6.6

Instance variables:
Kind_of_animal: Bird
Name: Tweety Pie
Owner_name: Sherry
Date_of_Birth: 4/4/2008
Species:Parakeet
Wingspan: 6.6

Animal's name:
```



Submission

- Put your Java class files into a folder and zip it.
 - Animal.java, Dog.java, Cat.java, Bird.java
 - Only .java files. Nothing else.
 - Don't submit the test drivers. (We already have them!)
- Submit your zipped folder via Canvas Assignments.
- Project is due by 11:59 PM
 - Sunday, April 24 **All Sections**
- Recommendation:

Do this project in your lab session or help session.



Ground Rules

- It is OK to *discuss* the project with other students BUT
 - Do not share your code with other students.
 - Before or after submitting the project.
- Do not copy any other student's code.
 - Or even look at it.
- Do not let anyone copy or examine your code.



Ground Rules

Except for code posted on the class web site

- Do not copy code from the Internet
 - or any other source (other than the textbook.)
- Do not ask for help on an Internet forum.
 - If you need help, ask your instructor or a TA.
 - Come to lab and help sessions.
- Write your own code.