CIS 1068

November 14, 2024

Administrative Stuff

- ► Lab 9 due
- ► Lab 10 soon
- ► Reading. Not as simple

Administrative Stuff. Dates

```
   November 2024
Su Mo Tu We Th Fr Sa
                1  2
 3  4  5  6  7  8  9
10 11 12 13 14 15 16
17 18 19 20 21 22 23
24 25 26 27 28 29 30

   December 2024
Su Mo Tu We Th Fr Sa
 1  2  3  4  5  6  7
 8  9 10 11 12 13 14
15 16 17 18 19 20 21
22 23 24 25 26 27 28
29 30 31
```
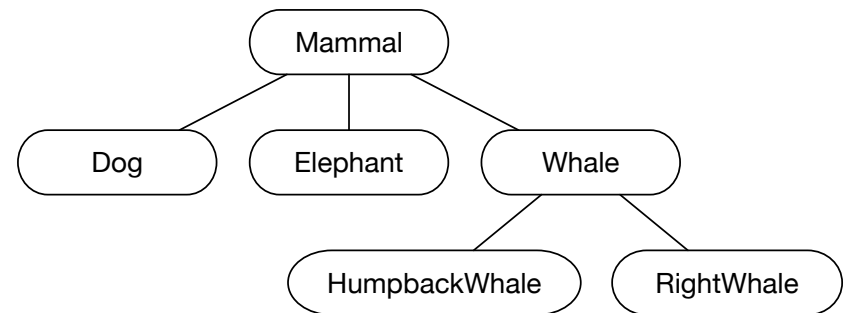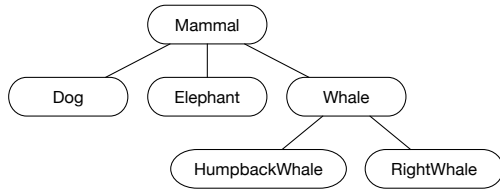
dates

Nov 14 today

Nov 23 - Dec 1 break

Dec 5 last class

Dec 9 last lab

Dec 10 study day

Dec 12 final exam 8 AM - 10 AM. Really 8 AM

Warm up. Class Hierarchy

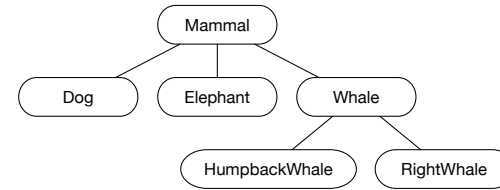## OK?

Mammal m = new Whale();
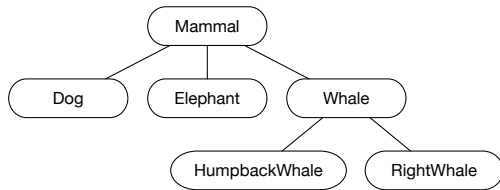
- ► What can m store?
- ► What is stored?
- ► OK?

OK?

Mammal m = new Whale();

- ► What can m store?
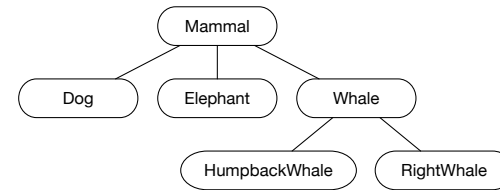 **location of a mammal**
- ► What is stored?
- ► OK?

OK?

Mammal m = new Whale();

- ► What can m store?
  location of a mammal
- ► What is stored?
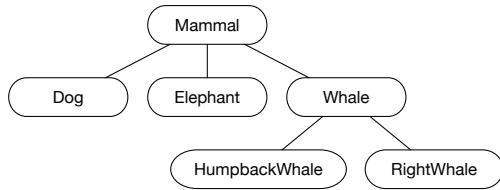 **location of a Whale**
- ► OK?

OK?

Mammal m = new Whale();

- ► What can m store?
  location of a mammal
- ► What is stored?
  location of a Whale
- ► OK? **Yes**

OK?
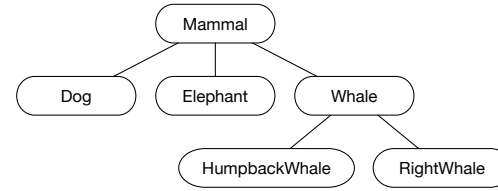


```
RightWhale r =
        new Whale();
```

- What can `r` store?
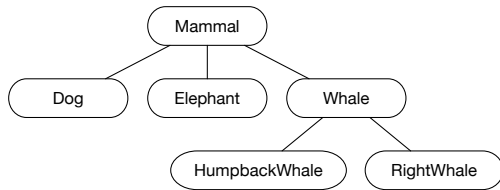- What is stored?
- OK?

OK?


```
RightWhale r =
        new Whale();
```

- What can `r` store?
  location of a RightWhale
- What is stored?
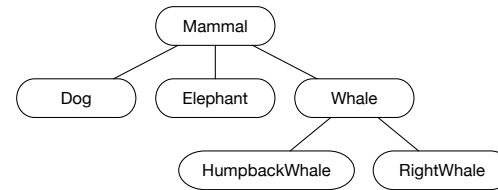- OK?

OK?


```
RightWhale r =
        new Whale();
```

- What can `r` store?
  location of a RightWhale
- What is stored?
  location of a Whale
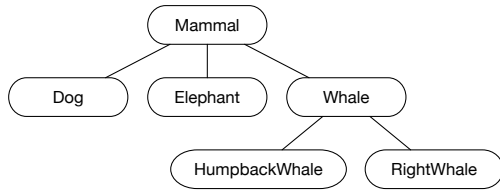- OK?

OK?


```
RightWhale r =
        new Whale();
```

- What can `r` store?
  location of a RightWhale
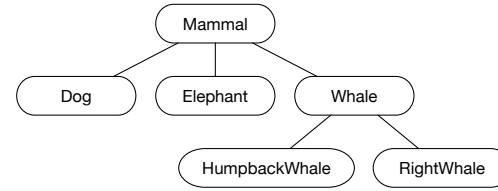- What is stored?
  location of a Whale
- OK? **No**

```
Mammal m =
    new RightWhale();
```

► What can m store?
► What is stored?
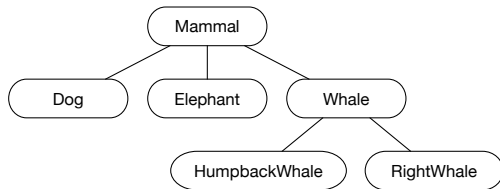► OK?

```
Mammal m =
    new RightWhale();
```

► What can m store?
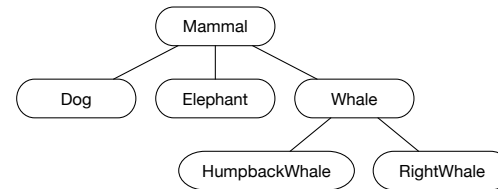**location of a mammal**
► What is stored?
► OK?

```
Mammal m =
    new RightWhale();
```

► What can m store?
location of a mammal
► What is stored?
location of a RightWhale
► OK?

```
Mammal m =
    new RightWhale();
```

► What can m store?
location of a mammal
► What is stored?
location of a RightWhale
► OK? **Yes**

Last time: lots on polymorphism

- polymorphism practice
- recursion
- exceptions