



# CMSC 131

Fall 2018

# Privacy Leaks

Recall the Pirate example from last time...

- Assume that the bird class is **mutable**. (Perhaps there is an instance method called “changeColor”.)
- Assume that we don’t want the state of the bird changed by anyone other than the Pirate who has the bird.

Consider the getter (getBird) of the Pirate class. Any problems?

This is called a “privacy leak”.  
How can we fix it?

Assume that the EyePatch is **immutable**.  
Is the getPatch method OK?

## Complex Example

Take a look at the CD and RewritableCD classes.  
What property distinguishes them?

Take a look at the two getters in the CDOwner class.  
Let's draw memory diagrams.  
Does either generate a memory leak?  
Which one is best?

Now let's do the same analysis for the  
RewritableCDOwner class.

## Complex Example, continued...

Take a look at the three getters in the CDCollectionOwner class.

Let's draw memory diagrams.

Do any generate memory leaks?

Which one is best?

Now let's do the same analysis for the RewriteableCDCollectionOwner class.

## Complex Example, continued...

Changing date of birth example

Date.java, Person.java, PersonDriver.java

Do any generate memory leaks?

# Arrays of Array References

Since arrays are objects, can we make an array containing a list of other Arrays?