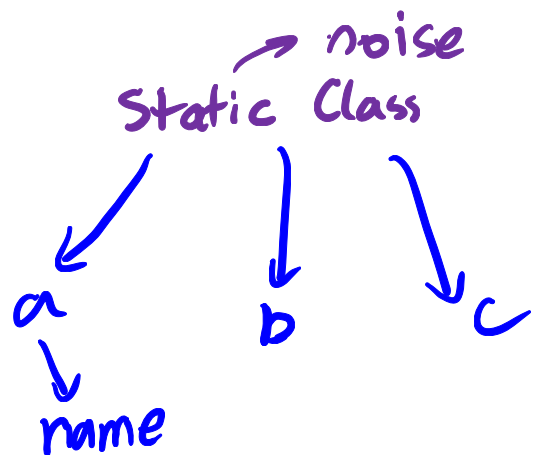# 1   Cat

```
1   public class Cat {
2       public String name;
3       public static String noise;`
4
5       public Cat(String name, String noise) {
6           this.name = name;
7           this.noise = noise;
8       }
9
10      public void play() {
11          System.out.print(noise);
12          System.out.println(" I'm " + name + " the cat!");
13      }
14
15      public static void anger() {
16          noise = noise.toUpperCase();
17      }
18
19      public static void calm() {
20          noise = noise.toLowerCase();
21      }
22
23      public static void main(String[] args) {
24          Cat a = new Cat("Cream", "Meow!");
25          Cat b = new Cat("Tubbs", "Nyan!");
26          a.play();
27          b.play();
28          Cat.anger();
29          a.calm();
30          a.play();
31          b.play();
32      }
```

What would be output after the execution of main is completed? Select answers such that they match the order of the output you have determined.

[X] A. Nyan! I'm Cream the cat!
[ ] B. Meow! I'm Cream the cat!
[ ] C. nyan! I'm Cream the cat!
[ ] D. nyan! I'm Tubbs the cat!
[X] E. Nyan! I'm Tubbs the cat!

[ ] F. Meow! I'm Tubbs the cat!
[ ] G. Nyan! I'm Cream the cat!
[✗] H. nyan! I'm Cream the cat!
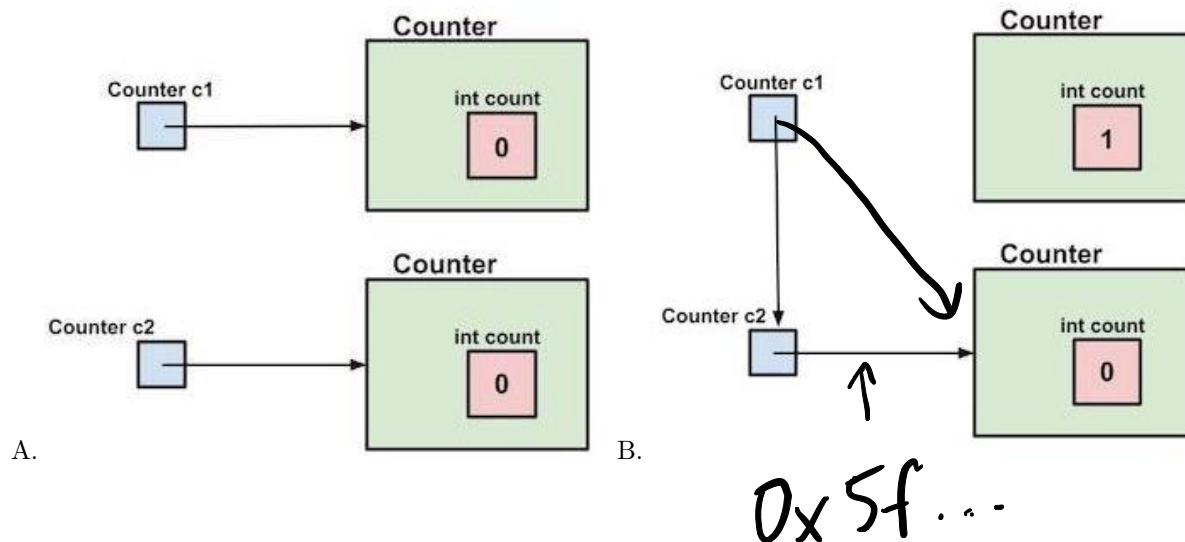[✗] I. nyan! I'm Tubbs the cat!
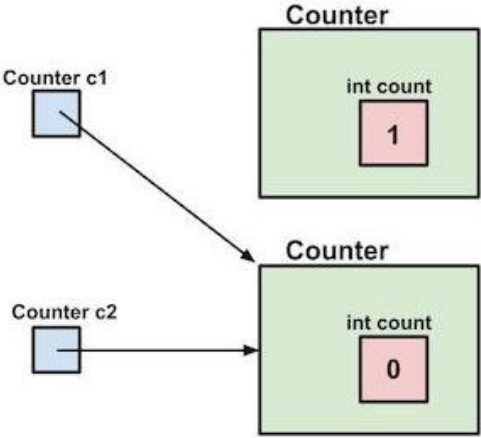[ ] J. nyan! I'm Cream the cat!

2   Box and Pointer

```java
public class Counter {

    int count = 0;

    void increment() {
        count = count + 1;
    }

    public static void main(String[] args) {
        Counter c1 = new Counter();
        c1.increment();
        Counter c2 = new Counter();
        c1 = c2;
    }
}
```

Circle the box-and-pointer diagram which best represents the state of the program
at the end of the main method before exiting. (For those of you with some Java-foo,
there is no garbage collection).

A.

B.

0x 5f ...

Counter

Counter c1

int count

1

Counter

Counter c2

int count

0

C.

( ) A ( ) B ( ) C

$$F_n = F_{n-1} + F_{n-2}$$

3 Fib

```
1  public static int fib(int n) {
2      if (n <= 1) {
3          // line one
4      } else {
5          // line two
6      }
7  }
```

Implement `fib` which takes in an integer `n` and returns the `nth` Fibonacci number.

The Fibonacci sequence is

$$0, 1, 1, 2, 3, 5, 8, 13, 21, \ldots$$

Select all valid statements for line one:

[✗] return n;
[ ] return;
[ ] return n + fib(1);
[ ] return n + 1;

Select all valid statements for line two:

[ ] return n + n - 1;
[✗] return fib(n-1) + fib(n-2);
[ ] return fib(n) + fib(n-1);
[✗] return fib(n-2) + fib(n-1);
[ ] return fib(n-1) + fib(n);

$$fib(2) = fib(0) + fib(1)$$
$$= 0 + 1 = 1$$
$$fib(3) = fib(1) + fib(2)$$
$$= null + 1 \quad fib(2)$$

4 Git

We have a project with files `Cat.java`
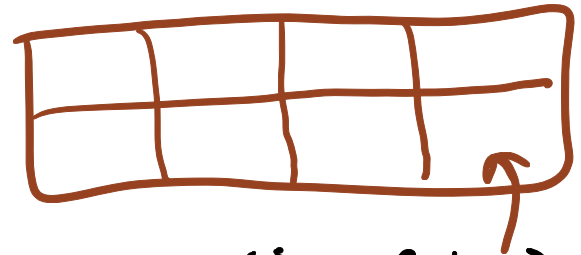`Animal.java`
`Dog.java`
`Poodle.java`
and `_data.txt` a very large data file that should only be stored locally.

Help us get credit for our work! What steps should we do in order to ensure our code is correctly reflected in our repo?

[ ] A. git add *
[✗] B. git add Cat.java Animal.java Dog.java
[ ] C. git add _data.txt
[✗] D. git add Poodle.java
[✗] E. git commit -m "Committing Animals project!"
[ ] F. git commit *
[✗] G. git push
[ ] H. git pull