Exercise 1 - pre04

Exercise 1

Logical operators can simplify nested conditional statements. For example, can you rewrite this code using a single `if` statement?

```
if (x > 0) {
    if (x < 10) {
        System.out.println("positive single digit number.");
    }
}
```

Answer:

Exercise 1

```
if (x > 0 && x < 10) {
        System.out.println("positive single digit number.");
    }
```

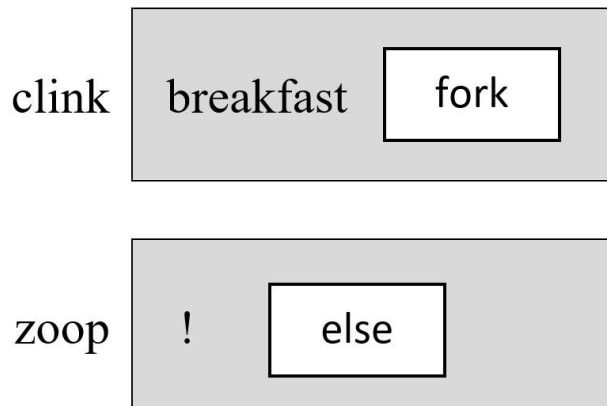# Exercise 2 -pre04 (program file Zoop02.java)

Exercise 2 For the following program:

1. Draw a stack diagram that shows the state of the program the *second* time `zoop` is invoked.

2. What is the complete output?

```java
public static void zoop(String fred, int bob) {
    System.out.println(fred);
    if (bob == 5) {
        ping("not ");
    } else {
        System.out.println("!");
    }
}

public static void main(String[] args) {
    int bizz = 5;
    int buzz = 2;
    zoop("just for", bizz);
    clink(2 * buzz);
}
```

```java
public static void clink(int fork) {
    System.out.print("It's ");
    zoop("breakfast ", fork);
}

public static void ping(String strangStrung) {
    System.out.println("any " + strangStrung + "more ");
}
```
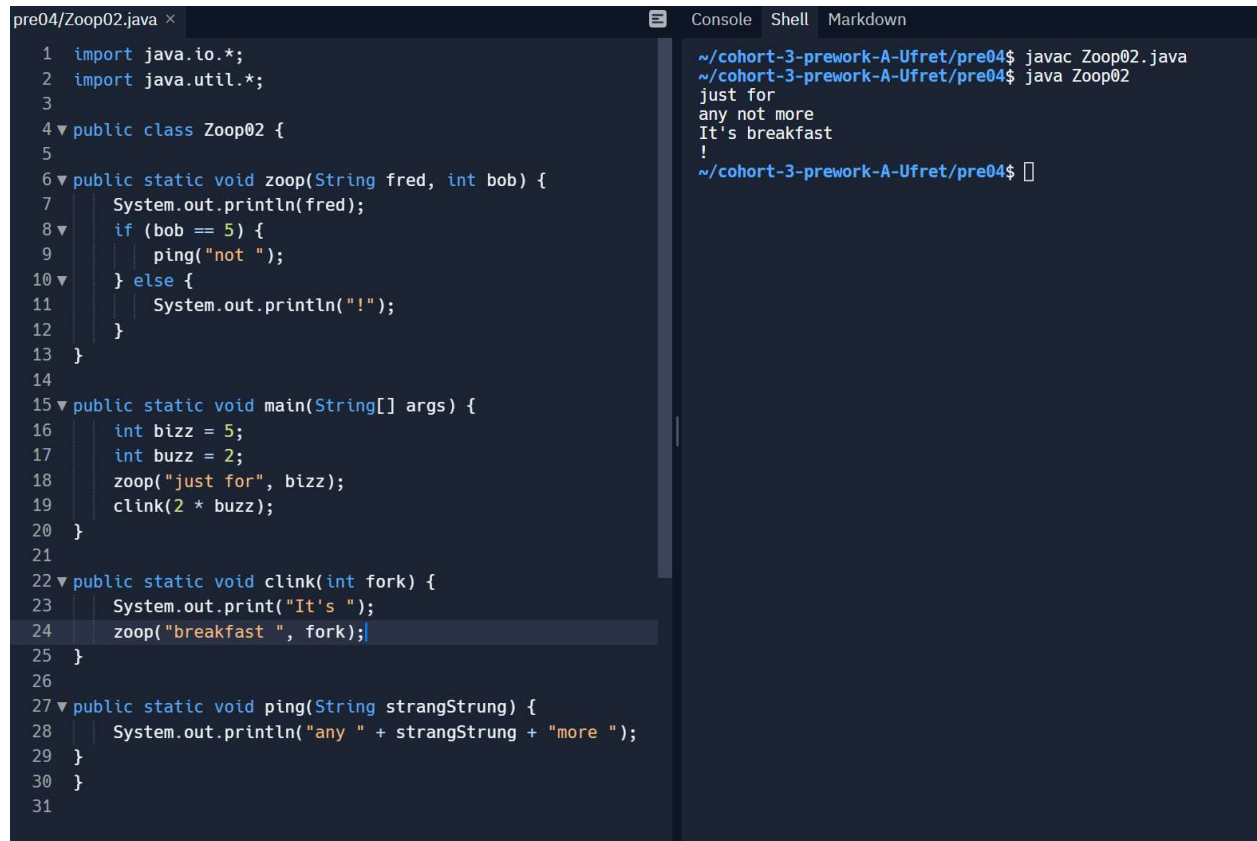
1. Draw a stack diagram that shows the state of the program the *second* time `zoop` is invoked.

The stack diagram for the **second** time zoop is involved is:

2. The complete output is:

    just for

    any not more

    It's breakfast

    !


```
pre04/Zoop02.java ×                                          Console  Shell  Markdown

  1   import java.io.*;                            ~/cohort-3-prework-A-Ufret/pre04$ javac Zoop02.java
  2   import java.util.*;                          ~/cohort-3-prework-A-Ufret/pre04$ java Zoop02
  3                                                 just for
  4 ▼ public class Zoop02 {                         any not more
  5                                                 It's breakfast
  6 ▼ public static void zoop(String fred, int bob) {   !
  7       System.out.println(fred);                 ~/cohort-3-prework-A-Ufret/pre04$ ▯
  8 ▼     if (bob == 5) {
  9           ping("not ");
 10 ▼     } else {
 11           System.out.println("!");
 12       }
 13   }
 14
 15 ▼ public static void main(String[] args) {
 16       int bizz = 5;
 17       int buzz = 2;
 18       zoop("just for", bizz);
 19       clink(2 * buzz);
 20   }
 21
 22 ▼ public static void clink(int fork) {
 23       System.out.print("It's ");
 24       zoop("breakfast ", fork);
 25   }
 26
 27 ▼ public static void ping(String strangStrung) {
 28       System.out.println("any " + strangStrung + "more ");
 29   }
 30   }
 31
```

Ch5 Ex2 - pre04 Zoop02.java with output (Screenshot)

Exercise 3 -pre04

Exercise 3

Draw a stack diagram that shows the state of the program in Section 5.8 after

`main` invokes `nLines` with the parameter `n == 4`, just before the last

invocation of `nLines` returns.

```java
public static void countdown(int n) {
    if (n == 0) {
        System.out.println("Blastoff!");
    } else {
        System.out.println(n);
        countdown(n - 1);
    }
}
```

main

nLines   n   4

nLines   n   3

nLines   n   2

nLines   n   1