

Model 1 Months of the Year

Open *JShell* on your computer. Type (or copy and paste) the following enum definition:

```
public enum Month {  
    JAN, FEB, MAR, APR, MAY, JUN, JUL, AUG, SEP, OCT, NOV, DEC;  
}
```

Then type each line of code below in *JShell*, *one at a time*, and record the results. You only need to record the output to the right of the “==>” symbol. For example, if *JShell* outputs \$8 ==> true, then just write true. If an error occurs, summarize the error message.

Java code	Shell output
Month m = null; m = JUN; m = Month.JUN; m.toString()	
Month spring = Month.MAR; Month summer = Month.JUN; m == spring m == summer Month.JUL = summer;	
m.ordinal() spring.ordinal() Month.OCT.ordinal() m.compareTo(spring) m.compareTo(Month.OCT)	
m = Month.valueOf("Mar"); m = Month.valueOf("MAR"); m == spring m = Month.valueOf(5); m = new Month("HEY");	
Month[] all = Month.values(); all[0] all[11] all[12]	

Questions (25 min)

Start time:

1. Consider the variables JAN, FEB, MAR, etc. Based on how they were used:
 - a) Are they `public`?
 - b) Are they `static`?
 - c) Are they `final`?
2. Is the variable "m" a primitive type or a reference type? Justify your answer. (If primitive, what is its value? If not, what does it reference?)
3. What ability do classes have that enums do not have? (*Hint*: Review the error messages.)
4. Based on your previous answers, explain why it's okay to compare enum variables using the `==` operator (as opposed to calling the `equals` method).
5. What does the `ordinal` method return? Explain the range of possible values.
6. What does the `compareTo` method return? Explain how to interpret the results.
7. What does the `valueOf` method return?
8. What does the `values` method return?

9. Which of the aforementioned methods are `static`? Explain how you can tell.

10. The following code snippet prompts the user to input their birth month:

```
Scanner in = new Scanner(System.in);  
System.out.print("What month were you born? ");  
String line = in.nextLine();
```

- a) Declare a variable named `birth` and initialize it to the `Month` object that corresponds to the user input. (*Hint: Use `valueOf`.*)

- b) Output a message that tells the user the number of their birth month. For example, if the user inputs `MAY`, output the message `You were born in month #5.` (*Hint: Use `ordinal`.*)

- c) Write an enhanced `for` loop that outputs each of the `Month` names that come before `birth`. (*Hint: Use `values` and `compareTo`.*)