Midterm Practice

The following problems are similar to what you might see on an exam. **NOTE:** this is not the format for the exam. The format is explained in the slides from the midterm review. There will be more problems than are shown here, these are examples.

Given the following values, write the solution to each expression:

```java
int x = 3;
int y = 2;
int z = 5;

(x <= y || z > x)  
Math.pow(z,y)  
y / x
```

Convert the following switch statement to an if statement: (and know how to do the opposite!)

```java
String colorString = "blue";
int colorVal = 0;

switch(colorString){
  case "red":
    colorVal = 1;
    break;
  case "green":
    colorVal = 2;
    break;
  case "blue":
    colorVal = 3;
    break;
  default:
    colorVal = -1;
    break;
}
```
Write an if statement to check if the 1st character of the following string is a digit:

String inputString = “123 abc”;

____________________________________________________

Write how to convert the *myDouble* variable into the following variables: (2 points)

double myDouble = 5.0;

String myString = ________________________________

int myInt = ________________________________

____________________________________________________

Complete the following code snippet by entering in the blank below, how you would check to see if the contents of string1 are also found in string2: (1 points)

String string1 = "hey";
String string2 = " hey students";

    if (______________________________________){
        System.out.println("string1 is inside string2");
    }
}
Part 2

Write a **program** that asks the user for 3 doubles - 2 numbers \( a \) and \( b \), and a *target* number and prints out which distance is smallest (closer to the target). For example, see the following input and output:

1.0 -1.0 0.0 → distance is equal
1.0 -1.5 0.0 → distance from a is smaller
1.5 -1.0 0.0 → distance from b is smaller

**Note:** This program should work for any given doubles, positive or negative. The doubles will be given by the user (responding to a prompt) with a space between each double.

Write a **method** that takes an array of integers as a parameter and loops through the array, shuffling it randomly, then returns a string representation of that array.

For example, starting with an array

\[ [1, 2, 3, 4, 5, 6] \]

the resulting string may be:

“[5, 4, 2, 6, 1, 3]”

**Note:** Do not write a complete program, but a complete *method* only (including a method signature with parameters and return type)
Write a **method** that takes a string as an input and prints out each character an increasing amount of times, starting with 1. For example, if the string passed was "hello" the output would be:

```
h
ee
lll
llll
ooooo
```

**Note:** Do not write a complete program, but a complete *method* only (including a method signature with parameters and return type)

---

Write a **method** that takes a width and height as parameters and prints out a box in those dimensions in the following format:

- The corners are o's
- all other positions are -'s

For example, if the input was a width of 2, height of 2, the output would be:

```
O O
O O
```

If the input was a width of 5, height of 4, the output would be:

```
O - - - O
- - - - -
- - - - -
- - - - -
```
Write Two overloaded methods: one that takes as a parameter an int from 0-9, one that takes an alphabetic char (not a digit or special character). The method that takes an int should return a different random int from 0-9 (inclusive). The method that takes a letter should return the letter 5 places following the given letter in the alphabet. If the letter is at the end of the alphabet, it should wrap. For example:

```java
System.out.println(gimmieSomething(8));
System.out.println(gimmieSomething('a'));
System.out.println(gimmieSomething('x'));
```

would print:

4
f
c

Note:
- Don't write a full program (no main method), just the 2 overloaded methods
- Make sure the int method returns a random number, but does not return the same number passed in
- You do not need to validate the input – assume the arguments passed are correct