**Java 3 Self Test**

Write 2 Java projects based on each of the specified requirements. Feel free to go above and beyond for project 2.

Project #1: Calendar Printer

* 1. Ask user for 2 inputs:
		1. The number of days in the month (between 28 and 31)
		2. The day of month the first Sunday falls on (between 1 and 7)
	2. Print out an ASCII Calendar with the correct number of days, and with the first Sunday on the correct day.
	3. Hint: Use if statements, modulus, and for loops. Try to do this with as little code as possible. You may also find having a separate method that takes numbers of one or two digits, and returns a String with a space if the number is one digit, or the number if it is two digits e.g. 1 --> “ 1”, 10 --> “10”.



1. Project #2: Build a Fraction class and a MixedNumber Class
	1. The MixedNumber class will extend the Fraction class but have an extra whole number field.
	2. Both classes have a toString and an equals method. The equals method must take into account all combinations of possible calls i.e. Fraction.equals(MixedNumber) or MixedNumber.equals(MixedNumber) both classes should also implement the comparable interface.
	3. Both classes will have basic add, divide, multiply, and subtract methods, the MixedNumber one must override the ones in Fraction. These methods will be able to be called on Fraction objects from a client program. It is your choice whether to change the object being divided/added/multiplied/etc., or return the answer. It is also up to you and your specific client program whether you choose to allow Fractions to be added/multiplied/etc. with MixedNumbers and vice versa.
	4. Don’t forget to build a client program to test and demonstrate functionality! (Use a scanner to ask the user for two fractions and what to do with them (add, multiply, etc.)
	5. If you cannot figure out how to implement the comparable interface, or extend the Fraction class, just turn in a Fraction class with basic constructors and as much functionality as you know how to implement. Good luck!