COP3330C Module 5 Graded Assignment

In this assignment we will create and implement a generic interface to dispatch appointment reminders. We will also use the Queue interface from Java Collections to act as a primitive message queue (refer to the practice exercise for more information on message queues).

Follow the steps specified here:

- Include the generic dispatch package from the practice exercise (modify the package statement)
- Cut the Contact and Appointment classes from the AppointmentApp.java file and implement as separate public classes in the application's calendar package.
- Cut the ReminderPreference enumerated type from the AppointmentApp.java file and implement as a separate public enum.
    - Your CalendarReminder should already be implemented as a separate public interface in the package.
- Create a new public class which represents a reminder. This class should include three private members representing the **reminder text** (String) the **date and time of the reminder** (as a ZonedDateTime) and the **contact**. Include an overloaded constructor and accessors for this class.
    - Note that the Appointment class will continue to store only the ZonedDateTime reminder value as a member variable, do not change this. When you build the reminder you can use that value (via the accessor) to instantiate the Reminder object along with the text and contact using the overloaded Reminder class constructor.
- Modify the CalendarReminder interface as follows (see red text):

```
public interface CalendarReminder {
    // build a reminder in the form of a formatted String
    public Reminder buildReminder(Appointment appt);

    // send a reminder using contact's preferred notification method
    public void sendReminder(Reminder reminder);
}
```

- Implement the Reminder dispatcher in your AppointmentApp class. Determine the reminder preference in the sendReminder method (remove this logic from the checkReminderTime method if you used my solution from Module 4) and then call the dispatch method from the sendReminder method. Test to verify correct operation, then comment out the call to the dispatch method and implement a lambda to dispatch the Reminder as demonstrated in the practice exercise.
- Something to think about which may become relevant in upcoming assignments: why is it useful to include the contact and reminder time in the Reminder class? Is there a better way to represent this information?

Be sure to remember to add (accurate) ID headers for all of your new files.
Submit your solution to the GitHub classroom repo created when you accepted the assignment

Sample output is shown below; you should test with both SMS and EMAIL notifications and display the queue length as reminders are dispatched.

Sending the following SMS message to John Smith at (904) 555-1212
current queue length is 1
Sending the following email message to Sally Rogers at SallyRogers@email.com
current queue length is 2