CS410J Project 3: Pretty Printing an Appointment Book (10 points[1])

In this project you will continue to develop your appointment book application. The appointments in an appointment book will be sorted and you will add the ability to "pretty print" the appointment book.

Goal: Learn about Java's facilities for sorting and working with dates.

For this project you will have to add the following functionality to your appointment book application:

- ~~An `Appointment`'s begin and end times will now be represented by instances of `java.util.Date`. You should override `AbstractAppointment`'s `getBeginTime` and `getEndTime` methods. Your `getBeginTimeString` and `getEndTimeString` methods should now return strings that are formatted using `DateFormat.SHORT`.~~
- An `AppointmentBook`'s `Appointments` should be sorted chronologically by their beginning time. If two appointment begin at the same time, they should be sorted chronologically by their ending time. Otherwise, the two appointments should be sorted lexigraphically by their descriptions. To facilitate sorting, your `Appointment` class should implement the `java.lang.Comparable` interface.
- Class `PrettyPrinter` that implements the `AppointmentBookDumper` interface creates a nicely-formatted textual presentation of an appointment book. I suggest that you make use of `DateFormat` to make the dates look nice. Your pretty printer should include the duration of each appointment in minutes.
- Class `Project3` that has all of the functionality of the previous project plus the ability to pretty print an appointment book. `Project3` should have the following command line interface:

```
usage: java edu.pdx.cs410J.<login-id>.Project3 [options] <args>
  args are (in this order):
    owner                    The person whose owns the appt book
    description              A description of the appointment
    beginTime                When the appt begins
    endTime                  When the appt ends
  options are (options may appear in any order):
    -pretty file             Pretty print the appointment book to
                             a text file or standard out (file -)
    -textFile file           Where to read/write the appt book info
    -print                   Prints a description of the new appointment
    -README                  Prints a README for this project and exits
```

  The appointment book should be pretty printed **after** the new appointment has been added.

  The date and time format on the command line is no longer 24-hour time. An example of the date and time[2] would be `01/02/2016 9:16 pm`.

[1] 9 for code, 1 for POA

[2] You might want to use a `java.text.SimpleDateFormat` to parse the date and time.

Error handling: Your program should exit "gracefully" with a user-friendly error message under all reasonable error conditions. Examples of such conditions include

- Something is missing from the command line or there are extraneous command line arguments.
- The format of the day or time is incorrect.