Assignment #4:

Inheritance, Interfaces & Polymorphism

Requirements:

- Implement a superclass `Appointment` and subclasses `Onetime`**,** `Daily`**, and** `Monthly` for onetime appointment, daily appointment, and monthly appointment. Package java.time is allowed.
- An appointment has a **description**, a **year**, a **month**, and a **day**
  E.g.,
  Description: Dental Appointment
  Year: 2020
  Month:12
  Day: 8
- Write a method
  getYear(void): int
  that returns the appointment year
- Write a method
  getMonth(void): int
  that returns the appointment month
- Write a method
  getDay(void): int
  that returns the appointment day
- Override the method
  toString(void): String
  that returns the appointment description
- Write a method
  occursOn(int year, int month, int day): boolean
  that checks whether the appointment occurs on that day.
  For a **daily** appointment, you must **override** this method to check whether the appointment occurs on the **everyday later** than the appointment date in the given year.
  For a **monthly** appointment, you must **override** this method to check whether the appointment occurs on the **same day** for the **following months** within the given year.
- Implement a `AppointmentTester` class which can construct different instances of daily, monthly, and onetime appointment.
- Write a method
  addAppointment(int year, int month, int day, String description, String type): Appointment
  give the user the option to add new onetime/daily/monthly appointments by entering the year, month, and day of the appointment as well as the description of the appointment by specify the occurrence frequency of appointments
  e.g. “O” for creating onetime appointment, “D” for daily appointment, and “M” for monthly appointment.
- Then fill an ArrayList of Appointment objects with a mixture of appointments.
- Write a method
  displayAppointment(int year, int month, int day, ArrayList appointment):void
  have the user enter a date then display all appointments occur on that day and the type of occurrence frequency is onetime, daily, or monthly.

- Write a method
  displayAppointment(int startyear, int startmonth, int startday, int endyear, int endmonth, int endday):void
  have the user to enter the start date and end date then show all appointments in this period of time.

  (extra 1%)
- Write a method
  deleteAppointment(int year, int month, int day, ArrayList appointment):void
  give the user the liberal to delete an appointment or appointments occur on that day or within the year if the type is daily or monthly.
  e.g. if the appointment is onetime, delete the single appointment on that day;
  if the appointment type is daily or monthly, then delete the following appointment within the year.

Submission: Your Eclipse project is named `yourStudentID_HW4.`The project is submitted as `yourStudentID_4.zip`. Submit via eCourse. No other submissions will be graded.

Academic dishonesty: You may not do work for another student nor may any student copy or plagiarize someone else's work. Severe penalties will be imposed on all parties involved.

Deadline: Saturday, January 9, 2020 (end of the day)