

Lab 05

Date and Time Methods

Objective:

Write a class that will test dates and times inputted by the user and determine whether or not it is valid. This will focus on the usages of methods to organize code.

Lab Solution

Requirements:

- Functionality. (80pts)
 - No Syntax, Major Run-Time, or Major Logic Errors. (80pts*)
 - *Code that cannot be compiled due to syntax errors is nonfunctional code and will receive no points for this entire section.
 - *Code that cannot be executed or tested due to major run-time or logic errors is nonfunctional code and will receive no points for this entire section.
 - Set up the project (10pts)
 - Download the [driver](#) file and include it in your project's source folder (SRC for most IDE's).
 - DO NOT ALTER THE DRIVER
 - Create a class and name it exactly **DateAndTimeTester** with NO main method.
 - If a main method is accidentally added, then remove the method from the class before submitting the code.
 - Create the following methods in the **DateAndTimeTester** class (70pts)
 - For reference MM/DD hh:mm is Month / Day Hour : Minute respectively.
 - run (8pts): This method returns nothing and takes no parameters. This is called by the driver and should handle all of the input from the Scanner and dialog for the user.
 - isValid (10pts): returns true or false if a given String has the correct date and time. The String parameter should be formatted "MM/DD hh:mm" This method should call the methods isValidDate and isValidTime to determine this.
 - Also, the MM, DD, hh, and mm could either be single or double digits. Meaning that "09/09 01:01" is equally as valid as "9/9 1:1".
 - isValidDate (10pts): returns true or false if a given String has a correct date. The String parameter should be formatted "MM/DD" and should use the method getMonth and getDay to determine the date's validity. Also assume February only has 28 days.

- isValidTime (10pts): returns true or false if a given String has a correct time. The String parameter should be formatted “hh:mm” and should use the getHour and getMinute to determine the time’s validity. Valid times are from 1 to 12.
 - getMonth (8pts): returns an integer value representing the month for a given String. The String parameter is expected to be formatted “MM/DD”.
 - getDay (8pts): returns an integer value representing the day for a given String. The String parameter is expected to be formatted “MM/DD”.
 - getHour (8pts): returns an integer value representing the hour for a given String. The String parameter is expected to be formatted “hh:mm”.
 - getMinute (8pts): returns an integer value representing the minute for a given String. The String parameter is expected to be formatted “hh:mm”.
 - HINT: For each one of the methods pay close attention to *how the data is separated (delimited)*. For instance, we can see that Month and Day are separated using a ‘/’, and finding the index of this character can help us to separate the Month from the Day.
- Coding Style. (10pts)
 - Readable Code
 - Meaningful identifiers for data and methods.
 - Proper indentation that clearly identifies statements within the body of a class, a method, a branching statement, a loop statement, etc.
 - All the above must apply for full credit.
 - Comments. (10pts)
 - Your name in the file. (5pts)
 - At least 5 meaningful comments in addition to your name. These must describe the function of the code it is near. (5pts)

Example Dialog:

```
Enter a date and time (MM/DD hh:mm) and I
will determine if it is valid
06/22 3:00
The date and time is valid!
Would you like to exit? Type "quit" to exit
or press [ENTER] to continue
```

Enter a date and time (MM/DD hh:mm) and I will determine if it is valid

9/31 12:00

The date and time is not valid

Would you like to exit? Type "quit" to exit or press [ENTER] to continue

Enter a date and time (MM/DD hh:mm) and I will determine if it is valid

12/08 13:00

The date and time is not valid

Would you like to exit? Type "quit" to exit or press [ENTER] to continue

quit

Good bye

Solution Tests:

1. Is your name written as a comment in all source files?
2. Does the solution compile (no syntax errors)?
3. When entering "06/22 3:00" does the program print that it is valid?
4. When entering "9/31 12:00" does the program print that it is invalid?
5. When entering "12/08 13:00" does the program print that it is invalid?

Lab Report

1. Create a section named "Problem" and describe this lab's problem in your own words. (10pts).
2. Create a section named "Solution Description" and describe how the code solves the problem in your own words. (10pts).
3. Create a section named "Problems Encountered" and describe the various syntax, run-time, and logic errors that were encountered while implementing the solution. (10pts).
4. What is the purpose of a parameter for a method.
5. If a variable is declared in a method, then can it be used outside of that method?
6. If a method returns nothing, then what should its return type be?
7. Given the following code snippet, there appears to be some kind of syntax error with this method. Where is the error and how can it be fixed?

```
public void printToConsole(String msg)
{
    System.out.println(msg);
    return msg;
}
```

8. Given the following code snippet, there appears to be some kind of syntax error with this method. Where is the error and how can it be fixed?

```
public void printStars(double numberOfStars)
{
    if(numberOfStars < 0);
    {
        return;
    }
    for(int i=0;i<numberOfStars;i++)
    {
        System.out.println("*");
    }
}
```

9. Given the following code snippet, there does not appear to be a syntax error, but there is a strange logic error where the method always returns 0. Explain the logic error and how best to fix it.

```
public double oneThird(double number)
{
    return (1/3)*number;
}
```

10. Given the following code snippet, are there any kind of errors? If so, name the types, where, and how this code can be fixed. Otherwise, what does this code *exactly* print to the console?

```
public boolean isValid(int value)
{
    if(value >= 0)
    {
        return true;
    }
    else if(value < 0)
    {
        return false;
    }
}
```

Finally:

Upload the source code (.JAVA File Extension) and written lab report (.DOC, .DOCX, or .PDF file extension) to the CSCE Dropbox.