

Homework #2

due Tuesday October 2nd, 13:20pm

For this assignment you will write two separate Java programs. One will be a textual output oriented room reservation processor. The other will be a generalization of the clock animation applet from the book.

Undergraduate students may work in teams of 2 for this homework. Use AFS to give both of you permission to access a homework directory *before you start work*. Do not share passwords. Make sure to indicate in comments on the files that both of you are authors. Groups larger than two, or which include graduate students are not permitted. If you have problems getting group permissions to work, see me.

1 Room Reservation

You are provided with a driver for a room reservation system that uses the following classes in a package `schedule`:

Time An immutable class representing a time of day (24 hour clock). It should have `equals` defined and should be `Comparable`.

Period A mutable class representing a period of time; it has a start time and a duration.

Day A enumeration class representing days of the week.

Room An immutable class with object-identity (no `equals` overriding) with a name.

Reservation A mutable class representing a room reservation.

You should document the classes and indicate any invariants, using the techniques and conventions described in the book.

You need to provide a good design for these classes (this homework will not list required methods and fields), but in order to provide some constraints, you are provided with a driver. Your classes must work correctly with the provided driver. We will also provide the expected output on one test file. During grading, there will be other test cases as well.

1.1 Instructions

You should put all the Java files (except the driver) in a package “`schedule`” in a project (or directory) named `Part1` in the `homework2` directory of your AFS volume.

You may use any of the classes in `java.lang`, especially `System`, `Integer` and `Float` (the latter two classes are used only for their `parse...` functions). In addition, your code will need to use `BufferedReader` and `IOException` from the `java.io` package. You should *not* use any classes from `java.util`, such as `Hashtable`, `LinkedList` or `ArrayList`.

If you use an IDE such as Eclipse, we suggest you begin with the driver and work “down,” because the IDE will suggest stubs for functions as you use them in your code. (In Eclipse, the suggestions show up as combination light-bulb/stop-sign markers in the margin.)

Please use Java conventions for method names and fields: use `mixedCase` rather than `under_scores` to separate words, except for `CONSTANT_NAMES`. If you wish, you may start field names with an underscore (and then please be consistent).

2 International Date/Time

The textbook defines a digital clock animation applet. Please extend it *with inheritance* in a new class `DateClock`. It should accept an applet parameter `timezone` that should be a valid `java.util.TimeZone` ID (see the JavaDoc).

It should display a digital clock, together with the date in the default locale `DateFormat`. The date should appear below the time in the applet. Make sure your applet works correctly when the day changes!

3 Files

Please put your room reservation files in a subdirectory `Part1` (this can be an Eclipse project), with the ADT files in `Part1/schedule`. The driver that we provide is available under AFS as `/afs/cs.uwm.edu/users/classes/cs552/src/homework2/Driver.java`. The same directory will also hold sample input and output.

The code from Chapter 4 (and in particular the `DigitalClock` class is available in

```
/afs/cs.uwm.edu/users/classes/cs552/src/book/Chapter04
```

Please import the necessary files into a subdirectory (or Eclipse project) called `Part2`. Your internationalized clock/date applet should be called `DateClock.java`. Please have an `html` file which loads your applet to show the date and time in Japan (but not in Japanese!).

Both projects must be runnable from within their respective directories. In other words, on a grid, it must be possible to

```
grid.cs: cd homework2
grid.cs: cd Part1
grid.cs: cs bin (optionally)
grid.cs: java Driver
...
grid.cs: cd ../Part2
grid.cs: appletviewer DateClock.html
```

As usual, all work must be completed by 12:30pm, Tuesday, October 2nd, 2007.