Tool Rental Application - Technical Specification

This technical specification details the design for the tool rental application described in the functional specification document. Consulting both documents will be required to complete this work.

The design consists of the five main classes listed below. These main classes should not directly depend on the Spring Framework, for increased flexibility of integration into other systems, but can require dependencies such as Apache Commons components.

Along with these main classes, appropriate unit tests should be included. The Spring Framework can be used in these tests as well as a development and testing harness.

Main Classes

Tool

A simple class that represents a rentable tool, with attributes corresponding to columns in the two tool data tables contained in the functional specification document: tool code, tool type, brand, daily charge amount, weekday charge, weekend charge, and holiday charge. This class must have a setter and getter for each of these attributes, a no-argument constructor, and a copy constructor.

ToolData

Implements lookup of Tool instances by tool code. On construction, it should load the tool data from CSV file resources, join the two tables on tool type, create a Tool instance for each resulting row, and store those Tool instances in a thread-safe map keyed by tool code. Loading the data from CSV will make it easier to modify or add tools in the future. After looking up a tool code, instead of returning the Tool instance from the map, it will make a copy to return using the Tool class copy constructor, to prevent modification of the Tool instance in the map.

Holiday

An enum with an element for each holiday listed in the functional specification document. The enum elements should have a method which tests whether or not a specified date is an occurrence of that holiday. There should be a static method on the enum which tests whether or not a specified date is an occurrence of any of the holidays defined in the enum.

RentalAgreement

A simple class that represents an agreement covering the rental of a single tool, with the twelve attributes described in the functional specification document, with setters and getters for each of these attributes. This class must have a method that prints, to a specified PrintStream, the values of its attributes in the format described in the functional specification document, and also a no-argument print method that uses the former print method to print to stdout. The ability to specify the printing destination will allow for future flexibility. The standard Java text formatters NumberFormat and DateTimeFormatter should be used to format currency amounts and dates respectively in order to meet the functional specification.

Checkout

A builder for RentalAgreement instances, containing the business logic of the tool rental application. This class should follow the builder pattern, with setters for four input attributes: tool code, rental day count, discount percent, and checkout date. These setters must throw exceptions with informative messages for invalid argument values. The Checkout class will use the ToolData class to look up the price attributes of the specified tool code, the Holiday class to determine if rental dates qualify as holidays, and the Java LocalDate class for date calculations. Following the functional specification, this class will create and return a RentalAgreement instance and set its attributes: the input attributes, the corresponding tool type attributes, and the calculated value attributes.