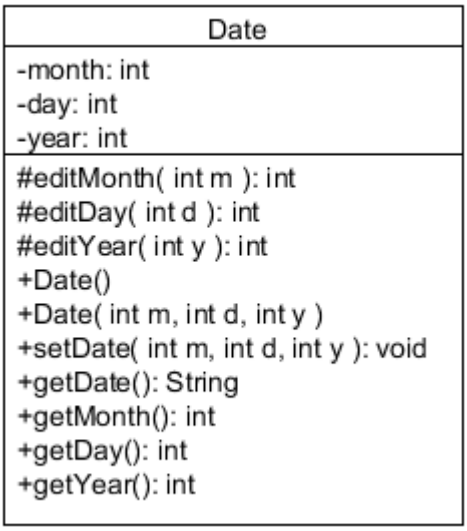
Johns Hopkins University
CTY*Online*

Advanced Java Programming

Assignment 1

The purpose of this assignment is to help you get familiar with using inheritance. One of the advantages of inheritance is that it enables programmers to reuse existing classes…and to extend those classes to do more things…without having to see the implementation details or duplicate code. It is sufficient to know what the existing class does and how to interface to it using the class's visible methods.

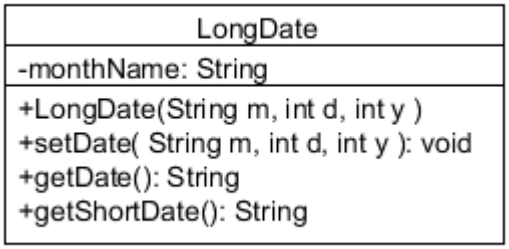
The following class models some basic functionality associated with dates, and already exists. The compiled class file is available in the Course Downloads folder.



As you can see, the Date class models a date using integer month, day, and year values. It has a default constructor that sets the date values to zero, and a parameterized constructor that takes specific values for the date. When a Date object is instantiated using the parameterized constructor and when the values of the date data members are set using the setDate() method, the parameters are edited using three protected methods. The month parameter must be 1-12, the day parameter must be 1 to 28-31 (the class code checks this against the

specified month), and the year parameter must be greater than 1. If the date parameters are invalid, the user is prompted to re-enter them until valid values are entered. The edit methods return integer values that are valid parameter values and which are used to initialize the month, day, and year members. The getDate() method returns the date as a String type, formatted as mm/dd/yyy. To get a feel for the Date class edit methods you may want to write a simple program that creates some Date objects with invalid parameters.

Using inheritance, develop the code for a class called LongDate, that uses the name of a month as its data attribute. The LongDate class shall also have at least the methods indicated below:



The getDate() method shall return the date in long form (e.g., May 1, 2011) and the getShortDate() should return the date as mm/dd/yyyy. The LongDate class shall edit the date parameters when the parameterized constructor and the setDate() methods are used. Edit criteria are the same as for the Date class, and the month name (January, …, December) must begin with an upper case letter. If invalid parameters are specified, the user shall be prompted to re-enter the information until it is correct. The Input class methods can be used to prompt the user for input. The compiled code for the Input class methods can be found in the Course Downloads folder, and a tutorial on how to use these methods is provided in the Unit 1 lecture links.

Please note that the inherited Date class performs editing for the integer month, day, and year values, so the LongDate class should not take responsibility for doing those edits…and it should not be necessary for the LongDate class to call the Date class edit methods directly.

Write a program that creates an array of type Date, and populates the array with several Date objects and LongDate objects. The program should loop through

the array and display the output of the getDate() method for each object in the array.

Please comment your code and make good use of indentation and white space. Submit the source code for the LongDate class and for your program.