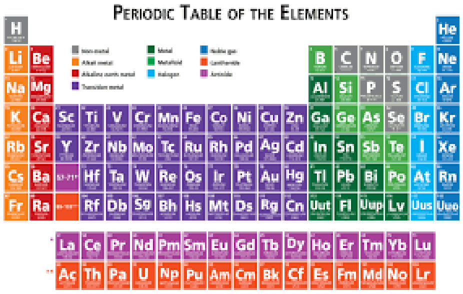
CST 183 Java Programming
Object Array Program Assignment (30 pts)


Periodic Table of the Elements
H He
Li Be B C N O F Ne
Na Mg Al Si P S Cl Ar
K Ca Sc Ti V Cr Mn Fe Co Ni Cu Zn Ga Ge As Se Br Kr
Rb Sr Y Zr Nb Mo Tc Ru Rh Pd Ag Cd In Sn Sb Te I Xe
Cs Ba Hf Ta W Re Os Ir Pt Au Hg Tl Pb Bi Po At Rn
Fr Ra Rf Db Sg Bh Hs Mt Ds Rg Cn Uut Fl Uup Lv Uus Uuo
La Ce Pr Nd Pm Sm Eu Gd Tb Dy Ho Er Tm Yb Lu
Ac Th Pa U Np Pu Am Cm Bk Cf Es Fm Md No Lr

Purpose

To create a Java program that uses objects, array concepts, and file processing.

Specifications

Write a Java program that will aid a chemistry student with the periodic table of elements.

The text file named periodic.txt includes the basic information for each chemical element including the element symbol (1 or 2 characters), the element name, the element number, and the atomic weight. Design an Element class to store information for one element. Your program should open the input file, read in a record and use the information to instantiate an element object. Then the program should store the object in an array that represents the periodic table.

Once you've processed the input file, your program should present a simple menu to the user. Allow the user to search the periodic table using either the element code or the atomic number of the element. When you match the given user request, provide all of the information on the element including the name, code, atomic number, and atomic weight.

Include methods as detailed here. (These are in addition to the constructor, accessors, and mutators detailed below). Consider the main tasks that are being performed and the data "flowing" between actions. Be sure to include error checking to return an error message to the user if either the element number or element code is not found in the data set. Finally, you will need to use the String comparison operations taught in previous chapters for this program.

LoadArray	**Primary responsibility**: Loads periodic table array with records. HOW: Pass the address of your element object array to the function. You can use this method to read in the entire array or handle one element each time it is called.
DisplayMenu	**Primary responsibility:** Displays the program menu. HOW: Display a menu that allows the user to search the periodic table array using either the element code or the atomic number. Return the menu selection to the calling method.
CodeSearch	**Primary responsibility:** Searches the element object array by element symbol(code). How: Pass the address of your element array and the symbol for the desired element. Set up a loop to process the array, searching for the symbol. If found, return the array

	index where the element was found. If not found, return -1. Use the linear search strategy that was covered in class.
NumberSearch	**Primary responsibility:** Searches the element array by element number. HOW: Pass the address of your element array and the element number for the desired element. Set up a loop to process the array, searching for the element number. If found, return the array index where the element was found. If not found, return -1.

In addition to the above functions, you should display the result of the search (e.g. whether the element was found or not) and the associated information.

Design Requirement

Use planning documentation tools (e.g. flowchart, pseudocode, etc.) as needed to design your program.

Consider our object discussions and plan on defining attributes for each of the input fields. Specifically:

a) Field 1 – element symbol: String
b) Field 2 – element name: String
c) Field 3 – element number: int
d) Field 4 - element atomic weight: double

You will need to provide accessor and mutator methods (setters and getters) for each of these attributes. You will also need to define a constructor that will be used to instantiate an element object and initialize the fields based on the element record read from the data file.

You can either use a single Element Class file (with embedded main() method) or separate files; an Element Class file and separate Periodic Class file that has the main() method.

Final Deliverables

Deliver the following as your final product:

- Cover page with assignment name, student name, and list of attachments
- Hierarchy chart that shows method structure.
- UML diagram for the Element class.
- Printed copy of source code
- Output screenshot(s)
- Upload source file(s) to your Desire 2 Learn (elearning) folder.