

Building a simple Note App

Purpose: Build a driver class and enhance a Notebook class with several useful ArrayList methods.

Learning Objectives:

- Refresher on ArrayList
- Practice good testing – especially boundary testing;
- Using the console for output

Instructions:

A. Using a class with an ArrayList.

Here is the Notebook.java program shown in class:

```
import java.util.ArrayList;

public class Notebook {
    private ArrayList<String> notes;
}
```

Add a getter and setter for notes. Make sure that Notebook has a getter and setter for notes. Create a method numberOfNotes() that will return the size of the ArrayList. Make a driver program, NotebookTest, to test your methods.

B. Adding useful methods to Notebook.

After part A, we should be able to:

- get the ArrayList notes using our Notebook object's getter;
- set the ArrayList notes by using our Notebook object's setter and sending it an ArrayList of Strings;
- get the size of the ArrayList from our Notebook object.

But let's continue to enrich our Notebook class with the following methods:

- addNote() will add an individual note to the ArrayList
- deleteNote() will delete a note using its position in the ArrayList
- getNoteNumber() will get the position of a note based on its value, e.g., in what position is note "Buy milk"?
- getNote() will get the note text for a given position
- setNote() will set (update) the note text for a given position

C. Preventing duplicate notes.

Modify your "add note" method above so that it will not add a duplicate value to the ArrayList..

*The following sections deal with methods that can reposition notes in the notebook. The user should specify which note to reposition **by its text**, and not its index position in the ArrayList.*

D. Moving notes up and down.

Imagine if your notes were in priority order. It might be nice to select a note and then move it up (prioritize) or down (deprioritize). Create a moveNoteUp() method that will move a note up one spot, and a corresponding moveNoteDown() method that will move a note down one spot.

E. Moving notes multiple times

In your NotebookTest driver program, use a For loop to move a note up for a specified number of moves (which you will store in a variable). Set the value of this variable to 2 and test your For loop. Use the method you created in step D.

Make a corresponding For loop to move a note down a specified number of times.

F. Moving notes to the top or bottom of the list.

Finally, make two last methods to move a list item to the top of the list (moveNoteToTop), or the bottom of the list (moveNoteToBottom).