Data Structures - Fall 2013
Project 4: Employee Inquiry

[v1.0 10/28/2013 10:12pm]

Objective: practical application using a random access file.

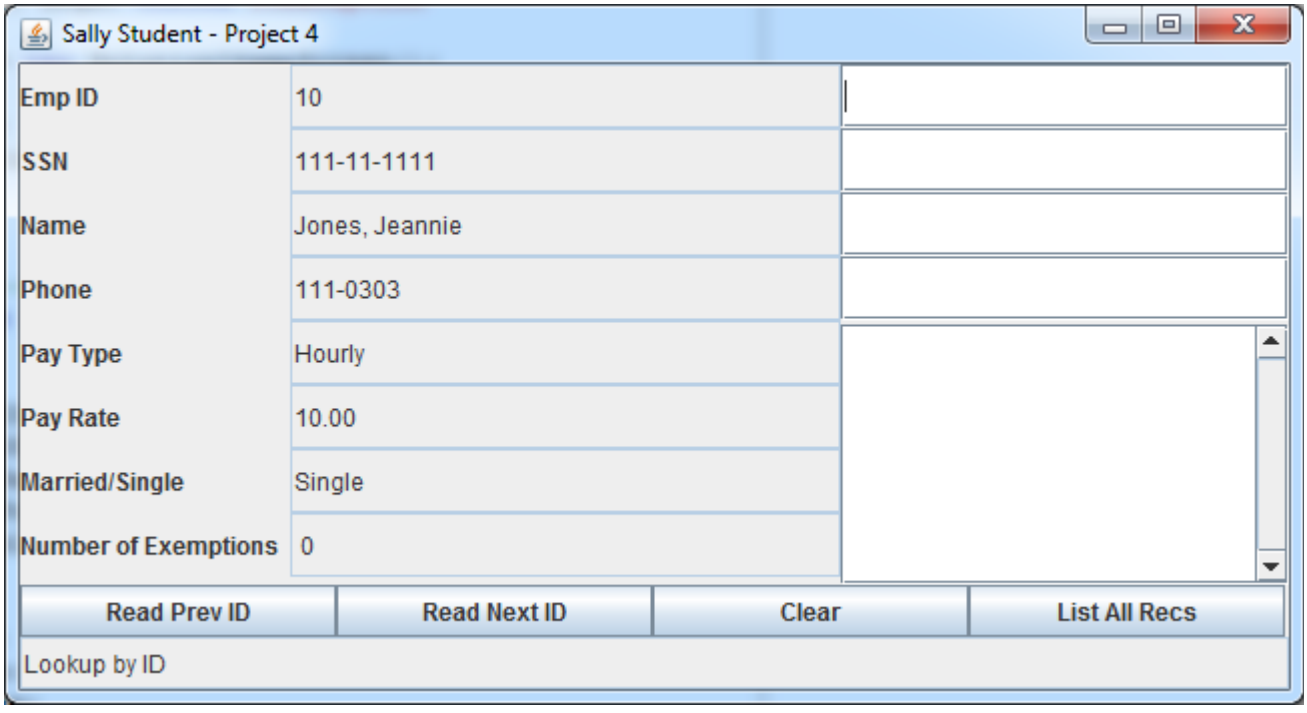
Part 1: To start you will need to write a program that will read the text file I've provided and create the random access file with the following fields. This is a "one time" program and is not part of the gui inquiry.

Starter input file (ascii text)			
columns		description	specs
01	09	ssn	
10	34	name	last, first
35	41	phone	
42	42	pay type	s=salary, h=hourly
43	53	pay rate	two decimals
54	54	marital status	s=single, m=married
55	56	dependents	

Random Access File		
position	desc	storage format
1	ssn	four-byte integer
2	name	ASCII, character array
3	phone	four-byte integer
4	pay type	ASCII, single character
5	pay rate	double floating-point
6	marital status	ASCII, single character
7	dependents	four-byte integer

note: emp id is not stored in the record. It is the relative record number (physical record number) in the file, beginning with 1.

Part 2: I have provided a starter program that presents the following gui:



The center column contains the record information fields, and are protected.
Read Prev/Next ID navigates to next (or previous) record from the current file position.
Clear button clears everything in the display and repositions cursor to emp id with that fields message.
List All Recs causes console (System.out.printf( )) output. See sample posted on assignment page.

General specifications:

1	The screen navigation expected will be demonstrated in class. Pressing the enter key on an empty input field will cause the cursor to advance to the next field, wrapping around from phone back to emp id. The initial position will be emp id. (starter program provided already does this)
2	The input request fields are the four unprotected text fields in the upper-right area.
3	When an input field gets focus because you advanced there by using the enter key, you should display a message indicating the allowed action in the message area text field at the bottom of the screen. “Lookup by ID”, “Lookup by SSN”, “Lookup by Name (partial match allowed)” , “Lookup by Phone”
4	Numeric fields (emp id, ssn, phone): When something is entered into a numeric field (emp id, ssn, phone) you should strip it of any non-numeric characters and use the remaining numbers for the inquiry. Note: should be a valid length. If no numbers remain after stripping non-numeric characters you should display the original entry and notify the user that it is an invalid request using the message text field at the bottom of the screen
5	Name field: The name field should allow a partial match anywhere within the name field, and not be case sensitive. Spaces used in the inquiry request would be meaningful, so don't remove them. For example, to search for a first name that begins with `"an"`, and not just any name that contains those characters anywhere within the name, you could search for `" an"` with a leading space.
6	Pay type and marital status fields: when displaying a record that doesn't contain a valid value for pay type or marital status, display “(not on file)” for that field.
7	Error handling and input validation are not required other than as specified.
8	Inquiry specifics: Inquiry by Emp ID is a direct read of the record in the file. A 5 entered would cause the 5th record to be read by computing the displacement from the beginning of the file and going directly to it. Inquiry by the other fields (SSN, Name, Phone) is to be a linear search of the file beginning with the first record. Inquiry by SSN and Phone: are an exact match. In the linear search, when you get to that record you stop. If you get to the end of file without finding it, you display a message to that effect in the message line at the bottom. Inquiry by Name is a partial match, therefore you will read all records in the file from beginning to end. If you only find one match you display that record in the information fields (center column of the screen). If you find more than one record that matches the specification, you display the record number and name in the text area (lower right portion of screen) as shown in the screen shot below. If no records are found that match, report it as such in the message line at the bottom. Results are to be formatted as shown in the examples.
9	Note: I shouldn't have to say this, but... you will not be “loading the file up into an array” to do the searches. The objective of this project is to learn to work with random access files. You are not to use any alternate data storage techniques (such as arrays) or api class members that circumvent the objective.

10	Inquiries that don't result in a match: When an inquiry doesn't result in a match, clear the search input field (leave the cursor there), clear all the record information fields (if anything is there from a previous inquiry), and display a message such as "Invalid request: xxxxxxxxxx...x".
11	Read Prev/Next ID: display appropriate message when you are at beginning or end of file and can't read again in that direction. Do not "wrap around".

The following screens show inquiry results. ("anna", and " anna" with a leading space)