Arrays of Structures - basics

This exercise focuses on making you practice using data structures stored in automatically allocated arrays.

• **a01.h**	will contain the headers (aka declarations) of our functions.
• **a01.c**	will contain the definitions of all the functions we will implement.
• **main.c**	will contain the main function used to provide the user with a small text-based menu to test interactively all the functions.

User-defined data structure

Declare a structure *struct record_s* containing a string name (an array of *char* of max size 20) and a long SSN. This declaration will go in your header file (*a01.h*) which will be included by both the implementation file (*a01.c*) and the main file (*main.c*) so that they can instantiate variables of type *struct record_s*.

Interactive, menu-driven, main function

You will declare a local variable *mydata* in the *main* of your application which will be an array of 5 elements of type *struct record_s*.
The *main* function will offer a text-based menu to the user allowing him or her to;

(1) enter some data for each of the 5 employee records stored in the array *mydata*
(2) display the contents of each employee record stored in *mydata*
(3) display an arbitrary element of the array which index is between 0 and 4 (perform some boundary checking here to avoid boundaries bugs).

Each of these options will call the appropriate function described in the following subsection.

Functions to implement

The following functions need to be implemented and used from the *main*, don't forget to add their prototype in the header file;

void scanData (struct record_s data[], int size);

- Scans from the user information (name and social security number) to store in the array of structures *data* of size *size*

void displayDataAt (struct record_s data[], int i);

- Displays the content of the employee record at index *index* in the array *data*.

void displayData (struct record_s data[], int size);

- Displays the content of the whole array
- Use the displayDataAt function to do so