CS118 Program
Find Name

Write a Python program that does the following (in order):

- Creates three lists: `ID`, `FirstName`, `LastName`.
  These should <u>each</u> be hardcoded with at least three string values.
  The `ID` values must be <u>strings</u> of exactly 5 digits.

- Displays all of the available `ID`s for the user
- Collects an ID string from the user;
  - If the string is less than five characters, pads it with enough leading zeros so that it is five characters long
  - If more than five characters, changed to contain only the first five characters
- Calls the programmer-defined function `find_name()` with appropriate arguments (see below);
- Collects the return values from the function;
- Prints out the name of the person located, or an error message if the ID provided is invalid

It is not necessary for the program to repeat. It is not necessary to validate user input since the function will in essence be doing that.

Along with the main program, write a Python function called `find_name()`. The purpose of the function is to return a list with the first and last name for a given ID string. The function has four parameters - a list of ID strings; a list of first names; a list of last names; and a search number string. None of the lists may be assumed to be sorted, and all are the same length.

The function should look for the search string in the list of `ID` strings, find the position of that search string, and return (**<u>as a list</u>**) the first and last name which are at *that same position* in the `FirstName` and `LastName` lists. If the given search string is not in the list of ID strings, have the function return an empty list.

The function may not define the lists, and must not change the lists in any way. The function must not interact with the user in any way - it should only interact with the calling program. This means that all input requests from the user and all output to the user (e.g. printing to the shell) must occur in the program, not in the function.

```
>>>
Available numbers:
00001
00002
00003
00004
00005
00006

For which number do you wish to search? 3
ID 00003 is Brian Bixby
>>>
```

```
>>>
Available numbers:
00001
00002
00003
00004
00005
00006

For which number do you wish to search? 0000567
ID 00005 is Abe Allman
>>>
```

```
>>>
Available numbers:
00001
00002
00003
00004
00005
00006

For which number do you wish to search? 123456789
I'm sorry - ID 12345 is not an acceptable option.
>>>
```