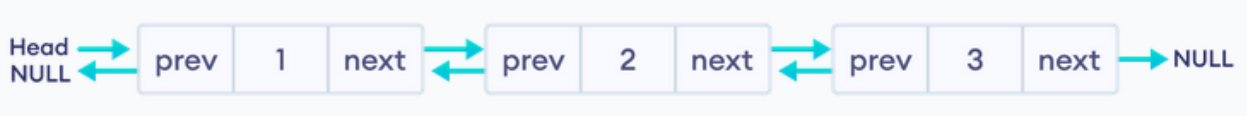
One Table Demographic Database

Summary
Design an application to read demographic records from a file into objects. Assemble the objects into a double linked list. Print the list of records into a table with the fields rearranged such that the last name is first.

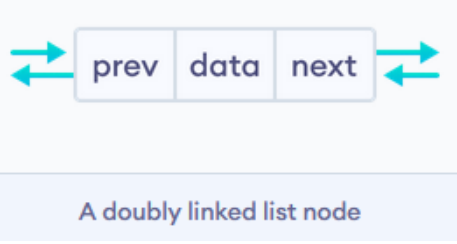
Of course, there is much more that can be done with a database. But, we'll confine it to this.

Database (relational)
Table:
A relational database is made up of multiple tables that have relationships to each other. For this assignment, there is only one table. The data for this single table, will be read from the provided file: demographicInfo.dat
Record:
Each row in the provided file is considered a database record. Tables may have multiple records. Each record will be read from the file into a DemographicInfo object.
Field:
Database records may have multiple fields. The fields in these records will be stored in the data members (attributes) of the DemographicInfo objects.

Doubly Linked List (see https://www.programiz.com/dsa/doubly-linked-list)
Like the linked list in the textbook, each node has a link to the next node. A doubly linked list adds a link to the previous node. This facilitates easy traversal of the list both forward and backward.



Each node in the list will contain 3 pieces of information:



Of course there are the two links – previous and next. The data in this case will be a DemographicInfo object, containing the individual fields read from a record that was input from the provided file.

The provided file data – single database table,…

Jacob,Smith,27,male,single,software engineer,123 Main St,San Francisco,CA,94105
Ashley,Hernandez,32,female,married,accountant,456 Elm St,Miami,FL,33131
Liam,Jones,45,male,married,construction worker,789 Oak Ave,Seattle,WA,98101
Emily,Chen,29,female,single,graphic designer,1010 Broadway St,New York,NY,10003
Michael,Lee,33,male,married,dentist,2345 Main St,Boston,MA,02108
Samantha,Wilson,26,female,single,marketing assistant,678 Maple Rd,Los Angeles,CA,90001
Daniel,Kim,39,male,married,IT consultant,890 Pine St,Chicago,IL,60601
Sarah,Johnson,31,female,single,nurse,1212 4th Ave,Denver,CO,80203
Jason,Davis,41,male,married,sales representative,345 Cedar Blvd,Austin,TX,78701
Laura,Brown,25,female,single,student,567 Park Ave,Philadelphia,PA,19103
Robert,Jackson,50,male,married,business owner,789 Madison St,San Diego,CA,92101
Olivia,Martin,29,female,single,journalist,987 Hillside Rd,Portland,OR,97201
William,Garcia,35,male,married,teacher,1234 Cherry St,Miami,FL,33131
Grace,Kim,24,female,single,web developer,5678 Vine St,San Francisco,CA,94102
Ethan,Martinez,28,male,single,architect,9012 Sunset Blvd,Los Angeles,CA,90046
Elizabeth,Rodriguez,33,female,married,accountant,3456 Park Ave,New York,NY,10016
Kevin,Nguyen,26,male,single,data analyst,7890 Pine St,Boston,MA,02108
Maria,Hernandez,40,female,married,homemaker,123 Spring St,Houston,TX,77002
Joshua,Lee,31,male,single,consultant,456 Main St,Chicago,IL,60601
Rachel,Kim,28,female,single,journalist,789 Hillside Rd,San Diego,CA,92101
Steven,Davis,39,male,married,engineer,1011 Cedar Blvd,Austin,TX,78701
Jessica,Smith,27,female,single,teacher,1213 4th Ave,Denver,CO,80203
David,Brown,45,male,married,lawyer,567 Park Ave,Philadelphia,PA,19103
Amanda,Jackson,29,female,single,artist,789 Madison St,San Francisco,CA,94102
James,Martin,35,male,married,accountant,987 Hillside Rd,Portland,OR,97201
Samantha,Garcia,24,female,single,programmer,1234 Cherry St,Miami,FL,33131
Isabella,Green,31,female,single,journalist,234 Elm St,Boston,MA,02108
Christopher,White,40,male,married,financial analyst,567 Pine St,San Francisco,CA,94105
Sofia,Martinez,29,female,single,graphic designer,678 Maple Rd,Los Angeles,CA,90001
Benjamin,Wright,26,male,single,software developer,890 Pine St,Chicago,IL,60601
Victoria,Brown,38,female,married,nurse,1212 4th Ave,Denver,CO,80203
Matthew,Taylor,35,male,single,journalist,345 Cedar Blvd,Austin,TX,78701
Audrey,Davis,29,female,married,HR manager,567 Park Ave,Philadelphia,PA,19103

Noah,Rodriguez,27,male,single,graphic designer,789 Madison St,San Diego,CA,92101
Maya,Patel,30,female,single,marketing manager,987 Hillside Rd,Portland,OR,97201
Samuel,Wilson,43,male,married,financial advisor,123 Main St,San Francisco,CA,94105
Charlotte,Lee,25,female,single,accountant,456 Elm St,Miami,FL,33131
Aaron,Adams,34,male,married,engineer,789 Oak Ave,Seattle,WA,98101
Chloe,Campbell,28,female,single,writer,1010 Broadway St,New York,NY,10003
Vincent,Mitchell,36,male,single,IT specialist,2345 Main St,Boston,MA,02108
Elena,Wright,31,female,married,project manager,678 Maple Rd,Los Angeles,CA,90001
Nicholas,King,29,male,single,marketing analyst,890 Pine St,Chicago,IL,60601
Isabelle,Scott,36,female,married,homemaker,1212 4th Ave,Denver,CO,80203
Adrian,Carter,42,male,married,businessman,345 Cedar Blvd,Austin,TX,78701
Caroline,Evans,27,female,single,web designer,567 Park Ave,Philadelphia,PA,19103
Alexander,Lee,32,male,married,pharmacist,789 Madison St,San Francisco,CA,94102
Grace,Perez,28,female,single,data scientist,987 Hillside Rd,Portland,OR,97201
Justin,Coleman,30,male,single,consultant,1234 Cherry St,Miami,FL,33131
Emma,Rivera,26,female,single,photographer,456 Main St,Chicago,IL,60601
Lucas,Nguyen,33,male,married,teacher,789 Oak Ave,Seattle,WA,98101

Example Output

last	first	age	sex	status	occupation	street	citystate	zip
Smith	Jacob	27	male	single	software engineer	123 Main St	San Francisco CA	94105
Hernandez	Ashley	32	female	married	accountant	456 Elm St	Miami FL	33131
Jones	Liam	45	male	married	construction worker	789 Oak Ave	Seattle WA	98101
Chen	Emily	29	female	single	graphic designer	1010 Broadway St	New York NY	10003
Lee	Michael	33	male	married	dentist	2345 Main St	Boston MA	02108
Wilson	Samantha	26	female	single	marketing assistant	678 Maple Rd	Los Angeles CA	90001
Kim	Daniel	39	male	married	IT consultant	890 Pine St	Chicago IL	60601
Johnson	Sarah	31	female	single	nurse	1212 4th Ave	Denver CO	80203
Davis	Jason	41	male	married	sales representative	345 Cedar Blvd	Austin TX	78701
Brown	Laura	25	female	single	student	567 Park Ave	Philadelphia PA	19103
Jackson	Robert	50	male	married	business owner	789 Madison St	San Diego CA	92101
Martin	Olivia	29	female	single	journalist	987 Hillside Rd	Portland OR	97201
Garcia	William	35	male	married	teacher	1234 Cherry St	Miami FL	33131
Kim	Grace	24	female	single	web developer	5678 Vine St	San Francisco CA	94102
Martinez	Ethan	28	male	single	architect	9012 Sunset Blvd	Los Angeles CA	90046
Rodriguez	Elizabeth	33	female	married	accountant	3456 Park Ave	New York NY	10016
Nguyen	Kevin	26	male	single	data analyst	7890 Pine St	Boston MA	02108
Hernandez	Maria	40	female	married	homemaker	123 Spring St	Houston TX	77002
Lee	Joshua	31	male	single	consultant	456 Main St	Chicago IL	60601
Kim	Rachel	28	female	single	journalist	789 Hillside Rd	San Diego CA	92101
Davis	Steven	39	male	married	engineer	1011 Cedar Blvd	Austin TX	78701
Smith	Jessica	27	female	single	teacher	1213 4th Ave	Denver CO	80203
Brown	David	45	male	married	lawyer	567 Park Ave	Philadelphia PA	19103
Jackson	Amanda	29	female	single	artist	789 Madison St	San Francisco CA	94102
Martin	James	35	male	married	accountant	987 Hillside Rd	Portland OR	97201
Garcia	Samantha	24	female	single	programmer	1234 Cherry St	Miami FL	33131
Green	Isabella	31	female	single	journalist	234 Elm St	Boston MA	02108
White	Christopher	40	male	married	financial analyst	567 Pine St	San Francisco CA	94105
Martinez	Sofia	29	female	single	graphic designer	678 Maple Rd	Los Angeles CA	90001
Wright	Benjamin	26	male	single	software developer	890 Pine St	Chicago IL	60601
Brown	Victoria	38	female	married	nurse	1212 4th Ave	Denver CO	80203
Taylor	Matthew	35	male	single	journalist	345 Cedar Blvd	Austin TX	78701
Davis	Audrey	29	female	married	HR manager	567 Park Ave	Philadelphia PA	19103
Rodriguez	Noah	27	male	single	graphic designer	789 Madison St	San Diego CA	92101
Patel	Maya	30	female	single	marketing manager	987 Hillside Rd	Portland OR	97201
Wilson	Samuel	43	male	married	financial advisor	123 Main St	San Francisco CA	94105
Lee	Charlotte	25	female	single	accountant	456 Elm St	Miami FL	33131
Adams	Aaron	34	male	married	engineer	789 Oak Ave	Seattle WA	98101
Campbell	Chloe	28	female	single	writer	1010 Broadway St	New York NY	10003
Mitchell	Vincent	36	male	single	IT specialist	2345 Main St	Boston MA	02108
Wright	Elena	31	female	married	project manager	678 Maple Rd	Los Angeles CA	90001
King	Nicholas	29	male	single	marketing analyst	890 Pine St	Chicago IL	60601
Scott	Isabelle	36	female	married	homemaker	1212 4th Ave	Denver CO	80203
Carter	Adrian	42	male	married	businessman	345 Cedar Blvd	Austin TX	78701
Evans	Caroline	27	female	single	web designer	567 Park Ave	Philadelphia PA	19103
Lee	Alexander	32	male	married	pharmacist	789 Madison St	San Francisco CA	94102
Perez	Grace	28	female	single	data scientist	987 Hillside Rd	Portland OR	97201
Coleman	Justin	30	male	single	consultant	1234 Cherry St	Miami FL	33131
Rivera	Emma	26	female	single	photographer	456 Main St	Chicago IL	60601
Nguyen	Lucas	33	male	married	teacher	789 Oak Ave	Seattle WA	98101

```
Process returned 0 (0x0)   execution time : 0.443 s
Press any key to continue.
```

The Design (my solution – the way I did it):

DemographicInfo class (a separate code file)

Private data members/attributes for each field of a record read from the provided file:

firstName, lastName, age, sex, maritalStatus, occupation, streetAddress, city, state, zipCode

The provided input file is a Comma Separated Values (CSV) file specifically designed to separate data fields. Don't submit something where the fields have not been separated.

Functions/methods

DemographicInfo(string record): ctor that accepts a single record that was pulled from the provided file. This constructor calls setAll().

setAll(string record): This is where the record is parsed into its components and stored in the object data members/attributes.

printInfoShort(): The data members / attributes are output directly from here (as opposed to returning the data for output elsewhere. This is also where the data is rearranged to match the example output.

DemographicNode class (a separate code file)

Private data members/attributes (as mentioned previous, there are only 3):

previousNodeRef: contains a pointer/reference to the previous Node. The head node will point to NULL
demographic: This will contain one DemographicInfo object.
nextNodeRef: contains a pointer/reference to the next Node. The tail node will point to NULL

Functions/methods

DemographicNode(string aRecord): ctor

printNodeData(): Calls the print function for the DemographicInfo object

getters (accessors)/setters(mutators)

getPrev()

setPrev(prevNode)
getNext()
setNext(nextNode)

DemographicLL linked list class (a separate code file)

Private data members/attributes:

head: The head node for the linked list.
tail: The tail (last node) for the linked list. Initially, when the first node is added to the list, the head and tail are pointing to the same Node.

Functions/methods

DemographicLL(): ctor. Null the head and tail, then call readDataFile().

readDataFile(): Reads the provided datafile, calls addNode() for each record read from the file. Don't forget to close the file.

addNode(aRecord): Creates a DemographicNode with the provided record. It also sets the previous and next pointers. This is a bit tricky. The first time through, the head and tail will be the same node. New nodes are added to the end of the list, such that they become the tail and their next pointers are NULL (end of the linked list). The previous pointer of the new node, points to the previous node which was last in the list and that same node, now points its next pointer to the new node. Think about it. Don't be afraid to draw some pictures.

printList(): Prints the table header and then traverses the linked list, printing each DemographicNode in turn.

DemographicTest main() (a separate code file)

Declare a DemographicLL. This creates the link list from the single table file. Then, call the printList() function/method from DemographicLL.