

## MyHeader.h

```
1 /*****
2 * AUTHOR      : Amirarsalan Valipour & Ali Eshghi
3 * STUDENT ID  : 1103126 - 1112261
4 * LAB #10     : Creating an Ordered List
5 * CLASS       : CS 1B
6 * SECTION     : MW - 7:30 pm
7 * DUE DATE    : 11/05/2019
8 *****/
9 * PrintMenu
10 *****/
11 * This function will output the menu, gets the user's option and checks to see
12 *   if the input is eligible or not.
13 *****/
14
15 #ifndef MYHEADER_H_
16 #define MYHEADER_H_
17
18 #include<iostream>
19 #include<iomanip>
20 #include<string>
21 #include<fstream>
22 #include<limits>
23 #include<sstream>
24 using namespace std;
25
26 const string NAME = "Creating an Ordered List";
27 const char   TYPE  = 'L';
28 const int    NUM    = 10 ;
29 const string CLASS  = "CS1B";
30 const string SECTION = "MW: 7:30p - 9:50p";
31
32
33
34 enum MenuOption
35 {
36     EXIT,
37     CREATE,
38     DISPLAY,
39     ISEMPY,
40     SEARCH,
41     REMOVE,
42     CLEAR
43 };
44
45 struct PersonNode
46 {
47     string    name;
48     char      gender;
49     int       age;
50     PersonNode *next;
51     PersonNode *prev;
52 };
53
54 /*****
55 * Function - PrintHeaderFile
```

## MyHeader.h

```
56 * -----
57 * This function will output the class heading to the screen.
58 *
59 * return type - nothing
60 *           the function is void type
61 *****/
62 void PrintHeaderFile();
63
64 /*****
65 * Function - PrintMenu
66 * -----
67 * This function will output menu option to the screen and waits for the user to
68 * input an option to what to what to do. The options are as following:
69 *
70 * 1 - Create List
71 * 2 - Display List
72 * 3 - Is the list empty?
73 * 4 - Search by name
74 * 5 - Remove Node
75 * 6 - Clear List
76 * 0 - to Exit
77 *
78 * return type - Integer
79 *****/
80 int PrintMenu();
81
82 /*****
83 * Function - CreatList
84 * -----
85 * This function will get the data from the input file and puts the data into
86 * the contents of the nodes, then add the nodes to the empty lists passed by
87 * Reference to the function
88 *
89 * return type - nothing
90 *           the function is void type
91 *****/
92 void CreatList(PersonNode *&head);
93
94 /*****
95 * Function - DisplayList
96 * -----
97 * This function will output the contents of the nodes of the linked list
98 * created in the CreatList function
99 *
100 * return type - nothing
101 *           the function is void type
102 *****/
103 void DisplayList(PersonNode *head);
104
105 /*****
106 * Function - IsEmpty
107 * -----
108 * This function check if the list created or modified by the user is empty or
109 * no and then outputs if the list is empty or no
110 *
```

## MyHeader.h

```
111 * return type - nothing
112 *           the function is void type
113 *****/
114 void IsEmpty(PersonNode *head);
115
116 /*****
117 * Function - SearchName
118 * -----
119 * This function will get a name from the user and search the name content of
120 * each node to see if there is a matching name in the nodes with the name
121 * searched by the user. if found, the function prints out every content of the
122 * node for the user, if not found, the function outputs that the name searched
123 * by the user was not found.
124 *
125 * return type - nothing
126 *           the function is void type
127 *****/
128 void SearchName(PersonNode *head);
129
130 /*****
131 * Function - RemoveNode
132 * -----
133 * This function will ask the user which node the user wants to remove and it
134 * searches the nodes based on the name content of the nodes and if the name
135 * content of a node matches the name input by the user, the function removes
136 * the node from the list, if not, the function outputs that the name searched
137 * by the user has not found in the list.
138 *
139 * return type - nothing
140 *           the function is void type
141 *****/
142 void RemoveNode(PersonNode *&head);
143
144 /*****
145 * Function - ClearList
146 *****/
147 * This function Allows the user to delete all the nodes in the linked list and
148 * make an empty list out of the list of the names that we had
149 *
150 * return type - nothing
151 *           the function is void type
152 *****/
153 void ClearList(PersonNode *&head);
154
155
156 #endif /* MYHEADER_H_ */
157
```