

Alexandria Higher Institute For Engneering&Technology

Project: Contact list

Fares Mohammed Abdel Latif

Nardeen Samuel Botros

Rovana Mohammed Kamel

#include <stdio.h>  
#include <stdlib.h>  
void edit();  
int search();  
void System();  
void del();  
void add();  
void disall();  
void dis();  
struct node{  
 int ID,Number;  
 char Email[60],Name[70];  
 struct node \*next;  
};  
struct node \*node,\*head=NULL,\*tail=NULL;  
  
main()  
{  
 System ();  
}  
void System(){  
 int operation;  
 printf("\n1)Add item 2)Edit item\n3)Delete item 4)Display item\n5)Display all 6)Close program\nEnter operation number: ");  
 scanf ("%d", &operation);  
 switch(operation)  
 { case 1:  
 add ();  
 break;  
 case 2:  
 edit();  
 break;  
 case 3:  
 del();  
 break;  
 case 4:  
 dis();  
 break;  
  
 case 5:  
 disall ();  
 break;  
 case 6:  
 exit (0);  
 default:  
 printf("\nUnvalid operation\n");  
 System();  
  
  
  
 }  
 System();  
}  
int search()  
{  
 int item;  
 node=head;  
 printf ("\nEnter contact ID: ");  
 scanf ("%d",&item);  
 while (node != NULL)  
 {  
 if (node->ID == item)  
  
 return 1;  
  
  
 node = node->next;  
 }  
 if (node == NULL)  
 {  
 printf ("\nContact not found\n");  
 return 0;  
 }  
}  
  
void edit() {  
 int operation, check, contactID;  
 check = search();  
 if (check == 0)  
 System();  
 if (check == 1) {  
 printf("\n1)Edit number\n2)Edit name\n3)Edit Email\nEnter operation number:");  
 scanf("%d", &operation);  
 switch (operation) {  
 case 1:  
 printf("\nEnter the new number: ");  
 scanf("%d", &node->Number);  
 break;  
  
 case 2:  
  
  
 printf("\nEnter the new Name: ");  
 scanf("%s", &node->Name);  
  
 break;  
  
 case 3:  
  
 printf("\nEnter the new Email:");  
 scanf("%s", &node->Email);  
  
  
 break;  
 default:  
 printf("\nUnvalid operation\n");  
 break;  
  
  
 }  
 printf("\nContact edited successfully\n");  
  
 System();  
 }  
}  
void del()  
{ int check = search();  
 if (check == 0){  
  
 System();}  
 struct node \*prev;  
  
 if (node->ID== head->ID)  
 {  
 head=node->next;  
 free(node);  
 printf("\nContact deleted successfully\n");  
  
  
 System();  
 }  
 prev=head;  
  
 while(prev->next!=node)  
 {prev=prev->next;}  
  
  
 prev->next = node->next;  
  
 free(node);  
 printf("\nContact deleted successfully\n");  
  
 System();  
}  
void add()  
{  
 struct node \*newnode = (struct node \*) (malloc(sizeof(struct node)));  
  
 if(head==NULL) {  
 printf("\nEnter contact ID: ");  
 scanf("%d", &newnode->ID);  
 printf ("\nEnter number of contact: ");  
 scanf ("%d", &newnode->Number);  
 printf ("\nEnter name of contact: ");  
 scanf ("%s", &newnode->Name);  
 printf ("\nEnter Email of contact: ");  
 scanf ("%s", &newnode->Email);  
 newnode->next = NULL;  
 if (head == NULL)  
 {  
 head = newnode;  
 tail = newnode;  
  
 }  
 else  
 {  
 tail->next = newnode;  
 tail = newnode;  
 }  
 printf("\nContact added successfully\n");  
 System();  
 }  
 if(newnode==NULL)  
 { printf("\nNo memory available\n");  
 System();}  
 printf ("\nEnter ID of the contact: ");  
 scanf ("%d", &newnode->ID);  
 node=head;  
 while (node != NULL) {  
 if (node->ID == newnode->ID) {  
 printf("\nID used before\nreturning to menu....\n");  
 System();  
 }  
 node = node->next;  
 }  
 printf ("\nEnter number of the contact: ");  
 scanf ("%d", &newnode->Number);  
 printf ("\nEnter name of the contact: ");  
 scanf ("%s", &newnode->Name);  
 printf ("\nEnter Email of the contact: ");  
 scanf ("%s", &newnode->Email);  
 newnode->next=NULL;  
 tail->next=newnode;  
 tail=newnode;  
 printf("\nContact added successfully\n");  
 System();  
}  
  
void disall()  
{  
  
 node = head;  
 int x=1;  
 while (node!= NULL)  
 {  
 printf("\ncontact %d :\n", x);  
 printf("\nName: %s\n" , node->Name);  
 printf("\nPhone number: 0%d\n" , node->Number);  
 printf("\nEmail: %s\n" , node->Email);  
 node = node->next;  
 x++;  
 }  
 System();  
}  
void dis() {  
 int check=search();  
 if (check == 1) {  
 printf("\nName :%s\n", node->Name);  
 printf("\nPhone number: 0%d\n", node->Number);  
 printf("\nEmail: %s\n", node->Email);  
 System();  
 }  
}