Linked List Operations

CODE:

```
#include <stdio.h>
#include <stdlib.h>
struct Node {
    int data;
    struct Node* next;
};
struct Node* head = NULL;
void insertEnd(int value) {
    struct Node* newNode = (struct Node*)malloc(sizeof(struct Node));
    newNode->data = value;
    newNode->next = NULL;
    if (head == NULL) {
        head = newNode;
    } else {
        struct Node* temp = head;
        while (temp->next != NULL)
            temp = temp->next;
        temp->next = newNode;
    }
    printf("Inserted %d\n", value);
}
void displayList() {
    if (head == NULL) {
        printf("List is empty.\n");
        return;
    }
    struct Node* temp = head;
    printf("Linked List: ");
```

```
    while (temp != NULL) {
        printf("%d -> ", temp->data);
        temp = temp->next;
    }
    printf("NULL\n");
}
void deleteNode(int value) {
    struct Node* temp = head;
    struct Node* prev = NULL;
    if (temp != NULL && temp->data == value) {
        head = temp->next;
        free(temp);
        printf("Deleted %d\n", value);
        return;
    }
    while (temp != NULL && temp->data != value) {
        prev = temp;
        temp = temp->next;
    }
    if (temp == NULL) {
        printf("Value %d not found.\n", value);
        return;
    }
    prev->next = temp->next;
    free(temp);
    printf("Deleted %d\n", value);
}
int main() {
    int choice, value;
    while (1) {
```

```
        printf("\n Linked List Menu \n");
        printf("1. Insert at end\n");
        printf("2. Display list\n");
        printf("3. Delete by value\n");
        printf("4. Exit\n");
        printf("Enter your choice: ");
        scanf("%d", &choice);
        switch (choice) {
            case 1:
                printf("Enter value to insert: ");
                scanf("%d", &value);
                insertEnd(value);
                break;
            case 2:
                displayList();
                break;
            case 3:
                printf("Enter value to delete: ");
                scanf("%d", &value);
                deleteNode(value);
                break;
            case 4:
                printf("Exiting...\n");
                exit(0);
            default:
                printf("Invalid choice! Try again.\n");
        }
    }
    return 0;
}
```

OUTPUT:

```
 Linked List Menu
1. Insert at end
2. Display list
3. Delete by value
4. Exit
Enter your choice: 1
Enter value to insert: 12
Inserted 12

 Linked List Menu
1. Insert at end
2. Display list
3. Delete by value
4. Exit
Enter your choice: 1
Enter value to insert: 24
Inserted 24

 Linked List Menu
1. Insert at end
2. Display list
3. Delete by value
4. Exit
Enter your choice: 2
Linked List: 12 -> 24 -> NULL

 Linked List Menu
1. Insert at end
2. Display list
3. Delete by value
```

```
Linked List: 12 -> 24 -> NULL

 Linked List Menu
1. Insert at end
2. Display list
3. Delete by value
4. Exit
Enter your choice: 3
Enter value to delete: 12
Deleted 12

 Linked List Menu
1. Insert at end
2. Display list
3. Delete by value
4. Exit
Enter your choice: 2
Linked List: 24 -> NULL

 Linked List Menu
1. Insert at end
2. Display list
3. Delete by value
4. Exit
Enter your choice: 4
Exiting...

--------------------------------
Process exited after 39.81 seconds with return value 0
Press any key to continue . . .
```