P1 Write a program for creation of singly linked list.

#include <stdio.h>

#include <stdlib.h>

typedef struct Node {

    int data;

    struct Node\* next;

} Node;

Node\* createNode(int value) {

    Node\* newNode = (Node\*)malloc(sizeof(Node));

    newNode->data = value;

    newNode->next = NULL;

    return newNode;

}

Node\* insert(Node\* head, int value) {

    Node\* newNode = createNode(value);

    if (head == NULL) {

        return newNode;

    }

    Node\* temp = head;

    while (temp->next != NULL) {

        temp = temp->next;

    }

    temp->next = newNode;

    return head;

}

void display(Node\* head) {

    if (head == NULL) {

        printf("List is empty.\n");

        return;

    }

    Node\* temp = head;

    printf("Linked List: ");

    while (temp != NULL) {

        printf("%d -> ", temp->data);

        temp = temp->next;

    }

    printf("\n");

}

int main() {

    printf("Shiv Arora\n");

    Node\* head = NULL;

    int choice, value;

    printf("\nMenu:\n");

    printf("1. Insert\n2. Display\n3. Exit\n");

    while (1) {

        printf("Enter your choice: ");

        scanf("%d", &choice);

        switch (choice) {

            case 1:

                printf("Enter value to insert: ");

                scanf("%d", &value);

                head = insert(head, value);

                break;

            case 2:

                display(head);

                break;

            case 3:

                printf("Dead\n");

                return 0;

            default:

                printf("Invalid choice. Please try again.\n");

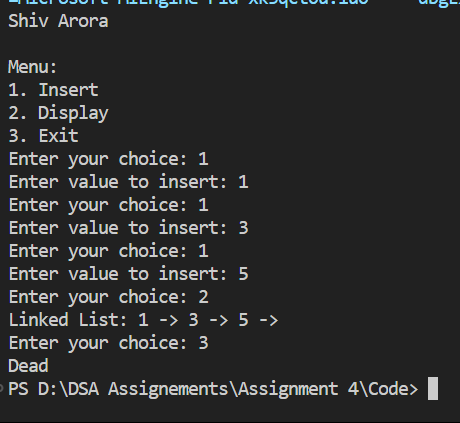
        }

    }

    return 0;

}

OUTPUT:



P2 Write a program for creation of doubly linked list.

#include <stdio.h>

#include <stdlib.h>

typedef struct Node {

    int data;

    struct Node\* next;

    struct Node\* prev;

} Node;

Node\* createNode(int value) {

    Node\* newNode = (Node\*)malloc(sizeof(Node));

    newNode->data = value;

    newNode->next = NULL;

    newNode->prev = NULL;

    return newNode;

}

Node\* insert(Node\* head, int value) {

    Node\* newNode = createNode(value);

    if (head == NULL) {

        return newNode;

    }

    Node\* temp = head;

    while (temp->next != NULL) {

        temp = temp->next;

    }

    temp->next = newNode;

    newNode->prev = temp;

    return head;

}

void display(Node\* head) {

    if (head == NULL) {

        printf("List is empty.\n");

        return;

    }

    Node\* temp = head;

    printf("Linked List: ");

    while (temp != NULL) {

        printf("%d <-> ", temp->data);

        temp = temp->next;

    }

    printf("\n");

}

int main() {

    printf("Shiv Arora\n");

    Node\* head = NULL;

    int choice, value;

    printf("\nMenu:\n");

    printf("1. Insert\n2. Display\n3. Exit\n");

    while (1) {

        printf("Enter your choice: ");

        scanf("%d", &choice);

        switch (choice) {

            case 1:

                printf("Enter value to insert: ");

                scanf("%d", &value);

                head = insert(head, value);

                break;

            case 2:

                display(head);

                break;

            case 3:

                printf("Dead\n");

                return 0;

            default:

                printf("Invalid choice. Please try again.\n");

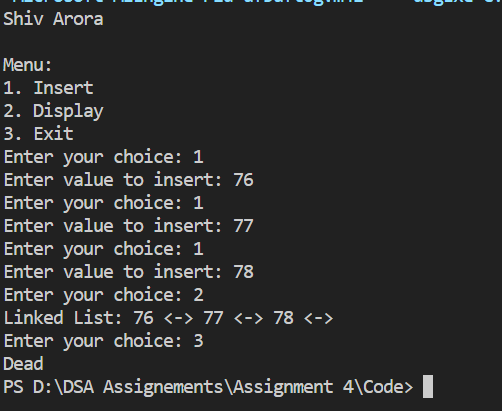
        }

    }

    return 0;

}

OUTPUT:



P3 Write a program for creation of circular singly linked list.

#include <stdio.h>

#include <stdlib.h>

typedef struct Node {

    int data;

    struct Node\* next;

} Node;

Node\* createNode(int value) {

    Node\* newNode = (Node\*)malloc(sizeof(Node));

    newNode->data = value;

    newNode->next = NULL;

    return newNode;

}

Node\* insert(Node\* head, int value) {

    Node\* newNode = createNode(value);

    if (head == NULL) {

        newNode->next = newNode;

        return newNode;

    }

    Node\* temp = head;

    while (temp->next != head) {

        temp = temp->next;

    }

    temp->next = newNode;

    newNode->next = head;

    return newNode;

}

void display(Node\* head) {

    if (head == NULL) {

        printf("List is empty.\n");

        return;

    }

    Node\* temp = head;

    printf("Linked List: ");

    do {

        printf("%d -> ", temp->data);

        temp = temp->next;

    } while (temp != head);

    printf("(head)\n");

}

int main() {

    printf("Shiv Arora\n");

    Node\* head = NULL;

    int choice, value;

    printf("\nMenu:\n");

    printf("1. Insert\n2. Display\n3. Exit\n");

    while (1) {

        printf("Enter your choice: ");

        scanf("%d", &choice);

        switch (choice) {

            case 1:

                printf("Enter value to insert: ");

                scanf("%d", &value);

                head = insert(head, value);

                break;

            case 2:

                display(head);

                break;

            case 3:

                printf("Dead\n");

                return 0;

            default:

                printf("Invalid choice. Please try again.\n");

        }

    }

    return 0;

}

OUTPUT:

