```cpp
//Linked List
// Author - Jyoti Lakhani

#include<iostream>
using namespace std;
//Structure of a Node
struct Node
    {
            int info;
            struct Node *next;
    }*new_node,*start,*ptr;

struct Node *create_node(int);
int value;
//Create a New Node
struct Node *create_node(int value)
{   new_node = new Node();
    new_node->info= value;
    new_node->next=NULL;
    return new_node;
}
// Insertion of Node at the begining of the Linked List
void insert_begin(struct Node *new_node)
{      //case 1: linked list is Empty
    if(start==NULL)
    {
            start=new_node;
    }
    else // Linked List not empty
    {
            new_node->next = start;
            start= new_node;
    }
}
//Display or Traverse a Linked List
void display()
{
    if(start==NULL)
    {
            cout<<"\n No node to diaplay...\nList is Empty";
            return;
    }
    else
    {
        ptr = start;
          while(ptr!=NULL)
          {
                cout<<ptr->info<<"->";
                ptr=ptr->next;

          }
          cout<<"!!!"<<endl;
    }
}

int main()
{
    int flag=0;
    int ch;
```

```cpp
        while(flag==0)
        {
                cout<<"Choose an option\n";
                cout<<"1. Create a node \n2. Insert at Front \n3. Display/Travarse \n4.
Exit\n" ;
                cin>>ch;
                switch(ch)
                {
                        case 1:
                                cout<<("Enter info for new node\n");
                                cin>>value;
                                new_node = create_node(value);
                                if(new_node!= NULL)
                                {
                                        cout<<"\nnew_node created sucessfully\n";
                                        cout<<"new_node->info = "<<new_node->info<<endl;
                                }
                                else
                                {
                                        cout<<"Problem in creating node\n Try Next Time\n";
                                        return 0;
                                }
                                break;
                        case 2:
                                insert_begin(new_node);
                                cout<<"\n New node inserted at the begining successfully\
n";
                                break;
                        case 3:
                                cout<<"\n Traversing Linked List..\n";
                                display();
                                break;
                        case 4:
                                cout<<"Exiting...\n";
                                return 0;
                }
        }
        return 0;
}
```