volur or.v | doublyll.c | test.java | import java.util.*; | game.java | quiz.java | wordtest.java

```c
#include<stdio.h>
#include<stdlib.h>
struct node
{
    int data;
    struct node *next;
    struct node *prev;
};
struct node *head=NULL;
void insertleft()
{
     struct node *new_node;
     new_node=(struct node*)malloc(sizeof(struct node));
     printf("Enter the number: \n");
     scanf("%d",&new_node->data);
     new_node->next=NULL;
     new_node->prev=NULL;

     if(head==NULL)
     {
          head=new_node;
     }
     else
     {
          new_node->next=head;
          head->prev=new_node;
          head=new_node;
     }

}


```

volur or.v | doublyll.c | test.java | import java.util.*; | game.java | quiz.java | wordtest.java

```c

void del()
{
    struct node *temp;
    int elem;
    if(head==NULL)
    {
        printf("Empty List \n");
        return;
    }
    printf("Enter the element to be deleted\n");
    scanf("%d",&elem);
    temp=head;
    while(temp->data!=elem)
    {
        temp=temp->next;
        if(temp==NULL)
        {
         printf("Element is not in the list\n");
         return;
        }
     }
     if(temp==head)
     {
         head=head->next;
     }
     else if(temp->next==NULL)
     {
          temp=temp->prev;
          temp->next=NULL;
     }

     else
     {
         temp->prev->next=temp->next;
         temp->next->prev=temp->prev;
     }
}
```

volur or.v ● doublyll.c × test.java × import java.util.*; ● game.java × quiz.java × wordtest.java ×

```
71  void insert_betweenL()
72  {
73      int listele;
74      struct node *new_node,*temp;
75      printf("Enter the element in the list\n");
76      scanf("%d",&listele);
77      new_node=(struct node*)malloc(sizeof(struct node));
78      printf("Enter the new node data\n");
79      scanf("%d",&new_node->data);
80      new_node->next=NULL;
81      new_node->prev=NULL;
82      if(head==NULL)
83      {
84          printf("Empty list\n"); return;
85      }
86      temp=head;
87      while(temp->data!=listele)
88      {
89          temp=temp->next;
90          if(temp==NULL)
91          {
92              printf("Element is  not in the list");
93              return;
94          }
95      }
96      new_node->prev=temp->prev;
97      temp->prev=new_node;
98      new_node->next=temp;
99      new_node->prev->next=new_node;
100 }
```

volur or.v ● doublyll.c × test.java × import java.util.*; ● game.java × quiz.java × word

```
101
102 void insert_betweenR()
103 {
104     int listele;
105     struct node *new_node,*temp;
106     printf("Enter the element in the list\n");
107     scanf("%d",&listele);
108     new_node=(struct node*)malloc(sizeof(struct node));
109     printf("Enter the new node data\n");
110     scanf("%d",&new_node->data);
111     new_node->next=NULL;
112     new_node->prev=NULL;
113     if(head==NULL)
114     {
115         printf("Empty list\n"); return;
116     }
117     temp=head;
118     while(temp->data!=listele)
119     {
120         temp=temp->next;
121         if(temp==NULL)
122         {
123             printf("Element is  not in the list");
124             return;
125         }
126     }
127     new_node->next=temp->next;
128     temp->next=new_node;
129     new_node->prev=temp;
130     new_node->next->prev=new_node;
131 }
132 void display()
133 {
134     struct node *temp;
135     temp=head;
136     while(temp!=NULL)
137     {
138         printf("%d\t",temp->data);
139         temp=temp->next;
140     }
141     printf("\n");
142 }
```

```c
int main()
{
    int choice;
     while(1)
     {

            printf(" 1. Insert to the left in the beginning \n");
            printf(" 2.insert a node before a given node\n");
            printf(" 3.insert a node after a given node\n");
            printf(" 4. delete \n");
            printf(" 5. display\n");
            printf(" 6. exit\n");
            printf("\nEnter your choice: \n");
            scanf("%d",&choice);
            switch(choice)
            {
                 case 1: insertleft(); break;
                 case 2:insert_betweenL();break;
                 case 3:insert_betweenR();break;
                 case 4: del(); break;
                 case 5: display(); break;
                 case 6: exit(0);
            }
     }
}
```

```
 1. Insert to the left in the beginning
 2.insert a node before a given node
 3.insert a node after a given node
 4. delete
 5. display
 6. exit

Enter your choice:
1
Enter the number:
22
 1. Insert to the left in the beginning
 2.insert a node before a given node
 3.insert a node after a given node
 4. delete
 5. display
 6. exit

Enter your choice:
1
Enter the number:
33
```

```
 1. Insert to the left in the beginning
 2.insert a node before a given node
 3.insert a node after a given node
 4. delete
 5. display
 6. exit

Enter your choice:
1
Enter the number:
44
 1. Insert to the left in the beginning
 2.insert a node before a given node
 3.insert a node after a given node
 4. delete
 5. display
 6. exit

Enter your choice:
5
44      33      22
```

```
 1. Insert to the left in the beginning
 2.insert a node before a given node
 3.insert a node after a given node
 4. delete
 5. display
 6. exit

Enter your choice:
2
Enter the element in the list
33
Enter the new node data
5
 1. Insert to the left in the beginning
 2.insert a node before a given node
 3.insert a node after a given node
 4. delete
 5. display
 6. exit

Enter your choice:
5
44      5       33      22
```

```
 1. Insert to the left in the beginning
 2.insert a node before a given node
 3.insert a node after a given node
 4. delete
 5. display
 6. exit

Enter your choice:
3
Enter the element in the list
33
Enter the new node data
6
 1. Insert to the left in the beginning
 2.insert a node before a given node
 3.insert a node after a given node
 4. delete
 5. display
 6. exit

Enter your choice:
5
44      5       33      6       22
 1. Insert to the left in the beginning
 2.insert a node before a given node
 3.insert a node after a given node
```

```
 1. Insert to the left in the beginning
 2.insert a node before a given node
 3.insert a node after a given node
 4. delete
 5. display
 6. exit

Enter your choice:
4
Enter the element to be deleted
22
 1. Insert to the left in the beginning
 2.insert a node before a given node
 3.insert a node after a given node
 4. delete
 5. display
 6. exit

Enter your choice:
5
44      5       33      6
```

```
 1. Insert to the left in the beginning
 2.insert a node before a given node
 3.insert a node after a given node
 4. delete
 5. display
 6. exit

Enter your choice:
6


...Program finished with exit code 0
Press ENTER to exit console.
```