Run Debug Stop Share Save {} Beautify Language C

main.c

```c
#include<stdio.h>
#include<stdlib.h>

struct node
{
    int id;
    char name[20];
    int sem;
    struct node *next;
};

struct node *head=NULL;

void linkedList();
void insertNodeAtBegin();
void insertNodeAtEnd();
void insertNodeAtAny();
void displayList();

int size=0;

int main()
{
    linkedList();
    return 0;
}

void linkedList()
{
    int choice1,choice2;
```

Search the web and Windows

21:31
03-01-2021

Run | Debug | Stop | Share | Save | { } Beautify | Language C

main.c

```c
{
    int choice1,choice2;
    printf("\n <---Enter the Operation--->");
    printf("\n 1.Insert Node. \n 2.Display List \n 3.Exit.\n Choice: ");
    scanf("%d",&choice1);
    switch(choice1)
    {
        case 1: printf("\n 1.At the First Position \t 2.At End of list\t 3.At Any Specified Location\n Choice:");
                scanf("%d",&choice2);
                switch(choice2)
                {
                        case 1: insertNodeAtBegin();
                                break;
                        case 2: insertNodeAtEnd();
                                break;
                        case 3: insertNodeAtAny();
                                break;
                        default: printf("\n Input Error, try Again!\n ");
                                 linkedList();

                }
                break;
        case 2: displayList();
                break;
        case 3: exit(0);
        default:printf("\n Input error, Try again!!\n");
                linkedList();
    }
}
```

Run | Debug | Stop | Share | Save | { } Beautify | Language C

main.c

```c

void insertNodeAtBegin()
{
    struct node *newnode;
    newnode=(struct node*)malloc(sizeof(struct node));
    printf("\n <--Enter the Details--> ");
    printf("\n ID: ");  scanf("%d",&(newnode->id));
    printf(" Name: ");    scanf("%s",(newnode->name));
    printf(" Sem: ");  scanf("%d",&(newnode->sem));
    newnode->next=head;
    head=newnode;
    size++;
    printf("\n Node created \n");
    linkedList();
}
void insertNodeAtEnd()
{
    struct node *newnode,*temp;
    newnode =(struct node *) malloc (sizeof(struct node));
    printf("\n <--Enter the Details--> ");
    printf("\n ID: ");  scanf("%d",&(newnode->id));
    printf(" Name: ");    scanf("%s",(newnode->name));
    printf(" Sem: ");  scanf("%d",&(newnode->sem));
    if (head==NULL)
     {
       newnode->next=NULL;
       head=newnode;
       printf("Node created\n");
       linkedList();
     }
```

Search the web and Windows

21:32
03-01-2021

Run Debug Stop Share Save { } Beautify Language C

main.c

```
    for(temp=head;(temp->next)!=NULL;temp=(temp->next));
    newnode->next=NULL;

    {
            temp->next=newnode;
            size++;
            printf("\n Node created  \n");
            linkedList();
    }

}
void insertNodeAtAny()
{
    struct node *newnode,*temp=head;
    newnode=(struct node*)malloc(sizeof(struct node));
    printf("\n <--Enter the Details--> ");
    printf("\n ID: ");  scanf("%d",&(newnode->id));
    printf(" Name: ");    scanf("%s",(newnode->name));
    printf(" Sem: ");  scanf("%d",&(newnode->sem));
    int pos=0,s=0;
            printf("\n Enter the position(pos>=1 and pos <%d) : ",size);
            scanf("%d",&pos);
            if(pos==0)
            {
                printf("\n Error position, check the operation menu!");
                linkedList();
            }

            for(temp;temp->next!=NULL;temp=temp->next)
            {
```

Search the web and Windows

21:33
03-01-2021

Run | Debug | Stop | Share | Save | { } Beautify | Language C

main.c

```
                    if(s==(pos-1))
                    {
                        newnode->next=(temp->next);
                        temp->next=newnode;
                        size++;
                        printf("\n Node created \n");
                        linkedList();
                    }
                    s++;
                    temp=temp->next;
                }
}

void displayList()
{
    if(head==NULL)
    {
        printf("\n Empty List!\n");
        linkedList();
    }
    printf("\n The List is :");
    for(struct node *temp=head;temp!=NULL;temp=temp->next)
       {
        printf("\n <---Student Details--->");
        printf("\n ID: %d ",temp->id);
        printf("\n Name: %s ",temp->name);
        printf("\n Sem: %d",temp->sem);
       }
       linkedList();
}
```

Search the web and Windows

21:34
03-01-2021

input

```
<---Enter the Operation--->
1.Insert Node.
2.Display List
3.Exit.
Choice: 1

1.At the First Position         2.At End of list        3.At Any Specified Location
Choice:1

<--Enter the Details-->
ID: 10001
Name: Rohan
Sem: 3

Node created

<---Enter the Operation--->
1.Insert Node.
2.Display List
3.Exit.
Choice: 1

1.At the First Position         2.At End of list        3.At Any Specified Location
Choice:2

<--Enter the Details-->
ID: 10002
```

input

```
<--Enter the Details-->
ID: 10002
Name: Ketan
Sem: 5

Node created

<---Enter the Operation--->
1.Insert Node.
2.Display List
3.Exit.
Choice: 1

1.At the First Position         2.At End of list        3.At Any Specified Location
Choice:1

<--Enter the Details-->
ID: 10003
Name: Yash
Sem: 2

Node created

<---Enter the Operation--->
1.Insert Node.
2.Display List
3.Exit.
```

input

```
1.At the First Position         2.At End of list        3.At Any Specified Location
Choice:1

<--Enter the Details-->
ID: 10003
Name: Yash
Sem: 2

Node created

<---Enter the Operation--->
1.Insert Node.
2.Display List
3.Exit.
Choice: 2

The List is :
<---Student Details--->
ID: 10003
Name: Yash
Sem: 2
<---Student Details--->
ID: 10001
Name: Rohan
Sem: 3
<---Student Details--->
ID: 10002
```

input

```
Sem: 2

Node created

<---Enter the Operation--->
1.Insert Node.
2.Display List
3.Exit.
Choice: 2

The List is :
<---Student Details--->
ID: 10003
Name: Yash
Sem: 2
<---Student Details--->
ID: 10001
Name: Rohan
Sem: 3
<---Student Details--->
ID: 10002
Name: Ketan
Sem: 5
<---Enter the Operation--->
1.Insert Node.
2.Display List
3.Exit.
Choice: 1
```

input

```
Sem: 2
<---Student Details--->
ID: 10001
Name: Rohan
Sem: 3
<---Student Details--->
ID: 10002
Name: Ketan
Sem: 5
<---Enter the Operation--->
1.Insert Node.
2.Display List
3.Exit.
Choice: 1

 1.At the First Position         2.At End of list        3.At Any Specified Location
 Choice:3

 <--Enter the Details-->
 ID: 10004
 Name: Mohan
 Sem: 8

 Enter the position(pos>=1 and pos <3) : 2


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