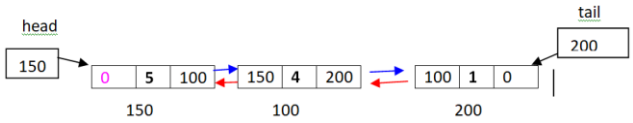
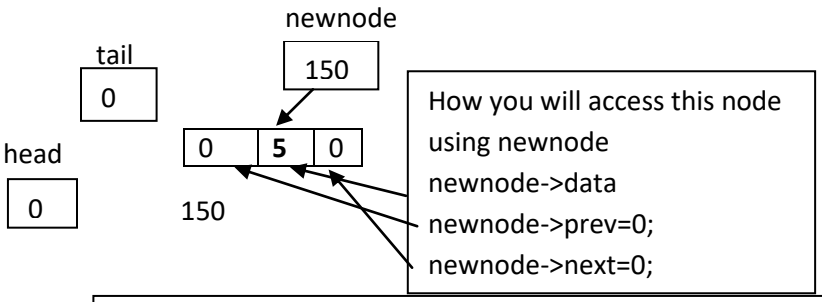
Insertion a node in the doubly linked list





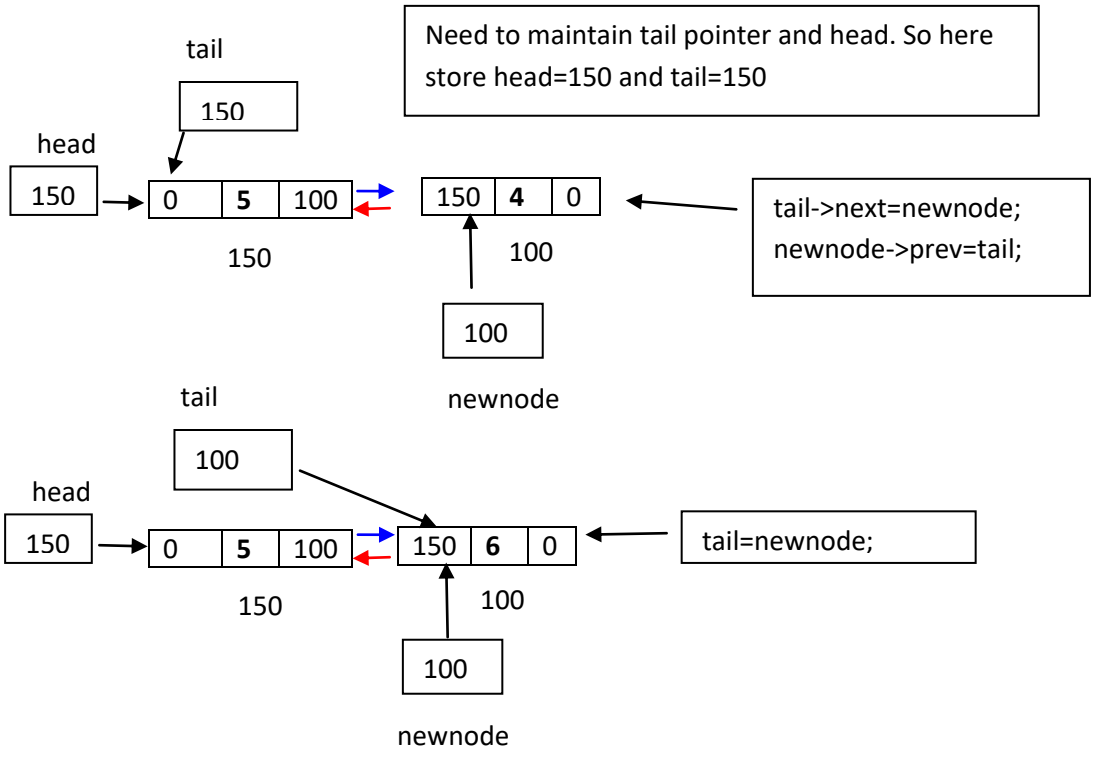

```
#include <stdio.h>
#include <stdlib.h>
struct node
{
int data;
struct node*next;
struct node*prev;
};
struct node *head, *tail;

void createDLL(){
   struct node *newnode, head=0;
   int choice=1;
   while(choice){
newnode= (struct node*)malloc(sizeof(struct node));
printf("Enter Data:");
scanf("%d", &newnode->data);

newnode->prev=0;
newnode->next=0;
if(head==0)
{
head=tail=newnode;
}

else
{
tail->next=newnode;
newnode->prev=tail;
tail=newnode;
}
}}
```

Here tail and head pointers should update to 150

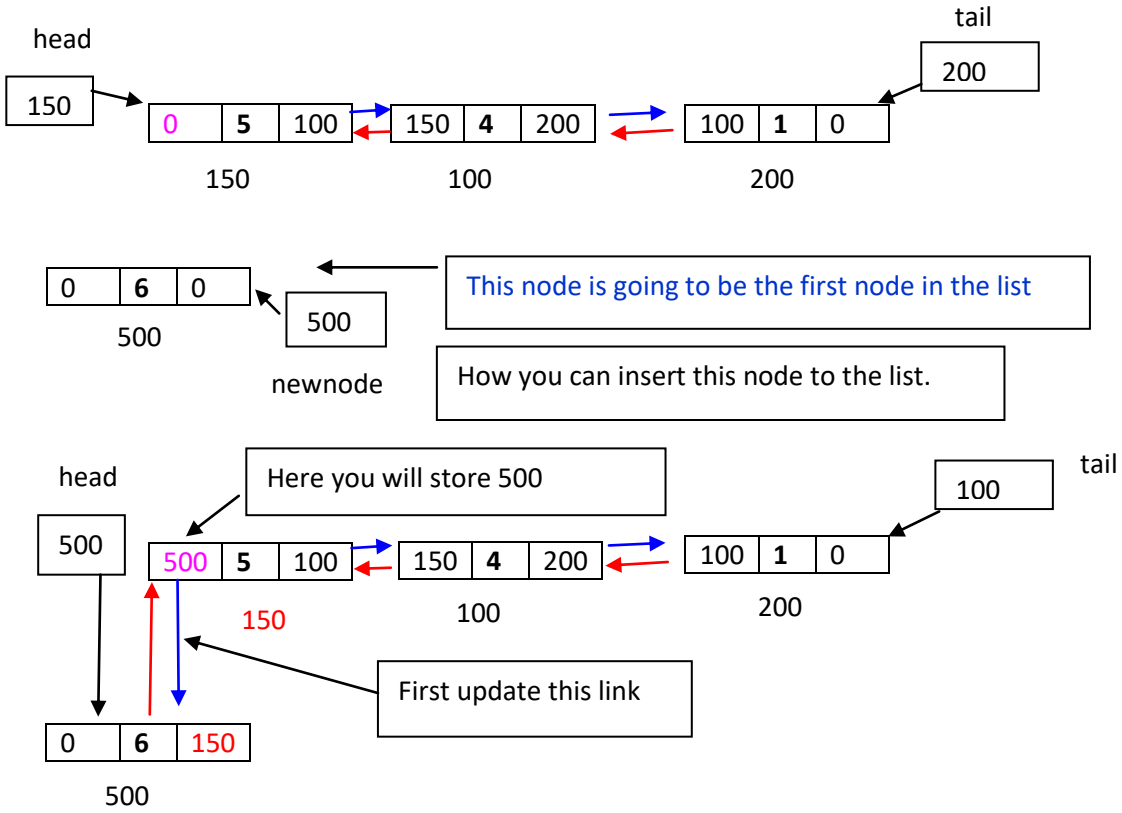
Only that case head=0, we can write head=tail=newnode;

But head is not equal to 0, we add a node to the list

We cannot move head so we move tail.

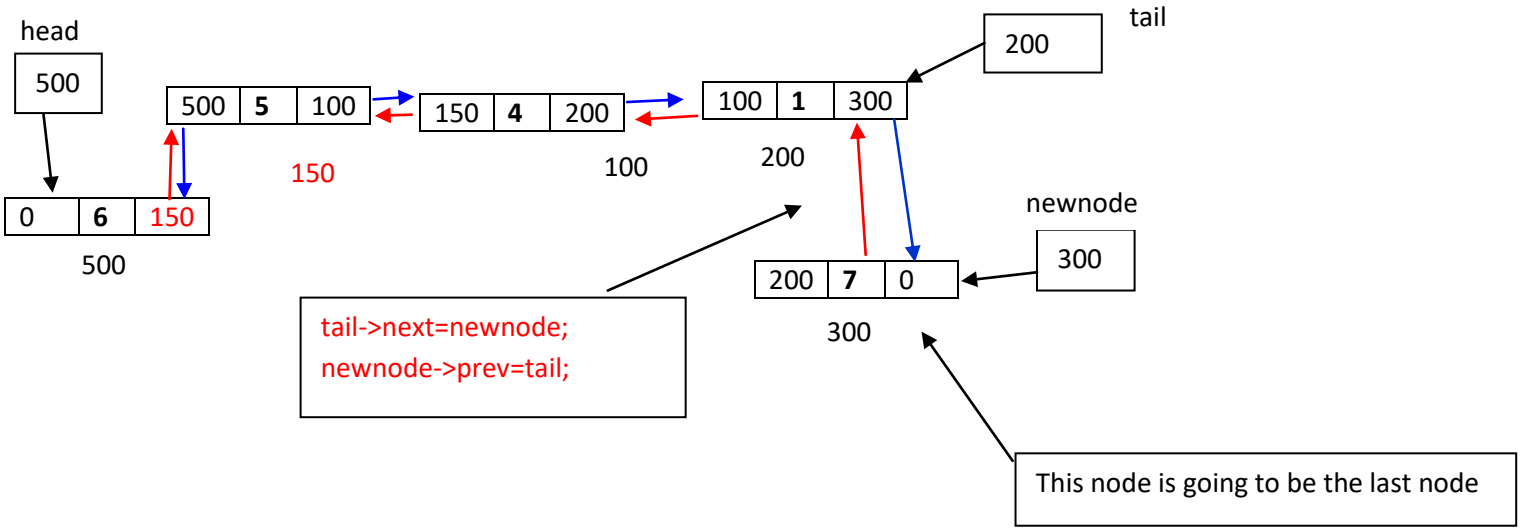


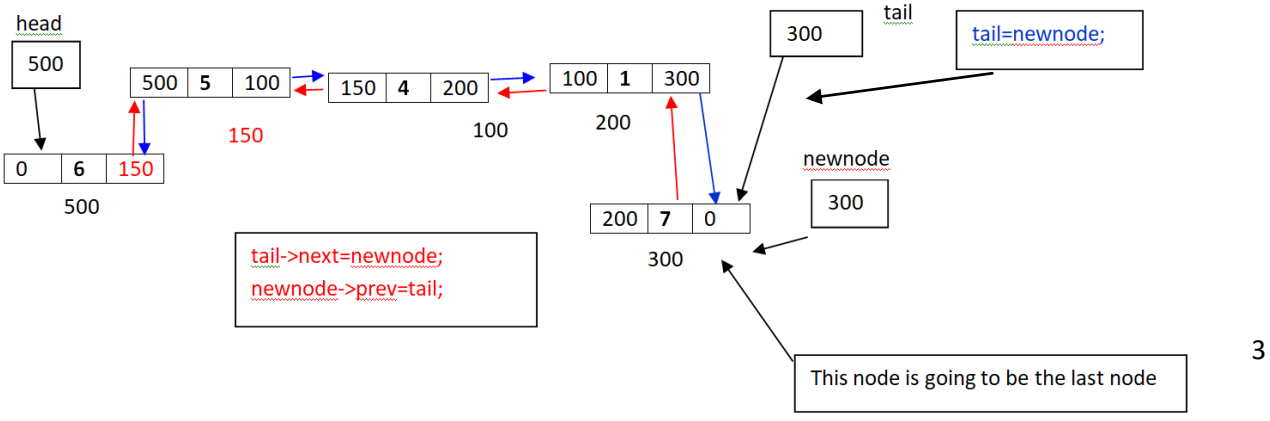
Insert a node at the beginning



```
void insertatbeg()
{
Struct node *newnode;
newnode= (struct node*)malloc(sizeof(struct node));
printf("Enter Data:");
scanf("%d", &newnode->data);
newnode->prev=0;
newnode->next=0;

head->prev=newnode;
newnode->next=head;
head=newnode;
}
```

Insert a node at the end





```
void insertatEnd()
{
struct node *newnode;
newnode= (struct node*)malloc(sizeof(struct node));
printf("Enter Data:");
scanf("%d", &newnode->data);
newnode->prev=0;
newnode->next=0;
tail->next=newnode;
newnode->prev=tail;
tail=newnode;
}
```

```
void display()
{
struct node *temp;
temp=head;
while(temp!=0)
{
printf("%d", temp->data);
temp=temp->next;
}
}
```

Full code

```
#include <stdio.h>
#include <stdlib.h>
struct node
{
int data;
struct node*next;
struct node*prev;
};
struct node *head;
struct node *newnode;
void createDLL(){
   head=0; struct node *temp;
   int choice=1;
   while(choice){
newnode= (struct node*)malloc(sizeof(struct node));
printf("Enter Data:");
scanf("%d", &newnode->data);

newnode->prev=0;
newnode->next=0;
                if(head==0)
                {
                head=temp=newnode;
            }
else
{
temp->next=newnode;
newnode->prev=temp;
temp=newnode;
}
printf("Do you want to continue:");
scanf("%d",&choice);
}
}
void display()
{
struct node *temp;
temp=head;
while(temp!=0)
{
printf("%d", temp->data);
temp=temp->next;
}
}
void insertatbeg()
{
struct node *newnode;
newnode= (struct node*)malloc(sizeof(struct node));
printf("Enter Data:");
scanf("%d", &newnode->data);
newnode->prev=0;
newnode->next=0;
head->prev=newnode;
newnode->next=head;
head=newnode;
}
int main() {
   createDLL();
   display();
   insertatbeg();
   display();
    return 0;
}
```

Full Code

tail should be **updated** in createDLL()

insertatbeg()

insertatEnd()

```
#include <stdio.h>
#include <stdlib.h>
struct node
{
int data;
struct node*next;
struct node*prev;
};
struct node *head,*tail;
struct node *newnode;
void createDLL(){
   head=0; struct node *temp;
   int choice=1;
   while(choice){
newnode= (struct node*)malloc(sizeof(struct node));
printf("Enter Data:");
scanf("%d", &newnode->data);

newnode->prev=0;
newnode->next=0;
                    if(head==0)
                    {
                    //head=temp=newnode;
                    head=temp=tail=newnode;//Update tail whenever a new node is added.
          }
else
{
temp->next=newnode;
newnode->prev=temp;
//temp=newnode;
 temp = tail = newnode;  // Update tail here!
}
printf("Do you want to continue:");
scanf("%d",&choice);
}
}
void display()
{
struct node *temp;
temp=head;
while(temp!=0)
{
printf("%d", temp->data);
temp=temp->next;
}
}
void insertatbeg()
{
struct node *newnode;
newnode= (struct node*)malloc(sizeof(struct node));
printf("Enter Data:");
scanf("%d",&newnode->data);

newnode->prev=0;
newnode->next=0;

//head->prev=newnode;
//newnode->next=head;
//head=newnode;
if(head==0) {
   head=tail=newnode;
} else {
   head->prev=newnode;
   newnode->next=head;
   head=newnode;
}
}
void insertatEnd()
{
struct node *newnode;
newnode= (struct node*)malloc(sizeof(struct node));
printf("Enter Data:");
scanf("%d",&newnode->data);

newnode->prev=0;
newnode->next=0;

if(tail==0) {
   head=tail=newnode;
} else {
   tail->next=newnode;
   newnode->prev=tail;
   tail=newnode;
}
//tail->next=newnode;
//newnode->prev=tail;
//tail=newnode;
}
int main() {
   createDLL();
   display();
   insertatbeg();
   display();
   insertatEnd();
   display();
    return 0;}
```

output

```
Enter Data:1
Do you want to continue:1
Enter Data:3
Do you want to continue:1
Enter Data:8
Do you want to continue:0
138Enter Data:4
4138Enter Data:5
41385
Process returned 0 (0x0)   execution time : 27.573 s
Press any key to continue.
```