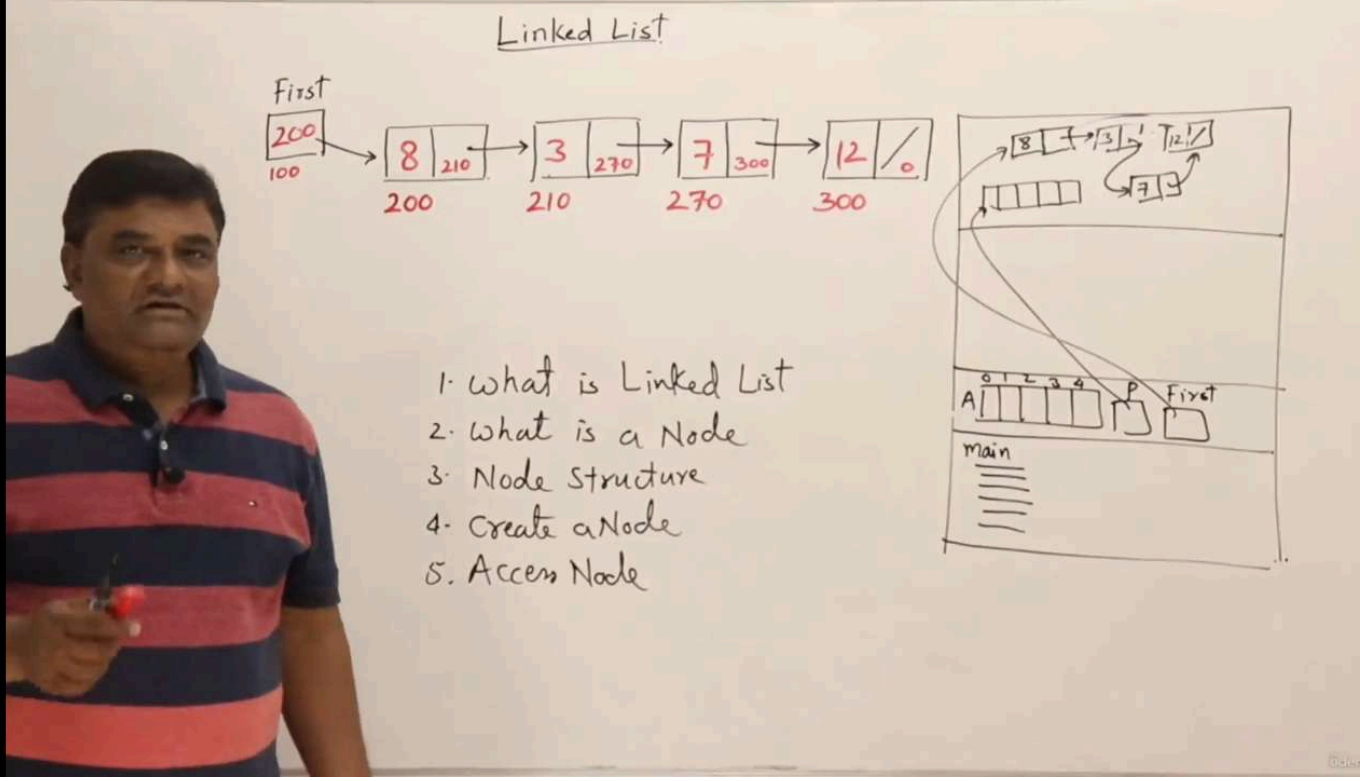
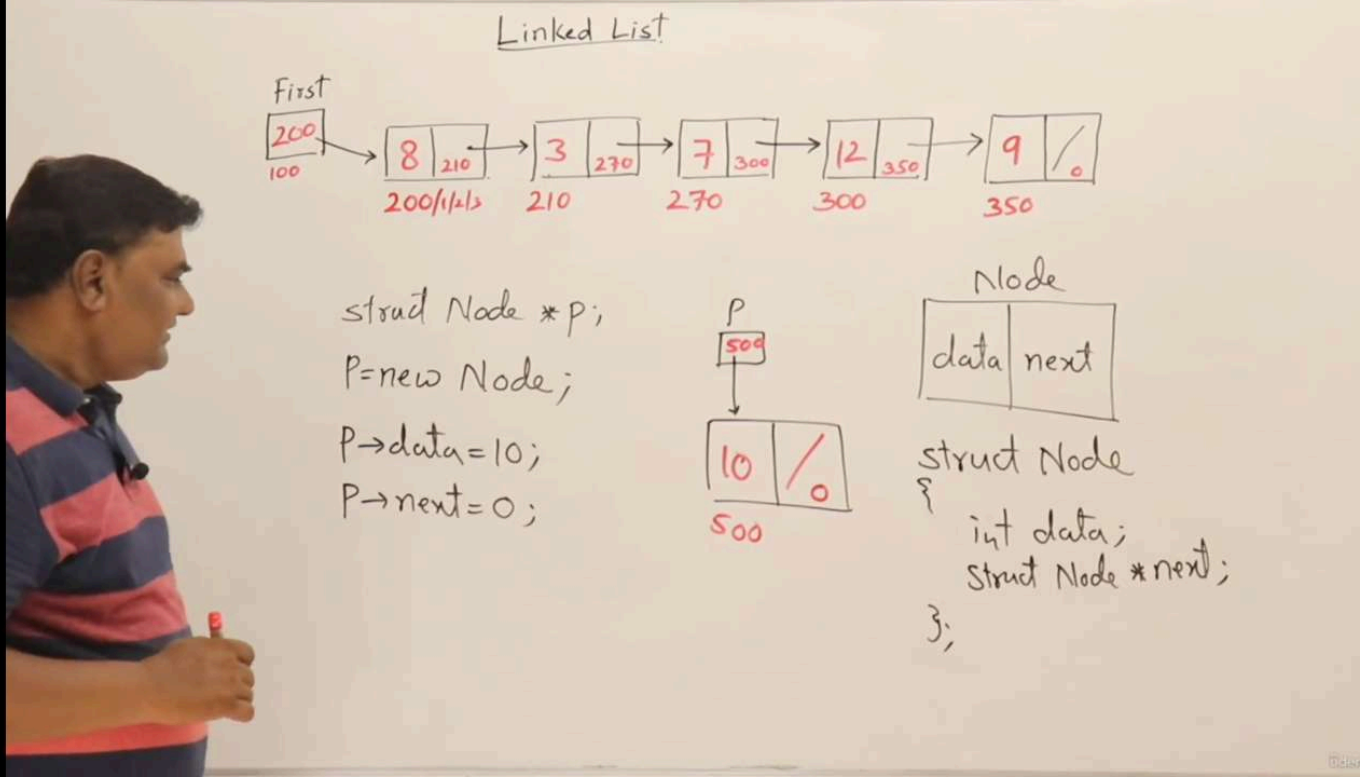
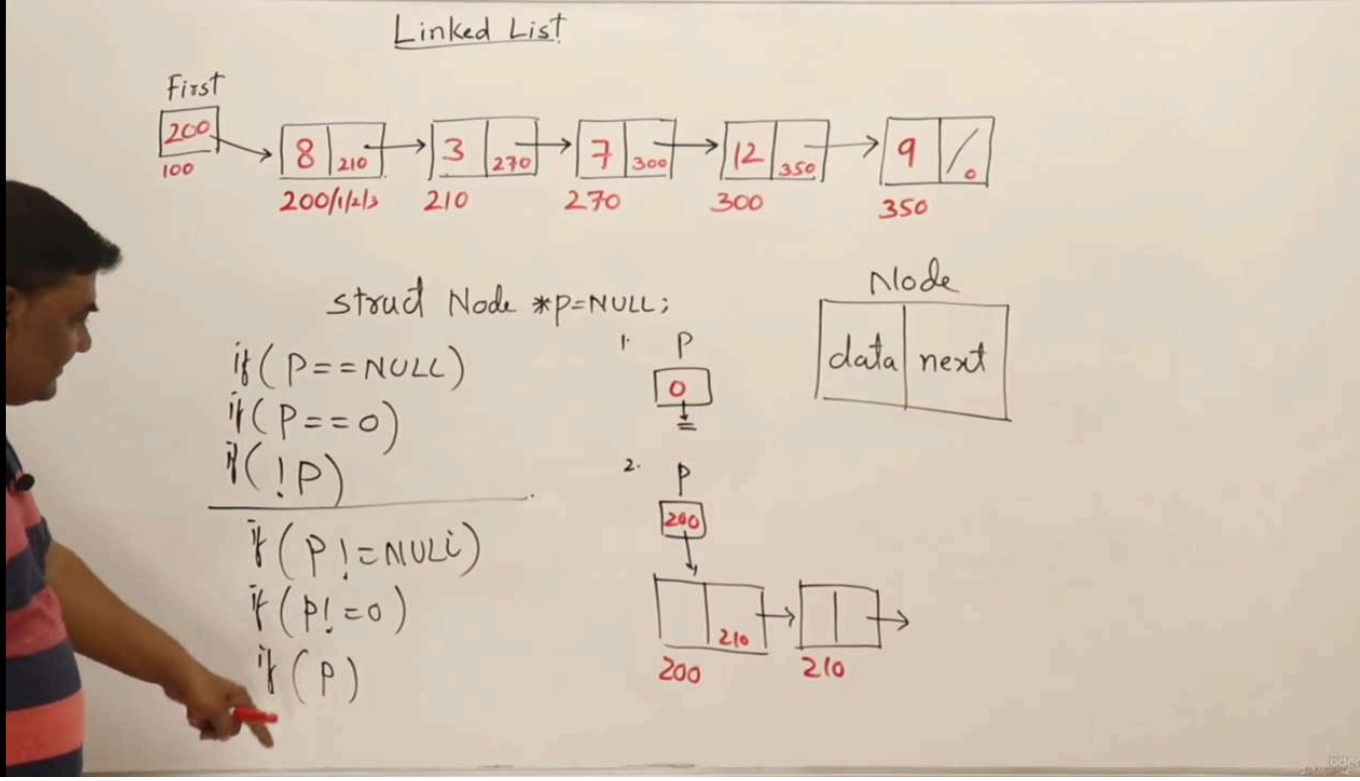
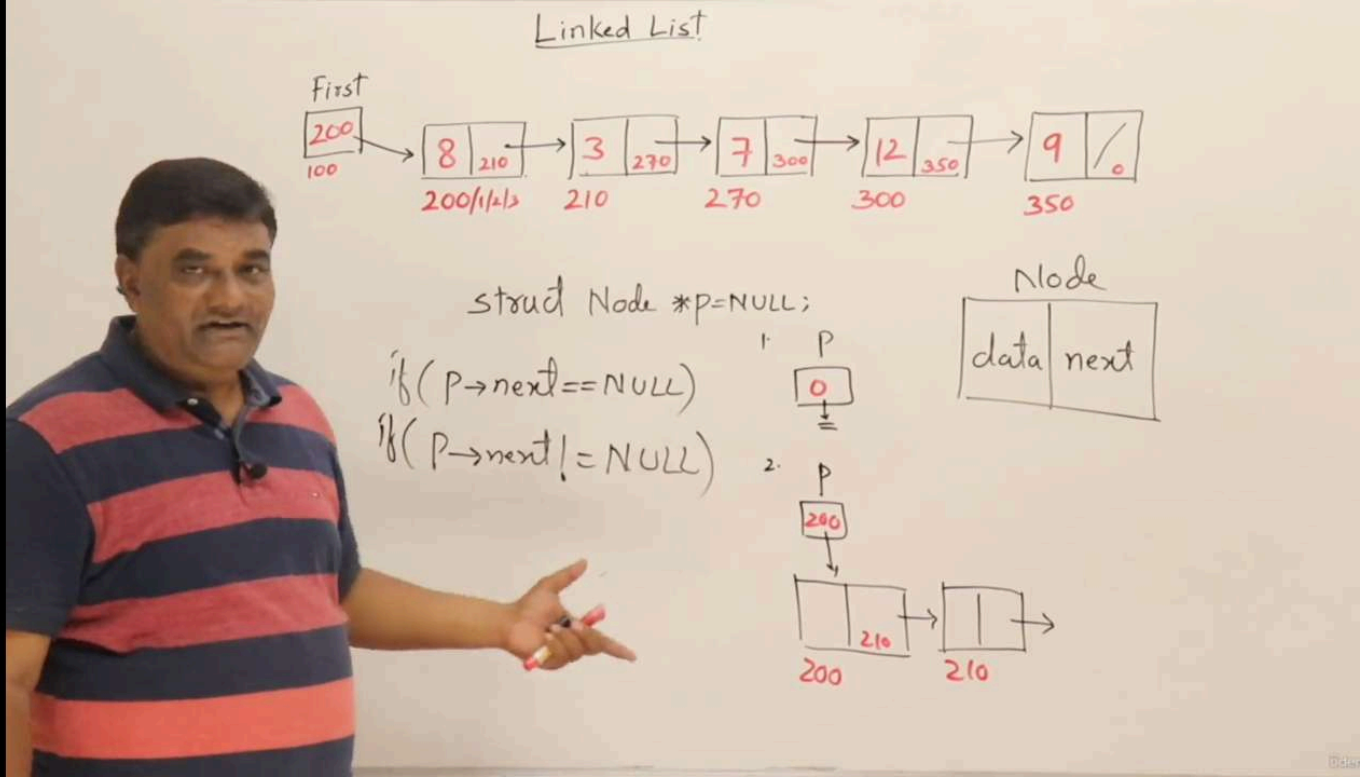
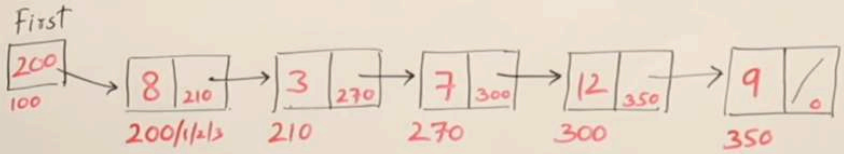

Linked List
First
200
100
8
210
200
3
270
210
7
300
270
12
/
0
300
1. what is Linked List
2. what is a Node
3. Node Structure
4. Create a Node
5. Access Node
A
0 1 2 3 4
p
First
main


Linked List
First
200
100
8
210
200/1/2/3
3
270
210
7
300
270
12
350
300
9
/
0
350
struct Node *p;
P=new Node;
P→data=10;
P→next=0;
P
500
10
/
0
500
Node
data
next
struct Node
{
int data;
Struct Node *next;
};


Linked List
First
200
100
8 210
3 270
7 300
12 350
9 /
0
200/1121
210
270
300
350
struct Node *p=NULL;
if(p==NULL)
if(p==0)
if(!p)
if(p!=NULL)
if(p!=0)
if(p)
1. p
0
2. p
200
210
200
210
Node
data next

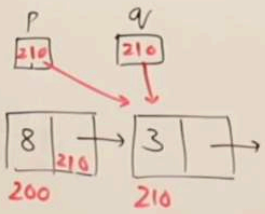

Linked List
First
200
100
8 210
3 270
7 300
12 350
9 /0
200/1121
210
270
300
350
struct Node *p=NULL;
if(P->next==NULL)
if(P->next!=NULL)
Node
data next
1.
P
0
2.
P
200
210
200
210

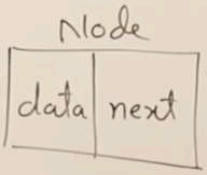
Linked List

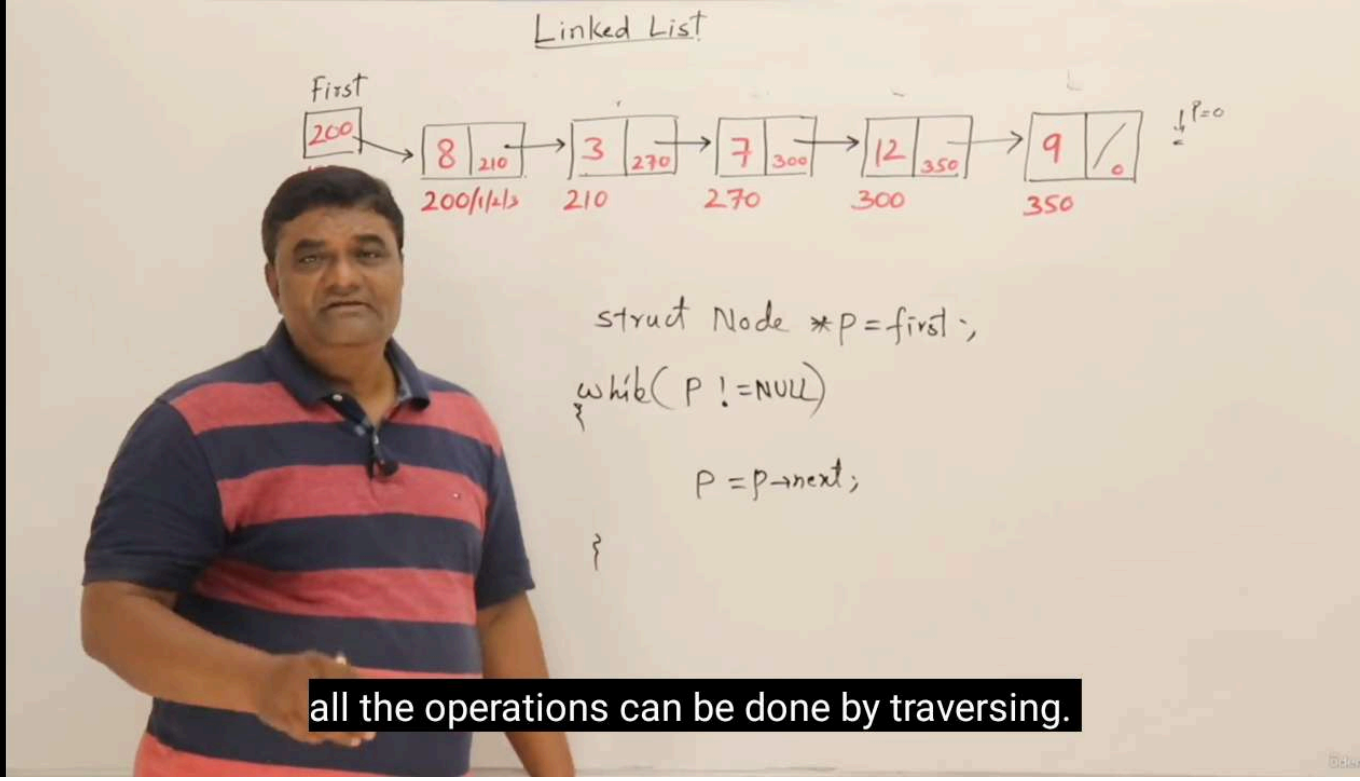


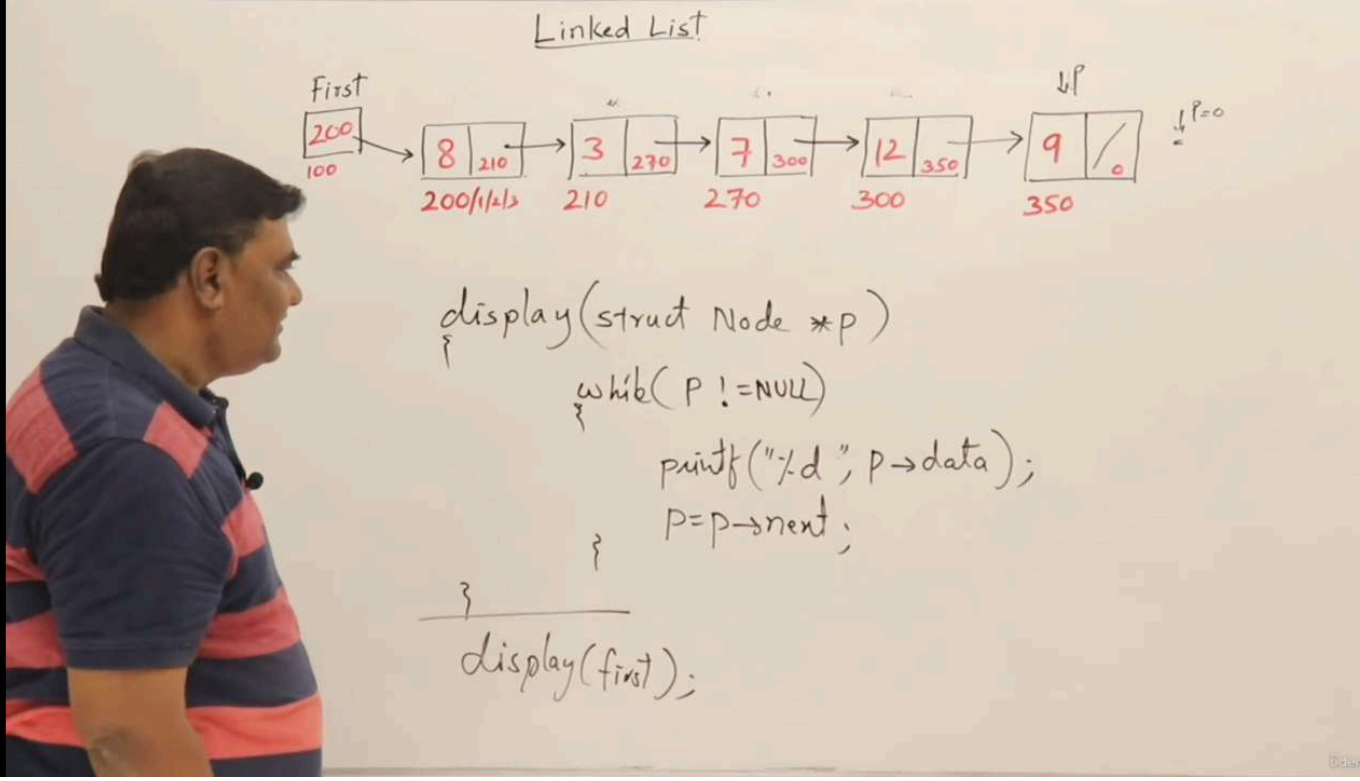
struct Node *p, *q;

1. q = p;
2. q = p->next;
3. p = p->next;






Linked List
First
200
8
210
3
270
7
300
12
350
9
200/1141
210
270
300
350
struct Node *p=first;
while(p!=NULL)
p=p->next;
all the operations can be done by traversing.


Linked List
First
200
100
8
210
200/1121
3
270
210
7
300
270
12
350
300
9
0
350
P=0
display(struct Node *p)
{
while(P !=NULL)
{
printf("%d", P->data);
P=P->next;
}
}
display(first);