

```
#include <iostream>

using namespace std;

template <class T>class Node{
public:
    T data;
    Node<T>* next;
    Node(int val){
        data=val;
        next=NULL;
    }
};

template <class T>class Stacks{
public:
    Node<T>*current=NULL;
    void push(int put){
```

```

    Node<T>* node=new
Node<T>(put);
    if(empty()){
        current=node;
        current->next=NULL;
    }else{
        node->next=current;
        current=node;
    }
}

void pop(){
    if(current==NULL){
        cout<<"Stack is under-flow
"<<endl;
    }else{
        int data=current->data;

```

```
    Node<T>*temp=current;
    current=current->next;
    cout<<temp->data<<endl;
    delete temp;
}
}
void clear(){
    current=NULL;
}
void display(){
    Node<T>*temp=current;
    if(!empty()){
        while(temp!=NULL){
            cout<<temp->data<<" "<<endl;
            temp=temp->next;
        }
    }
}
```

```

        }else{
            cout<<"stack is underflow
"<<endl;
        }
    }
    bool empty(){
        return current==NULL;
    }
};

int main() {
    cout<<"1.Push"<<endl;
    cout<<"2.Pop"<<endl;
    cout<<"3.Clear"<<endl;
    cout<<"4.Exit"<<endl;
    bool ch= true;
    int choice,val;

```

```
Stacks<int> s;
while(ch){
    cout<<"Enter choice: ";
    cin>>choice;
    switch (choice){
        case 1:
            cout<<"Enter number: ";
            cin>>val;
            s.push(val);
            break;
        case 2:
            s.pop();
            break;
        case 3:
            s.clear();
            break;
```

case 4:

ch= **false**;

break;

default:

cout<<" Wrong choice "<<endl;

}

}

}

