//21co24 ETHAN MENEZES COMP A

#include<stdio.h>

#include<malloc.h>

struct node //struct definition of the node

{

int info;

struct node\* next;

};

typedef struct node node; //typedef to use node alias instead

void display(node \*); //function declarations

node\* push(node \*,int);

node\* pop(node \*);

void peek(node \*);

int main() //menu to offer different options to the user

{

node \*top;

top=NULL;

int c;

while(c!=0)

{

printf("\n1.Push element in the stack\n2.Pop element from the stack\n3.Display stack\n4.Display the peek\n");

scanf("%d",&c);

switch(c)

{

case 1:

{

int elem;

printf("Enter the element to be inserted\n");

scanf("%d",&elem);

top=push(top,elem);

}

break;

case 2:

{

top=pop(top);

}

break;

case 3:display(top);

break;

case 4:peek(top);

break;

case 0:printf("Exiting");

}

}

}

node\* push(node \*top,int item) //push element in the stack

{

node \*tmp;

tmp=(node\*)malloc(sizeof(node));

tmp->info=item;

tmp->next=NULL;

if(top==NULL) //when stack is empty

{

top=tmp;

printf("%d has be successfully inserted in the stack\n",item);

return top;

}

tmp->next=top;

top=tmp;

printf("%d has be successfully inserted in the stack\n",item);

return top;

}

node\* pop(node \*top) //pops element in the stack

{

if(top==NULL) //when stack is empty

{

printf("The stack is empty\n");

return top;

}

node \*tmp;

tmp=top;

top=top->next;

printf("%d has been deleted successfully\n",tmp->info);

free(tmp);

return top;

}

void display(node \*top) //display elements in the stack

{

if(top==NULL) //when stack is empty

{

printf("The stack is empty\n");

return;

}

node \*ptr;

ptr=top;

printf("Top--->");

while(ptr!=NULL)

{

printf("%d--->",ptr->info);

ptr=ptr->next;

}

printf("NULL\n\n");

}

void peek(node \*top) //displays the peek element in the stack

{

if(top==NULL) //when stack is empty

{

printf("The stack is empty\n");

return;

}

printf("%d Is the peek element\n",top->info);

return;

}

**Output:**

