//Singly linear linked list insertion deletion & display

//20UET009

#include<stdio.h>

#include<stdlib.h>

int main()

{

int c,ch,choice,count,pos;

struct node

{

int data;

struct node \*next;

};

struct node \*temp,\*newnode,\*head;

temp=head=NULL;

while(1)

{

printf("\nenter choice\ncreat newnode:1 delition:2 display:3 exit:4\n");

scanf("%d",&choice);

switch(choice)

{

case 1:printf("creation new node\n");

newnode=(struct node\*) malloc(sizeof(struct node));

printf("enter node data\n");

scanf("%d",&newnode->data);

newnode->next=NULL;

if(head==NULL)

{

head=newnode;

temp=newnode;

}

else

{

temp->next=newnode;

temp=temp->next;

}

break;

case 2:printf("\nDeletion operation\n");

printf("1stnode:1 lastnode:2 position:3\n");

printf("enter choice\n");

scanf("%d",&ch);

struct node \*prevnode;

switch(ch)

{

case 1:printf("\n1stnode detetion\n");

temp=head;

head=temp->next;

free(temp);

break;

case 2:printf("\nlastnode detetion\n");

prevnode=NULL;

temp=head;

while(temp->next!=NULL)

{

prevnode=temp;

temp=temp->next;

}

prevnode->next=NULL;

free(temp);

break;

case 3:printf("\nposition node detetion\n");

prevnode=NULL;

temp=head;

count=1;

while(count!=pos)

{

if(temp==NULL)

{

printf("\nposition not in scope of list\n");

}

else

{

count ++;

prevnode=temp;

temp=temp->next;

}

}

prevnode->next=temp->next;

free(temp);

}

break;

case 3:

printf("\ndata in the the lists are:\n");

temp=head;

while(temp!=NULL)

{

printf("%d ",temp->data);

temp=temp->next;

}

case 4:

exit(0);

break;

}

}

return 0;

}

