DS LAB 5 Postlab

Name: Vivian Vijay Ludrick

Branch: SE Comps A Batch C

Roll No: 9914

Code:

#include <stdio.h>

#include <stdlib.h>

*// struct of car created*

typedef struct

{

char make[26];*// manufacturer name*

int year;

float price, mileage;

} Car;

*// Node of car created*

typedef struct node

{

Car data;

struct node \*next;

} Node;

*// LL created*

typedef struct

{

struct node \*start;

} LL;

*// insert*

void insert(LL \**l*, Car *c*)

{

Node \*p;

p = (Node \*)malloc(sizeof(Node));

p->data = *c*;

p->next = NULL;

*// put the p address in the last node*

if (*l*->start == NULL)

{

*// if first node*

*l*->start = p;

}

else

{

Node \*q;

q = *l*->start;

while (q->next != NULL) *// loop till end*

{

q = q->next;

}

q->next = p;

}

}

*// sorting a linked list*

void sort(LL \**l*)

{

if (*l*->start == NULL || *l*->start->next == NULL)

{

*// if one or no node*

return;

}

*//bubble sort*

int swapped;

Node \*p;

Node \*q = NULL;

do

{

swapped = 0;

p = *l*->start; *// pointer p is used for traversing*

while (p->next != q)

{ *// if the mileage in current node is less than mileage in next node*

if (p->data.mileage < p->next->data.mileage)

{

*// Swap the cars*

Car temp = p->data;

p->data = p->next->data;

p->next->data = temp;

swapped = 1;

}

p = p->next; *// incrementing pointer p*

}

q = p; *// address of p is assigned to q*

} while (swapped);

}

*// display the linkedList*

void display(LL *l*)

{

Node \*q = *l*.start;

if (q == NULL)

{

printf("Linked list is empty\n");

}

else

{

printf("The Linked List is:\n");

*//loop till end*

while (q != NULL)

{

printf("Details of car:\n\tMake :\t % s\n\tYear :\t % d\n\tPrice :\t % f\n\tMileage :\t % f\n ", q->data.make, q->data.year, q->data.price, q->data.mileage);

q = q->next;

}

}

}

*// start of main method*

int main()

{

LL l1; *// object creation*

Car c1, c2, c3;

l1.start = NULL; *// initializing start pointer to NULL*

int option;

*// Menu*

do

{

printf("\nEnter a option:1.Insert for car1 2.insert for car2 3.Insert for car3 4.Display 5.Sort 6.Exit: ");

scanf("%d", &option);

switch (option)

{

case 1:

printf("Enter details of car 1:\n");

printf("\tMake:\t");

scanf("%s", c1.make);

printf("\tYear:\t");

scanf("%d", &c1.year);

printf("\tPrice:\t");

scanf("%f", &c1.price);

printf("\tMileage:\t");

scanf("%f", &c1.mileage);

insert(&*l1*, c1);

break;

case 2:

printf("Enter details of car 2:\n");

printf("\tMake:\t");

scanf("%s", c2.make);

printf("\tYear:\t");

scanf("%d", &c2.year);

printf("\tPrice:\t");

scanf("%f", &c2.price);

printf("\tMileage:\t");

scanf("%f", &c2.mileage);

insert(&*l1*, c2);

break;

case 3:

printf("Enter details of car 3:\n");

printf("\tMake:\t");

scanf("%s", c3.make);

printf("\tYear:\t");

scanf("%d", &c3.year);

printf("\tPrice:\t");

scanf("%f", &c3.price);

printf("\tMileage:\t");

scanf("%f", &c3.mileage);

insert(&*l1*, c3);

break;

case 4:

display(l1);

break;

case 5:

sort(&*l1*);

break;

case 6:

break;

default:

printf("Enter valid option!!!\n");

break;

}

} while (option != 6); *// infinite loop until exit code entered*

return 0;

}

OUTPUT:

