/\*gavin skehan

21440824

22/02/2022\*/

#include <stdio.h> // libary's

#include <stdlib.h>

#include <string.h>

typedef struct { // sttucture for cars information

char make[20];

char model[20];

int year;

}car;

car garage[10]; // array garage

void readCars(char myfilePath[], int numCars); // functions with paramters

void displayGarage(int numCars);

int checkYear(int numCars, int year);

void main() {

char myfilePath[] = "C:\\Users\\skeha\\Downloads\\carsYear (1).csv"; // csv file from excel

readCars(myfilePath, 10); // calling functions in main

displayGarage(10);

int ans = checkYear(10, 2016);

printf("There are %d cars with year %d in the garage.\n", ans, 2016); // prints amount of cars from 2016

}

void readCars(char myfilePath[], int numCars)

{

FILE\* fptr; // calling the file pointer

car c; // using structure

int curCar = 0;

char line[200];

char delims[] = ","; // both used to separate strings

char delimspace[] = " ";

fopen\_s(&fptr, myfilePath, "r"); // open file in read mode

if (fptr != NULL) {

while (!feof(fptr) && curCar < numCars) {

fgets(line, 200, fptr);

char\* next = NULL;

char\* first = strtok\_s(line, delimspace, &next); // separates strings when space

strcpy\_s(c.make, 20, first); // copy strings

first = strtok\_s(NULL, delims, &next);

strcpy\_s(c.model, 20, first);

first = strtok\_s(NULL, delims, &next);

c.year = atoi(first); // convert int to string

strcpy\_s(line, 200, fptr); // copy string

curCar++; // increment

garage[curCar] = c; // store in array garage

}

fclose(fptr); // close file

}

}

void displayGarage(int numCars)

{

for (int i = 1; i <= numCars; i++) {

car c = garage[i]; // use file structure

printf("\nCar make = %s\nCar model = %s\nCar year = %d\n", c.make, c.model, c.year);

}

}

int checkYear(int numCars, int year)

{

int counter = 0;

for(int i = 1; i <= numCars; i++) {

car c = garage[i];

if (c.year == year) { // find cars in the year 2016

counter++;

}

}

return counter;

}

A picture containing text

Description automatically generated

Text

Description automatically generated