

Programozás TopCars2022

Ebben a feladatban az elmúlt néhány évben gyártott luxus sportkocsik adatai állnak rendelkezésünkre, melyekkel programozási feladatokat kell megoldania.

A topcars , txt UTF-8 kódolású forrásállomány soraiban egy-egy autótípus adatait tároltuk a következő sorrendben:

'Típus: Bugatti Veyron Vivere By Mansory

'Teljesítmény [LE], például: 12 0 0

'Ár [millió USD], például: 3 . 4

'Gyártott/tervezett darabszám, például: 2

'Leírás, például: A Bugatti Veyron a Bugatti cég legnépszerűbb modellje...

Az állomány első sora a mezőneveket tartalmazza, az adatokat pontosvesszővel választottuk el:

```
Típus;Teljesítmény;Ár;Darabszám;Leírás
Bugatti Veyron Vivere By Mansory;1200;3.4;2;A Bugatti Veyron a Bugatti cég
W Motors Lykan Hypersport;780;3.4;7;A Lykan Hypersport 2013-ban lett bemuta
Pagani Huayra Roadster BC;745;3.5;40;A Huayra Roadster BC a Huayra pályaori
Bugatti Chiron Pur Sport;1500;3.6;60;Akinek a Chiron nem elég különleges, a
...
```

- 1, Készítsen grafikus vagy konzolalkalmazást (projektet) a következő feladatok megoldásához, amelynek projektjét TopCars2 0 22 néven mentse el!
- 2 Hozzon létre saját osztályt SportCar azonosítóval, melynek adatait felhasználva egy-egy autótípus adatait tudja majd tárolni! A feladatot megoldhatja saját osztály definiálása nélkül is, de megoldása úgy nem lehet teljes értékű.
- 3, Készítse el a saját osztálykonstruktorát, mely hívásával az osztály adatait tudja inicializálni! Ügyeljen arra, hogy a tizedes szeparátor az „Ar” adatban a konverzió problémát okozhat.
- 4, Olvassa be a topcars . txt állomány sorait és tárolja az adatokat egy SportCar osztálytípusú összetett adatszerkezetben! Ügyeljen arra, hogy az állomány első sora az adatok fejlécét tartalmazza!
5. Határozza meg és írja ki az ablakba, hogy hány olyan autótípus van, melynek ára 5 millió USD feletti!
- 6 Határozza meg, hogy a gyártók autónként átlagosan mennyi bevételre tesznek szert az összes autó eladása után! Az eredményt három tizedesjegy pontossággal jelenítse meg!
7. Készítsen a SportCar osztályba LeírásRövid azonosítóval jellemzőt vagy függvényt, amely az eredeti leírásadat elejétől maximum 50 karakter hosszúságú leírással tér vissza úgy, hogy az utolsó szót nem csonkolja! Az írásjelek az előttük álló szó hosszát 1-gyel növelik. Feltételezheti, hogy a leírás legalább két szóból áll, és az első szó hossza 50 karakternél kisebb.
- 8, Határozza meg a legerősebb (teljesítmény alapján) autó adatait! A rövid leírást csak akkor írja ki, ha az előző feladatot meg tudta oldani!
9. Készítsen statisztikát az eredeti leírásokban lévő szavak számáról! A statisztikában ne vegye figyelembe az „a”, „A”, „az”, „Az”, „és”, „”, „;”, „ez”, „Ez” szavakat! Az első három legtöbbet előforduló szót és gyakoriságát jelenítse meg! Feltételezheti, hogy a 3. és a 4. helyen nem alakult ki holtverseny.
- 100 Hozzon létre topcars new. txt azonosítóval szöveges állomány! Az állomány fejlécsorát és tartalmát a minta szerint alakítsa ki! Az árakat forintban, a teljesítményt kilowattban írja az új állományba! Az átváltáshoz deklaráljon és használjon konstansokat programjában: USDFT = 332 . 87 és LEKW = 0 . 735! Az átváltott értékek a matematika szabályai szerint (.5-től fölfelé kerekítve) egészre kerekítve, ezres szeparálással kerüljenek az állományba!

Képernyőminta:

```
5. feladat: Autótípusok száma 5 millió USD felett: 7 db
6. feladat: A gyártók átlagos bevétele: 4,404 millió USD
8. feladat: A legerősebb autó adatai
Típus: Bugatti Centodieci
Teljesítmény: 1600 LE
Ár: 9 millió USD
Darabszám: 10
Rövid leírás: A Bugatti Centodieci minden szempontból egy
9. feladat: Leírás statisztika
autó - 13 db
Bugatti - 10 db
motor - 8 db
```

Minta topcars_new.txt:

```
Típus;Teljesítmény_kW;Ár_HUF;Darabszám;Leírás
Bugatti Veyron Vivere By Mansory;882;1 131 758 000;2;A Bugatti
W Motors Lykan Hypersport;573;1 131 758 000;7;A Lykan Hyperspor
Pagani Huayra Roadster BC;548;1 165 045 000;40;A Huayra Roadste
Bugatti Chiron Pur Sport;1 103;1 198 332 000;60;Akinek a Chiron
Lamborghini Sián;588;1 198 332 000;63;A Lamborghini Sián 2019 u
...
```

Megoldás:

```
using System;
using System.IO;
using System.Collections.Generic;
using System.Linq;
using System.Text;
```

```

using System.Threading.Tasks;
/*1, Készítsen grafikus vagy konzolalkalmazást (projektet) a következő feladatok megoldásához,
 * amelynek projektjét TopCars2022 néven mentse el!*/
namespace TopCars2022
{
    /*2 Hozzon létre saját osztályt SportCar azonosítóval,
     * melynek adattagjait felhasználva egy-egy autótípus adatait tudja majd tárolni!
     Típus;Teljesítmény;Ár;Darabszám;Leírás
Bugatti Veyron Vivere By Mansory;1200;3.4;2;A Bugatti Veyron a Bugatti cég legnépszerűbb
modellje. A Chiron elődje óriási ikonnak számít, a Mansory változat pedig azért különleges,
mert technikailag nem a Bugatti gyártotta. A Vivere By Mansory a Mansory tuningcég által
készült.
W Motors Lykan Hypersport;780;3.4;7;A Lykan Hypersport 2013-ban lett bemutatva, a dubai cég
első autójaként került a piacra. A kocsirajongók izgatottan várták, ugyanis ez volt az első
szupersport autó, amelyet Közel-Keleten gyártottak. Megjelenésekor a 3. legdrágább gyártásban
levő autóként indult, azonban a pénzért cserébe olyan extrákat is kaptál, mint például a
hologramos kijelző és a gyémánttal kiegészített LED lámpák.*/
    class SportCar
    {
        public string tipus { get; set; }
        public int teljesitmeny { get; set; }
        public double ar { get; set; }
        public int darabszam { get; set; }
        public string leiras { get; set; }
        public string leirasrovid { get; set; }

        /*7. Készítsen a SportCar osztályba LeírásRövid azonosítóval jellemzőt vagy függvényt,
         * amely az eredeti leírásadat elejétől maximum 50 karakter hosszúságú leírással tér
        vissza úgy,
         * hogy az utolsó szót nem csonkolja! Az írásjelek az előttük álló szó hosszát 1-gyel
        növelik. Feltételezheti,
         * hogy a leírás legalább két szóból áll, és az első szó hossza 50 karakternél
        kisebb.*/
        static string LeirasRovid(string szoveg)
        {
            string[] tomb = szoveg.Split(' ');
            var s = new StringBuilder();
            int i = 0;
            while (s.Length<50)
            {
                s.Append(tomb[i]+" ");
                i++;
            }

            return s.ToString();
        }
        /*3, Készítse el a saját osztálykonstruktorát, mely hívásával az osztály adattagjait
        tudja inicializálni!
         * Ügyeljen arra, hogy a tizedes szeparátor az „Ar” adatban a konverziónál problémát
        okozhat.*/
        public SportCar(string sor)
        {
            string[] sorelemek = sor.Split(';');
            this.tipus = sorelemek[0];
            this.teljesitmeny = Convert.ToInt32(sorelemek[1]);
            this.ar = Convert.ToDouble(sorelemek[2].Replace(".", ","));
            this.darabszam = Convert.ToInt32(sorelemek[3]);
            this.leiras = sorelemek[4];
            this.leirasrovid = LeirasRovid(sorelemek[4]);
        }
    }
}
class Program
{
    public static List<SportCar> adatok = new List<SportCar>();//adatok listában tárolom
az adatokat
    static void Main(string[] args)
    {
        /*4, Olvassa be a topcars . txt állomány sorait és tárolja az adatokat egy

```

```

    * SportCar osztálytípusú összetett adatszerkezetben!
    * Ügyeljen arra, hogy az állomány első sora az adatok fejlécét tartalmazza!*/
string[] fajl = File.ReadAllLines("topcars.txt", Encoding.UTF8); //összes sor
beolvasása
for (int i = 1; i < fajl.Length; i++) //az első sor a mezőneveket tartalmazza
{
    adatok.Add(new SportCar(fajl[i])); //listába
}
int autokszama = adatok.Count();
//Console.WriteLine("Adatok száma: {0}", autokszama);
//5. Határozza meg és írja ki az ablakba, hogy hány olyan autótípus van, melynek
ára 5 millió USD feletti!
//megszámolás tétele
int db = 0;
for (int i = 0; i < autokszama; i++)
{
    if (adatok[i].ar > 5) db++;
}
Console.WriteLine("5. feladat: Autótípusok száma 5 millió USD felett: {0} db",
db);

//másik megoldás
db = 0;
adatok.Where(x => x.ar > 5).ToList().ForEach(x => db++);
Console.WriteLine("5. feladat: Autótípusok száma 5 millió USD felett: {0} db",
db);

//6 Határozza meg, hogy a gyártók autónként átlagosan mennyi bevételre tesznek
szert az összes autó eladása után!
//Az eredményt három tizedesjegy pontossággal jelenítse meg!
//összegzés tétele
double ossz = 0;
db = 0;
for (int i = 0; i < autokszama; i++)
{
    ossz += adatok[i].ar * adatok[i].darabszam;
    db += adatok[i].darabszam;
}
Console.WriteLine("6. feladat: A gyártók átlagos bevétele: {0:0.000} millió USD",
ossz / db);

//8, Határozza meg a legerősebb (teljesítmény alapján) autó adatait!
//A rövid leírást csak akkor írja ki, ha az előző feladatot meg tudta oldani!
Console.WriteLine("7. feladat: A legerősebb autó adatai");
List<SportCar> rendezett = new List<SportCar>(); //sorbarendezt lista
rendezett = adatok.OrderByDescending(x => x.teljesitmeny).ToList(); //teljesítmény
szerint csökkenő sorrend
Console.WriteLine("\tTípus: {0}", rendezett[0].tipus); //az első azaz a 0. adatot
kell kiírni
Console.WriteLine("\tTeljesítmény: {0} LE", rendezett[0].teljesitmeny);
Console.WriteLine("\tÁr: {0} millió USD", rendezett[0].ar);
Console.WriteLine("\tDarabszám: {0}", rendezett[0].darabszam);
Console.WriteLine("\tRövid leírás: {0}", rendezett[0].leirasrovid);

/*9. Készítsen statisztikát az eredeti leírásokban lévő szavak számáról!
* A statisztikában ne vegye figyelembe az „a”, „A”, „az”, „Az”, „és”, „ ”, „; ”
ez”, „Ez” szavakat!
* Az első három legtöbbet előforduló szót és gyakoriságát jelenítse meg!
Feltételezheti,
* hogy a 3. és a 4. helyen nem alakult ki holtverseny.*/
Console.WriteLine("9. feladat: Leírás statisztika");
string[] nemSzavak = { "a", "A", "az", "Az", "és", "egy", "is", "ez", "Ez"
}; //ezeket a szavakat nem számoljuk
List<string> szavakLista = new List<string>(); //az összes szó listája
bool van = false;
string[] szavaktomb;
foreach (var item in adatok)
{

```

```

        szavaktomb = item.leiras.Split(' ');//egy autó leírás bekerül a szavaktomb-be
(a szóközök nem) szöveg-->szavak
        foreach(var d in szavaktomb)//végignézem a szavakat
        {
            van = false;
            for (int j = 0; j < nemSzavak.Length; j++)//a nemszavak-on végigmegyek
            {
                if(d==nemSzavak[j]) van = true;//ha a szó szerepel a nemszavak-ban
            }
            if (van == false) szavakLista.Add(d);//listába rakom a szavakat (nemszavak
nélkül)
        }
    }
    szavakLista.GroupBy(x => x).OrderByDescending(x => x.Count()).
        Where(x=>x.Count(>6).ToList().ForEach(x => Console.WriteLine("\t{0}-{1} db",
x.Key, x.Count()));//kiíratás
    //másik megoldás
    /*List<string> szavak = new List<string>();
    szavakLista.GroupBy(x => x).ToList().ForEach(x => szavak.Add(x.Key));
    Dictionary<string,int > dic = new Dictionary<string,int>();
    foreach(var adat in szavakLista)
    {
        if (dic.ContainsKey(adat))
        {
            dic[adat] ++;
        }
        else
        {
            dic.Add(adat, 1);
        }
    }
    foreach (var adat in dic.OrderByDescending(x=>x.Value).
        Where(x=>x.Value>6)) Console.WriteLine("\t{0}-{1} db", adat.Key,
adat.Value);*/

    /*100 Hozzon létre topcars new. txt azonosítóval szöveges állomány!
    * Az állomány fejlécsorát és tartalmát a minta szerint alakítsa ki!
    * Az árakat forintban, a teljesítményt kilowattban írja az új állományba!
    * Az átváltáshoz deklaráljon és használjon konstansokat programjában:
    * USD FT = 332 . 87 és LE KW = 0 . 735!
    * Az átváltott értékek a matematika szabályai szerint (.5-től fölfelé kerekítve)
    * egészen kerekítve, ezres szeparálással kerüljenek az állományba!*/
    double USD_FT = 332.87;
    double LE_KW = 0.735;
    FileStream fnev = new FileStream("topcars_new.txt",
    FileMode.Create);//létrehozásra megnyitjuk a fájlt
    StreamWriter fajlbairo = new StreamWriter(fnev,
    System.Text.Encoding.UTF8);//fájlbairáshoz
    fajlbairo.WriteLine("Típus;Teljesítmény_KW;Ár_HUF;Darabszám;Leírás");//első sor
    beírása a fájlba
    foreach (var item in adatok)
    {
        fajlbairo.WriteLine("{0};{1};{2:0,0};{3};{4}",
            item.tipus,Math.Round(item.teljesitmeny*LE_KW),
            Math.Round(item.ar*USD_FT*1000000), item.darabszam, item.leiras);
    }
    fajlbairo.Close();//fájl lezárása
    fnev.Close();

    Console.ReadKey();
}
}
}

```