

# Autók - Dokumentáció

## Tartalom

Program feladata .....	2
Függvények, eljárások .....	2
Függvénystruktúra .....	3
nyolcfeletti .....	3
evjarat .....	3
megszámolás .....	4
kivalogatas .....	4
minmax .....	4
Tesztelés .....	5
Általános tesztesetek .....	5
Speciális tesztesetek .....	5
Lehetséges hibák .....	6
Fejlesztési ötletek .....	7

## Program feladata

A `program.py` program több szempont szerint listázott autók adatait dolgozza fel. A program az adatokat szöveges fájlból fogadja, melynek felépítése szigorúan kötött. A fájl minden sorában pontosan 4 adat szerepelhet, egymástól szóközzel elválasztva:

1. **Autó márkája:** az adott autó márkája
2. **Rendszám:** az autó rendszáma (az angol abc betűi háromszor véletlenszerűen, -, számok 1-9-ig véletlenszerűen háromszor)
3. **Évjárat:** az autó évjárata (2000-2023-ig)
4. **Autó értékelése:** az adott autó értékelése (lebegő pontosan 0-10ig érték)

Helyes használat és sikeres beolvasás esetén a program **a következő kérdésekre ad választ:**

1. Mi a rendszáma annak az autónak aminek 8.0 felett van az értékelése (az első ilyen esett)?
2. Bekérsz egy szám adatot (évjárat) a felhasználtól, és eldöntöd hogy szerepel-e az évjáratok közt ilyen?
3. Ford márkájú autók darab száma?
4. A Felhasználó által megadott autó márkából a legújabbakat kell kiválogatni (2020 feletti), (a rendszámuk legyen a visszatérés érték)?
5. A felhasználó által bekért autó legkisebb értékelésnek és a legnagyobb értékelésnek az átlagát kell megvalósítani!

## Függvények, eljárások

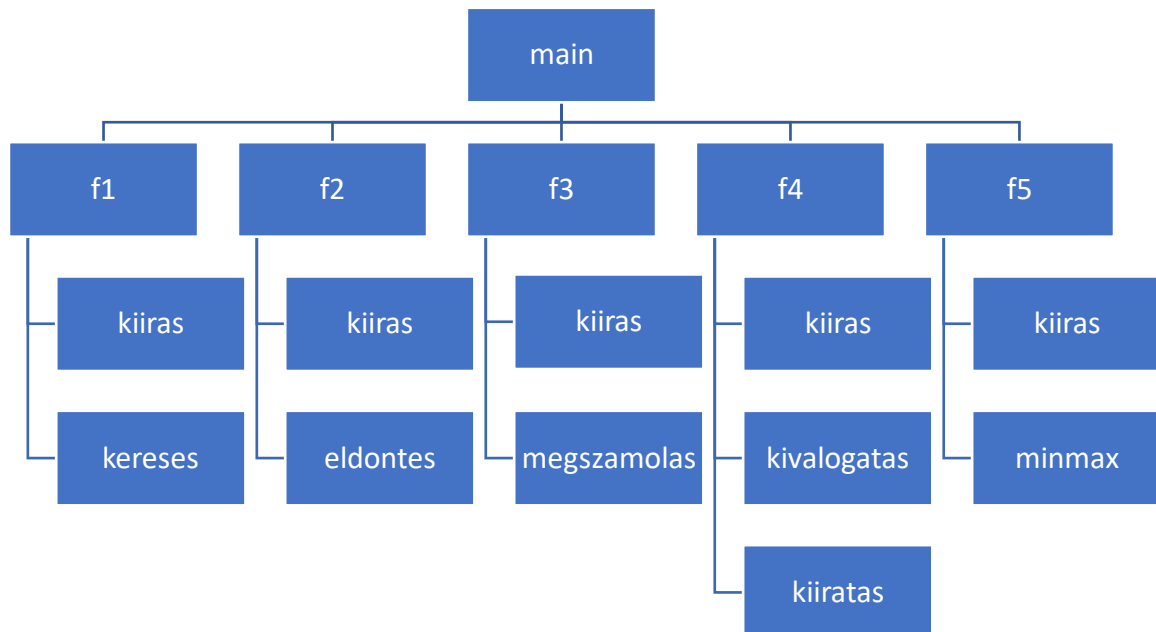
A program által használt modulok:

`program.py` – fő program

`generalas.py` – teszt adatokat előállító segédprogram

`random.py` – beépített modul random számok kezelésére

## Függvénystruktúra



### nyolcfeletti

**nyolcfeletti (rendszamok, ertekelesek): -> string, float**

Mi a rendszáma annak az autónak aminek 8.0 felett van az értékelése (az első ilyen esett)

**Paraméterek:**

rendszamok: autók rendszámainak listája (list)

ertekelesek: az autók értékeléseinek listája (list)

**Példa:**

**nyolcfeletti ([GBA-934, HUR-123, ERT-348], [2.2, 3.2, 1.2])**

### evjarat

**evjarat (evjaratok, kert\_evjarat): -> int, int**

Bekérsz egy szám adatott (évjárat) a felhasználótól, és eldöntöd hogy szerepel-e az évjáratok közt ilyen

**Paraméterek:**

evjaratok: az auto évjáratainak listája (list)

kert\_evjarat: a felhasználó által kért évjárat

**Példa:**

**evjarat ([2002, 2004, 2005], 2009)**

## **megszamolas**

**megszamolas (autonevek) -> string**

Ford márkájú autók darab száma

**Paraméterek:**

autonevek: az autó neveinek listája (list)

**Példa:**

**megszamolas ([Ford, Audi, Tesla])**

## **kivalogatas**

**kivalogatas (evjaratok, autonev, autonevek, rendszamok) - > int, string, string, string**

A Felhasználó által megadott autó márkából a legújabbakat kell kiválogatni (2020 felettiek), (a rendszámuk legyen a visszatérés érték)

**Paraméterek:**

evjaratok: az autó évjáraainak listája (list)

autonev: felhasználó által megadott autónév

autonevek: az autó neveinek listája (list)

rendszamok: az autók rendszámának listája (list)

**Példa:**

**kivalogatas ([2001, 2004, 2006], Ford, [Audi, Ford, Tesla], [RTE-456, DFA-321, SDG-589] -> [DFA-321]**

## **minmax**

**minmax (ertekelesek, kert\_auto, autonevek)**

A felhasználó által bekért autó legkisebb értékelésnek és a legnagyobb értékelésnek az átlagát kell megvalósítani!

**Paraméterek:**

ertekelesek: az autók értékeléseinek listája (list)

kert\_auto: felhasználó által bekért autó

autonevek: az autók neveinek listája (list)

**Példa:**

**minmax ([2.2, 3.2, 4.2], Ford, [Ford, Audi, Tesla] - > float**

## Tesztelés

### Általános tesztesetek

A teszteléshez egy 500 hosszú minta bemeneti fájlt készít a `generalas.py` program, amely sorai a `program.py` program által előírtaknak megfelelnek.

A neveket egy konstans listából véletlenszerűen sorsolja a program, egyenlő valószínűséggel.

Az autók neveinek kiválasztása előre beleégetett lista adatai közül kerül kiválasztásra véletlenszerűen (egyenlő valószínűséggel).

Az autók rendszámai az angol abc betűi háromszor véletlenszerűen, -, számok 1-9-ig véletlenszerűen háromszor.

Az autók évjáratái véletlenszerűen kerülnek kiválasztásra 2000-2023-ig. Az adott autó értékelése lebegő pontosan 0-10ig értékel egy tizedesjegy kerekítéssel.

Az `adatok.txt` fájl a `generalas.py` által előállított adatokat tartalmaz.

### Speciális tesztesetek

A `evjarat` függvény tesztelésére szolgálnak az alábbi állományok: `be1.txt`, `be2.txt`

A függvény működését a következő ekvivalencia osztályokra oszthatjuk:

1. Van a kért évjáratból (PI: 2003-as évjáratból van) (`be1.txt`)
2. Nincs a kért évjáratból (PI: 2003-as évjáratból nincs) (`be2.txt`)
3. Nem adott meg a felhasználó értéket.

## Lehetséges hibák

1. Nem létező fájlnev megadása.
2. Main függvénybe nem hívjuk meg a többi függvényt.
3. Számok megadásánál nem használunk „int-et”
4. Függvényeknél nem adunk meg jó paramétereket
5. Érthetetlen paraméterek megadása amin nem tudunk kiigazodni.
6. Szövegünktől eltérő kód.
7. Random függvényt nem hívjuk meg.

## Fejlesztési ötletek

1. Bekérhető legyen hogy hanyas értékelésű autót szeretnénk kiválogatni
2. Rendszám szerint növekvő vagy csökkenő sorrendbe felsorakoztatni az autókat
3. Saját megadott autót, rendszámát, évjárátát és saját véleményű értékelést az adat fájlba írni.
4. Új oszlopokkal teli adatok (eddiggi kilométer, ülések száma, dízeles/benzines)
5. 4-nél kisebb értékelésű autók törlése a fájlból