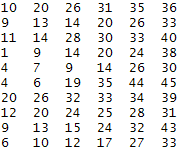
# 18. Fájlok 5.

## Lottó

Ebben a leckében a hatos lottóval kapcsolatos statisztikákat fogunk készíteni.

Az UTF-8 kódolású *hatos.txt* fájl a hatos lottó kihúzott számait tartalmazza 2005. januárjától 2012. júliusáig. Soronként vannak egy-egy hét számai (növekvő sorrendben, tabulátorokkal elválasztva):



A sorok száma 395. A számokat egy olyan kétdimenziós tömbben szeretnénk tárolni (szamok), amely 400 soros (hátha lesznek még adatok), és minden sorban 6 egész szám van.

A beolvasott sorok számát is tárolni kell, erre létrehozunk egy db nevű, egész típusú változót.

Mivel ezeket a változókat minden metódusból el kell tudni érni, az osztály elején statikus változóként adjuk meg őket.

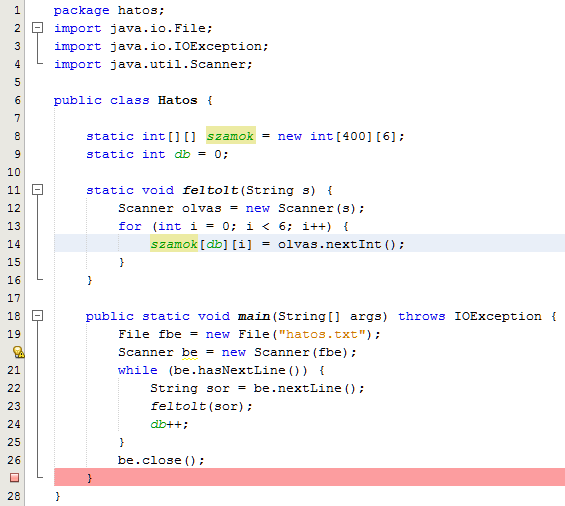
## Beolvasás

A program első változata beolvassa az adatokat a fájlból, és eltárolja őket a tömbben.

A szöveges fájl sorait a szokott módon olvassuk be, de a sorok feldolgozását a feltolt() metódussal végezzük.

Ebben létrehozunk egy másik Scanner objektumot, amely a kapott stringből olvas úgy, mintha a billentyűzetről írnák be a számokat. A beolvasott számokat elhelyezi a tömb megfelelő elemében.

Lássuk a programot! A törésponttal egyelőre ne foglalkozz, mindjárt lesz róla szó.



Készítsd el, és futtasd le a programot! Ne felejtsd el előtte a program mappájába másolni a hatos.txt fájlt!

## Ellenőrzés

A program futtatásakor a kimeneten nem jelenik meg semmi. Jó lenne megnézni, mi került a szamok tömbbe, de elég nehézkes lenne kiíratni az elemeket.

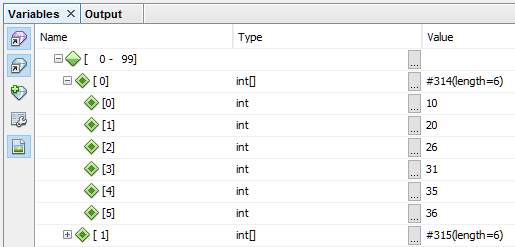
Ehelyett a hibakeresőt (debugger) fogjuk alkalmazni. Ennek indításához a Debug Project gombot vagy a Ctrl+F5 billentyűket kell megnyomni. Ha kipróbálod, akkor azt fogod tapasztalni, hogy a program ugyanúgy végigfut, mint eddig.

Meg kellene állítani a programot, és akkor megnézni a változók értékeit. A program megállítására szolgálnak a töréspontok.

Töréspontot a sor előtti sorszámra kattintva lehet elhelyezni vagy törölni. Helyezz el most egy töréspontot a 27. sorban! A rózsaszín sáv és a sorszám helyén a négyzet jelzi a bekapcsolt állapotot. Azért ebbe a sorba tettük, mert ez már az adatok beolvasása után van.

Indítsd el most a debug módot a Ctrl+F5 megnyomásával! A program lefut a 26. sorig, majd megáll. A NetBeans ablak alsó részén, a kimenetnél megjelenik a Variables fül. Válts át rá!

Kattints a static, majd a szamok előtti pluszra! A szamok tömbben nagyon sok érték van, ezért százanként csoportosítva jelennek meg. Jelenítsd meg a szamok[0]-ban lévő elemeket!



Vesd össze a számokat a *hatos.txt* első sorával! Ha mindent jól csináltál, akkor ezek megegyeznek. Ugyanígy végig lehetne nézni a többi sort is, de nagyon sokáig tartana. A biztonság kedvéért azért ellenőrizd az utolsó sort és az elemek számát (db)!

Az ellenőrzés után futtasd tovább a programot a zöld körben lévő nyíl vagy az F5 megnyomásával! Mivel nincs több utasítás, meg fog állni.

Végül töröld a töréspontot!

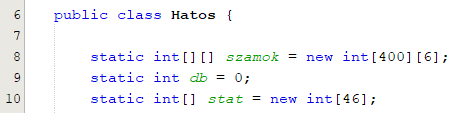
## Statisztika

Egy olyan statisztikát szeretnénk készíteni, amelyből kiderül, hogy melyik számot hányszor húzták ki az évek során. Ez egy kicsit hasonlít arra a feladatra, amikor nekünk kellett generálni a számokat.

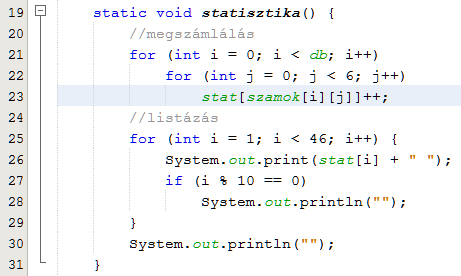
Ott egy olyan tömbbel modelleztük a szelvényt, amelynek minden eleme boolean típusú volt, és azt adta meg, hogy be van-e jelölve az adott szám.

Most egy olyan tömböt fogunk készíteni, amelynek minden eleme egy egész szám, és azt jelzi, hogy hányszor húzták már ki az adott számot.

A hatos lottón 45 számból húznak. A tömb 46 elemű lesz, a 0. elemet nem fogjuk használni.



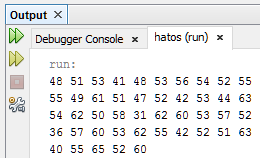
A statisztikát a statisztika() metódussal fogjuk elkészíteni. A metódus első felében megszámoljuk, melyik számot hányszor húzták ki, a második felében pedig kilistázzuk az eredményeket:



A szamok tömb elemein két for ciklussal tudunk végigmenni. A stat tömbben mindig azt az elemet növeljük eggyel, amelynek a szamok tömb adott eleme az indexe. Ha például a szamok i-dik sorának j-dik oszlopának értéke 23, akkor a stat[23]-hoz adunk hozzá egyet.

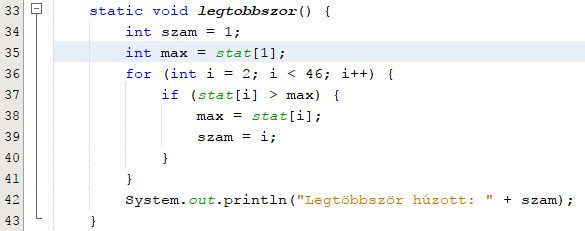
A listázásnál minden 10. elem után és a legvégén kiíratunk egy soremelést is.

Írd be a kódot és egészítsd ki a main() metódust a statisztika() metódus hívásával, majd próbáld ki a programot! Hogyan lehet értelmezni a kimenetet?

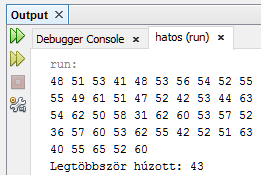


## Legtöbbször húzott szám

Ki szeretnénk íratni, hogy melyik számot húzták ki a legtöbbször. Ehhez meg kell keresnünk a stat tömb legnagyobb értékét, és ennek az indexét kell kiíratnunk:



Írd be a metódust, egészítsd ki a main() metódust a legtobbszor() metódus hívásával, majd próbáld ki a programot!



## Feladat

Egészítsd ki a programot úgy, hogy azt is írja ki, melyik számot húzták ki a legkevesebbszer! Melyiket?