# 28. Osztályok 9.

## Kép

Egy digitális kép tárolásánál minden egyes képpont színét tároljuk. A képpontok színét az RGB kód adja. Az RGB kód a vörös (R), zöld (G) és a kék (B) színösszetevő értékét határozza meg. Ezen színösszetevők értéke 0 és 255 közötti egész szám lehet.

A *kep.txt* fájlban egy 50×50 képpontos kép képpontjainak RGB kódjai vannak a következő formában. Az állomány a képet sorfolytonosan, a képpontok RGB kódját szóközzel elválasztva tartalmazza, minden képpontot egy újabb sorban:

200 96 64  
200 96 64  
200 96 64  
200 96 64  
200 96 64  
…

A mai feladatban ezt a képet fogjuk feldolgozni.

Nézd meg a *kep.txt* fájl tartalmát!

## Osztályok létrehozása

Először meg kell terveznünk, hogy milyen adatszerkezetben tároljuk az adatokat.

Minden képpont adatait egy-egy objektumban fogjuk tárolni, és ezekből hozunk majd létre egy kétdimenziós tömböt. Azért nem listát használunk, mert a méretek rögzítettek.

Az objektum típusa Keppont lesz. A Keppont osztály egy mezőt fog tartalmazni: a Szin mezőt, amely egy sornak felel meg a szöveges fájlokban. Tárolhatnánk külön-külön az egyes színösszetevőket, de mivel a feladatban nem kell velük külön dolgozni, a legegyszerűbb módon, egy stringben tároljuk őket.

A Keppont osztályt külön fájlba tesszük:

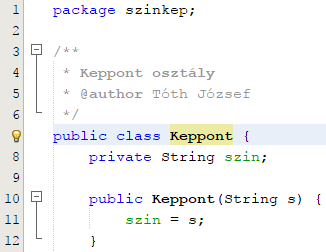
1. Először kezdj egy új projektet *szinkep* néven, a szokott módon!
2. A Projects ablakban kattints a jobb gombbal a *szinkep* csomagra, majd a menüből válaszd a New, Java Class... parancsot!
3. Az osztály neve legyen Keppont (nagy K-val!), majd kattints a Finish gombra!

Így most már két osztályból áll a programunk:

* A Keppont osztályban definiáljuk az egyes képpontok kezeléséhez szükséges adattagokat és metódusokat.
* A Szinkep osztályban van a main() metódus, a program indításakor az fog elindulni.

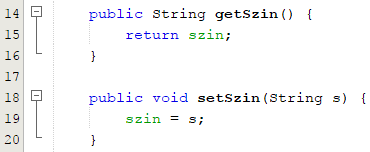
## A Keppont osztály

A Keppont osztályban először megadjuk az adattagot és elkészítjük a konstruktort. Írd be a következőket: (A szerző nevéhez írd a saját nevedet!)



Mivel a Keppont osztály példányait használjuk majd, itt nem kell a static szó sem az adattag, sem a metódusok elé.

A privát adattagokat publikus metódusokkal fogjuk kezelni. A szín lekérdezésére és beállítására ezt a két metódust használjuk:



Figyeld meg, hogy az adattag értékének beállításához set kezdetű metódusnevet adunk meg!

Mivel a Keppont osztály nagyon egyszerű, a teszteléstől kivételesen eltekintünk.

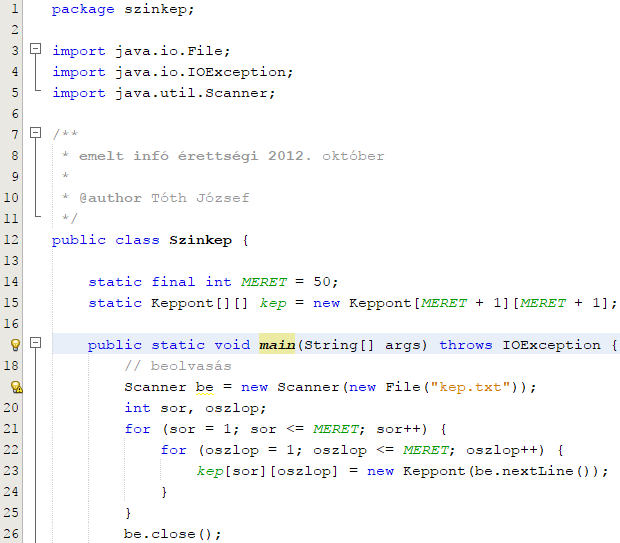
## Adatok beolvasása

Ezek után megírhatjuk a főprogramnak azt a részét, amely beolvassa az adatokat a fájlokból.

A kép mérete 50x50 képpont. Ezt a méretet egy állandóban rögzítjük.

Utána létrehozunk egy 51x51 elemű, képpontokból álló tömböt. A nulladik elemeket nem használjuk sem a sorokban, sem az oszlopokban.

Végül beolvassuk a 2500 képpont adatait a fájlból. Mivel tudjuk a méretet, két egymásba ágyazott for ciklust használhatunk. A külső for ciklus a sorokon, a belső az oszlopokon megy végig.



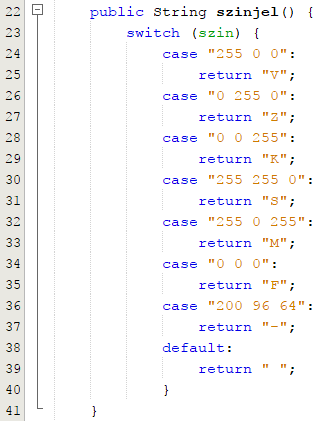
Ne felejtsd el a kipróbálás előtt a szöveges fájlt a projekt mappájába másolni!

## Kép megjelenítése

A beolvasás ellenőrzésére és a következő feladatok szemléltetésére valahogy meg kell jelenítenünk a képet.

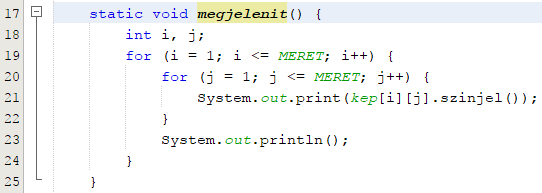
Rajzolni még nem tudunk, ezért az egyes képpontokat különböző betűkkel fogjuk jelölni. Például a vöröst V-vel, a zöldet Z-vel, és így tovább. Szerencsére a képen csak néhány szín szerepel.

Egészítsd ki a Keppont osztályt a következő metódussal:



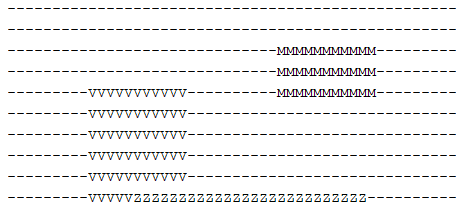
Miért nincs break az egyes ágak végén?

Ezután írd a főprogram elé a következő metódust:



A főprogramban hívd meg a megjelenit() metódust a beolvasás után!

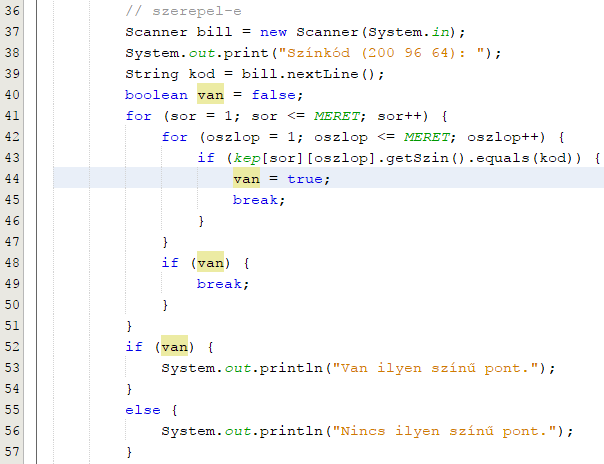
Próbáld ki!

  
…

## Szín szerepel-e

Kérj be egy színkódot, és írasd ki, hogy a megadott kódú szín szerepel-e a képen!

Ezt meg tudjuk oldani a főprogramban:



A beolvasás után a program sorban végigmegy a képpontokon. Ha talál egy adott színű pontot, akkor megszakítja a ciklusokat.

Egy ciklust a break utasítással lehet megszakítani. Ekkor a ciklus utáni első utasítással folytatódik a program.

A nehézséget az okozza, hogy ha a belső ciklust megszakítjuk egy break utasítással, akkor utána a külsőt is meg kell szakítani. Ezt a van változó vizsgálatával oldjuk meg.

A van változó értékétől függ az is, hogy a végén mit kell kiíratni.

Próbáld ki olyan színnel is, ami van a képen, és olyannal is, ami nincs!





A következő leckében folytatjuk.