# 8. Elágazás 1.

## Logikai kifejezések

Logikai kifejezésnek nevezzük azt a kifejezést, amelynek az eredménye igaz vagy hamis (true vagy false) lehet. Ezek valójában eldöntendő kérdések:

* a szám páros?
* a szám osztható 3-mal?
* a szám kisebb, mint 100?

Logikai kifejezést a legegyszerűbben két szám vagy kifejezés összehasonlításával készíthetünk.



Az összehasonlításhoz használható műveleti jelek:

|  |  |
| --- | --- |
| == | egyenlő |
| != | nem egyenlő |
| < | kisebb |
| <= | kisebb vagy egyenlő |
| > | nagyobb |
| >= | nagyobb vagy egyenlő |

Figyelj arra, hogy az egyenlőség vizsgálatához két egyenlőségjelet kell tenni! Ne keverd őssze az értékadással, amelyet egy egyenlőségjellel jelölünk!

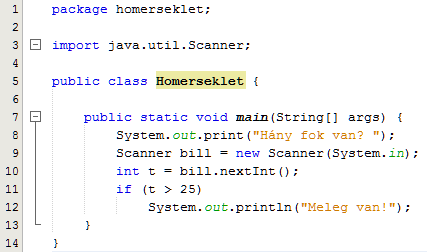
Logikai kifejezéseket elágazások és ciklusok feltételeiben használunk. Ismerkedjünk meg először az elágazásokkal!

## Az if utasítás

Eddigi programjainkban sorban egymás után hajtotta végre a Java az utasításokat. A logikai értékek (igaz, hamis) segítségével ezen változtathatunk: feltételt adhatunk meg, és ettől függ, hogy mely utasításokat kell végrehajtania.

Első ilyen programunk nagyon egyszerű lesz: Megkérdezi a hőmérsékletet (t), és ha ez 25 fok felett van, kiírja, hogy meleg van. Az egyszerűség kedvéért feltételezzük, hogy egész számot írnak be.

Kezdj egy új projektet *homerseklet* néven, és írd be a következőket:



A program a következőképpen működik: Ha a beolvasott szám (t) nagyobb, mint 25, kiírja a „Meleg van” szöveget. Figyeld meg, hogy az if után zárójelek között kell megadni a feltételt! A feltételesen végrehajtandó utasítást behúzással szokták írni, de ez nem kötelező.

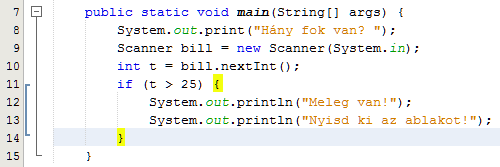
Figyeld meg, hogy az if (…) után nem teszünk pontosvesszőt!

Próbáld ki!



Mi történik, ha 25 fok alatti hőmérsékletet írsz be?

Egészítsd ki a programot úgy, hogy 25 fok feletti hőmérséklet esetén írja ki azt is, hogy „Nyisd ki az ablakot!”



Ha igaz feltétel esetén egynél több utasítást kell végrehajtani, akkor ezeket kapcsos zárójelek közé kell tenni. A nyitó kapcsos zárójelet általában az if sorának végére szokták tenni, de lehetne a következő sor elején is. A záró kapcsos zárójel beírásához nem kell a soron belül az if i-je alá állni, beíráskor a zárójel a megfelelő helyre ugrik. Figyeld meg, hogy a NetBeans a zárójel bezárásakor sárgával kiemeli a zárójel párokat!

Azt szokták javasolni, hogy egy utasítás esetén is tegyük ki a kapcsos zárójeleket, mert így biztosan nem felejtjük el őket, ha még egy utasítást hozzáadunk a feltételhez.

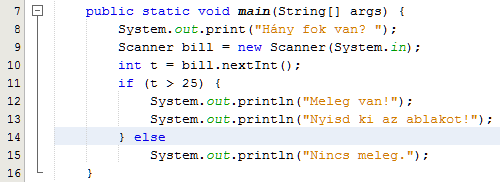
A kapcsos zárójelek közötti utasításokat utasításblokknak nevezzük. Bárhol, ahol egy utasítás állhat a programban, helyette ilyen utasításblokkot is írhatunk. Az utasításblokk után nem kell pontosvesszőt írni.

Ha egy blokkon belül deklarálunk egy változót, az csak a blokkon belül használható lokális (helyi) változó lesz.

Próbáld ki a programot!



Azt is megadhatjuk, hogy mi történjen, ha nem teljesül a feltétel. Egészítsd ki a programot úgy, hogy a hőmérséklet nem haladja meg a 25 fokot, akkor írja ki, hogy „Nincs meleg.” Így már elágazás lesz a programban, mert a t értékétől függően kell választani két irány között.



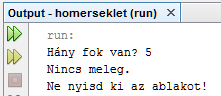
Azokat az utasításokat, amelyeket hamis feltétel esetén kell végrahajtani, az elágazás else (küönben, egyébként) ágába kell írni. Mivel most csak egy ilyen utasítás volt, nem használtunk kapcsos zárójeleket. Több utasítás esetén természetesen kell használni a kapcsos zárójeleket.

Próbáld ki az if utasítás mindkét ágát! Fontos, hogy a programok tesztelésénél minden esetet ki kell próbálni!

Tipp: Az if utasítás beírását megkönnyítheted, ha az if szó után megnyomod a Tab billentyűt. Ha else ágat is szeretnél, az ife szót írd be a Tab megnyomása előtt! Próbáld ki!



## Feladatok

1. Módosítsd a homerseklet programot úgy, hogy ha nincs 25 foknál melegebb, akkor írja ki azt is, hogy "Ne nyisd ki iaz ablakot!"  
   
2. Készíts programot *paros* néven, amely beolvas egy egész számot, majd kiírja, hogy a szám páros vagy páratlan! A programnak azt kell megvizsgálni, hogy a beolvasott szám kettővel osztva nulla maradékot ad-e (szam % 2 == 0)? Ha igen, akkor a szám páros, egyébként pedig páratlan. Teszteld a programot páros és paratlan számokkal is!  
   