# Alvás

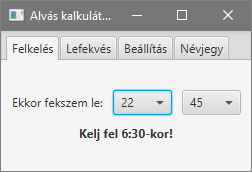
## Feladat

Alvás közben körülbelül másfél órás, vagyis 90 perces ciklusaink vannak. Ha egy ilyen ciklus végén ébredünk fel, akkor kevésbé leszünk álmosak, mintha egy ciklus belsejében ébredtünk volna fel. Ezt fogjuk felhasználni a mai programunkban.

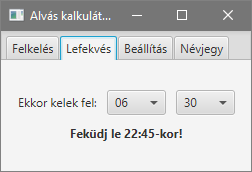
A programban kiszámoltatjuk, hogy mikor kell elaludnunk vagy felébrednünk úgy, hogy a 90 perc többszörösét aludjuk végig.

Tegyük fel, hogy lefekvés után 15 perc kell az elalváshoz, és 5 ciklusnyit (450 percet = 7,5 órát) szeretnénk aludni. Így összesen 465 perccel a felkelés időpontja előtt kell lefeküdnünk, illetve 465 perccel a lefekvés után kell felébrednünk.

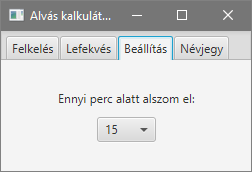
Ha tudjuk, hogy mikor fekszünk le, akkor kiszámíthatjuk, hogy mikor kell felkelnünk. Ezt fogja elvégezni a program első része:



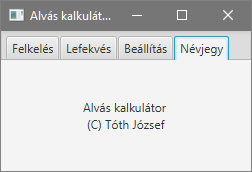
A program második része a lefekvés időpontját fogja kiszámítani az ébredés idópontja alapján:



Ezen kívül beállíthatjuk, hány perc kell az elalváshoz:



És végül lesz egy névjegy lap is:



## Előkészületek

Kezdj egy új JavaFX FXML projektet *alvascalc* néven!

Nyisd meg az FXML fájlt a Scene Builder-rel!

Töröld le az automatikusan létrehozott elemeket! (Mivel ezeknek van fx:id-jük, a törlés előtt figyelmeztet a program.)

Töröld a Controller fájlból is a felesleges kódot!

## Fülek elkészítése

Mint a fenti ábrákon látható, az ablakban négy fül van. Ezt a TabPane tárolóval lehet elkészíteni.

Ehhez az egyes füleket a hozzájuk tartozó lappal együtt Tab panelként kell hozzáadni, és ezek Text tulajdonságában kell megadni a fülek feliratait.

Az egyes oldalakon szereplő komponenseket a panelekhez kell hozzáadni.

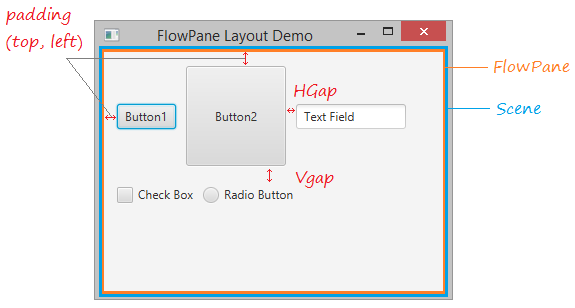
Helyezz el egy TabPane tárolót 250x140-es méretben, és rajta négy fület az ábrákon szereplő feliratokkal!

*Megjegyzés*: A TabPane tárolón alapból van két fül, csak két másikat kell hozzáadni. Sajnos a fülek hozzáadásánál a program néha lefagy, ezért előtte célszerű menteni. Ritkábban fagy le a program, ha a füleket a hierarchiához adod hozzá. Ha így sem sikerül, az FXML fájlban szövegesen is hozzáadhatod őket a TabPane-hez.

Futtasd a programot, és próbáld ki a fülek közötti váltást! (Ctrl+Tab-bal is lehet.) Mi történik, ha olyan keskenyre állítod az ablakot, hogy nem fér ki minden fül?

## FlowPane elrendezés

Helyezz el az első fülön egy FlowPane tárolót! Ez egymás mellé teszi az elemeket. Ha nem férnek ki, több sorba tördeli őket.

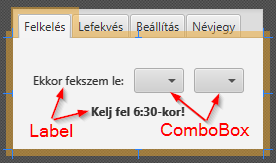


Állítsd a tároló Alignment tulajdonságát CENTER-re! A Hgap és a Vgap értéke 10 legyen!

## ComboBox

A ComboBox elem segítségével előre megadott listából választhatunk.

Helyezd el a következő elemeket a Felkelés lapon:



Az alsó címke betűi legyenek félkövérek, azonosítója legyen: lblKeljFel

A baloldali ComboBox azonosítója cbxLeOra, a jobb oldalié cbxLePerc legyen!

Állítsd mindkettőnél a Layout/PrefWidth tulajdonságot USE\_COMPUTED\_SIZE-ra!

Mindkét ComboBox-ból történő választásnál fusson le a felkel metódus!

Add meg a controller osztály nevét a bal alsó sarokban (Document, Controller)!

Ezután nyisd meg a View, Show Sample Controller Skeleton paranccsal a controllerbe másolandó kódot, és másold a vágólapra a szükséges deklarációkat!

Mentsd a jelenetet, és válts vissza a NetBeans-re!

Nézd át a generált FXML dokumentumot!

Másold be a kódot a vágólapról az FXMLDocumentController osztályba, és a ComboBox-oknál add meg a String típust a kérdőjel helyett!

Próbáld ki a programot!

Láthatod, hogy a ComboBox-okat fel kell tölteni a választási lehetőségekkel.

A ComboBox-hoz a JavaFX automatikusan hozzárendel egy listát. Ez az ArrayList-hez hasonlóan használható, és a getItems metódussal kaphatunk egy hivatkozást rá.

Például a cbxLeOra ComboBox listájára így hivatkozhatunk:

cbxLeOra.getItems()

A listához az add metódussal egy elemet, az addAll metódussal több elemet is hozzáadhatunk.

Például:

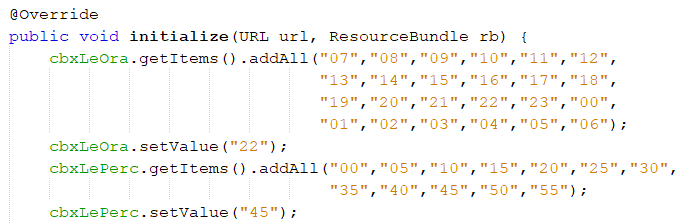
cbxLeOra.getItems().add("Egy elem");

cbxLeOra.getItems().addAll("kettő", "három", "négy");

A kiválasztott elemet a setValue metódussal állíthatjuk be, és a getValue metódussal olvashatjuk ki az értékét. Például:

cbxLeOra.setValue("Egy elem");

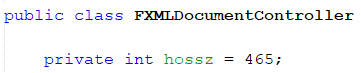
Egészítsd ki az initialize metódus kódját az alábbiak szerint:



Futtasd ismét a programot! Figyeld meg a fenti beállítások hatásait!

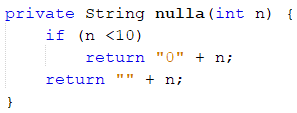
## Felkelés kiszámítása

Vegyél fel egy adattagot a controller osztályba az alvás hosszának tárolásához:



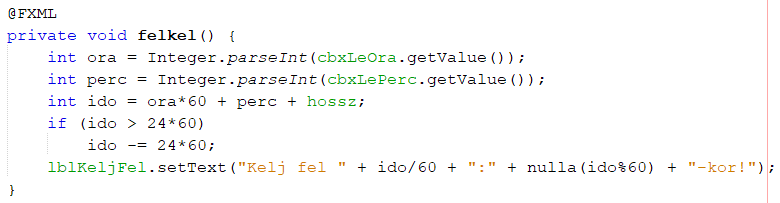
Az alapértelmezett érték 15 perces elalvást feltételez, a Beállításoknál lehet majd módosítani.

Készíts egy nulla nevű metódust, amely a kapott számot stringgé alakítja, és ha egyjegyű, akkor elé tesz egy nullát! Ezt a percek kiírásánál fogjuk használni.



A számításokhoz mindent percekre számolunk át. Például:  
lefekvés időpontja: 22\*60+45 = 1365  
ehhez hozzáadjuk az alvás hosszát: 1365+465 = 1830  
ebből kivonunk egy teljes napot: 1830-24\*60 = 390  
ezt elosztjuk 60-nal: 390/60 = 6 (óra)  
és vesszük a maradékot: 30 (perc)

Ezután készítsd el a felkel metódust így:

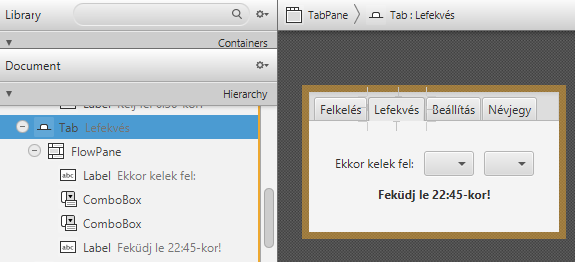


Figyeld meg, hogy a ComboBox értékét a getValue metódussal olvastuk ki!

Próbáld ki a programot!

## Lefekvés kiszámítása

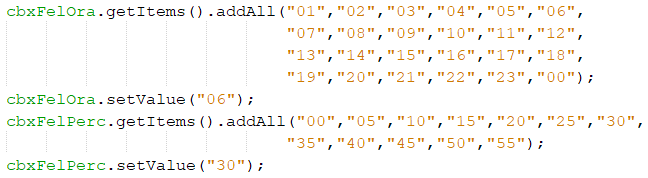
Készítsd el a Scene Builderben a Lefekvés lapot az alábbi minta alapján:



A ComboBox-ok azonosítói: cbxFelOra, cbxFelPerc. Mindkettőnél a lefekszik metódus fut le, miután megváltozott az értékük. Az alsó címke azonosítója: lblFekudjLe.

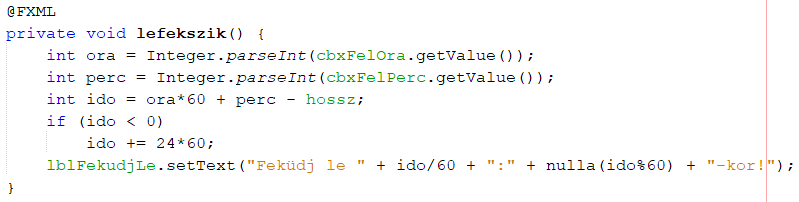
Másold át a controllerbe az új FXML deklarációkat!

Egészítsd ki az initialize metódust a következőkkel:



A számításokhoz itt is mindent percekre számolunk át. Például:  
ébredés időpontja: 6\*60+30 = 390  
ebből levonjuk az alvás hosszát: 390-465 = -75  
ehhez hozzáadunk egy teljes napot: -75+24\*60 = 1365  
ezt elosztjuk 60-nal: 1365/60 = 22 (óra)  
és vesszük a maradékot: 45 (perc)

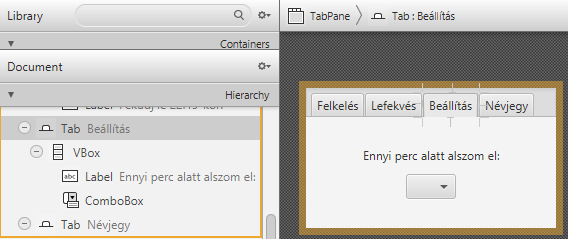
Ezután készítsd el a lefekszik metódust:



Próbáld ki a programot, és ellenőrizd a számítást!

## Elalvás beállítása

Ezután készítsd el a Scene Builderben a Beállítás lapot az alábbi minta alapján! A címkét és a ComboBox-ot egy VBox tárolóban helyezd el! A VBox középre igazítson, 10 pontos térközzel!



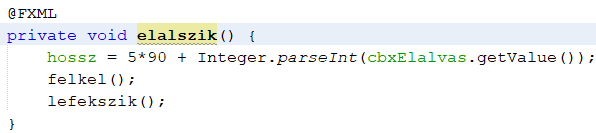
A ComboBox szélessége automatikusan számított, azonosítója cbxElalvas legyen!

A ComboBox-ból történő választás után az elalszik metódus fusson le!

Másold át a szükséges FXML deklarációkat a controller fájlba, majd egészítsd ki az initialize metódust:

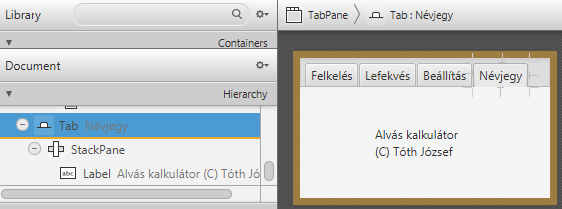


Ezután készítsd el az elalszik metódust, és teszteld a beállítás működését!



## Névjegy elkészítése

Végül készítsd el a Névjegy lapot is! Ezen mindössze egy címke van egy StackPane tárolón elhelyezve:



A címkét úgy tördeltem két sorba, hogy bekapcsoltam a Wrap Text (szöveg tördelése) tulajdonságát, és a preferált szélességét 100-ra állítottam.

A szerző nevéhez saját nevedet írd!

Próbáld ki!

## Feladat

Állítsd az ablak címét Alvás kalkulátorra! Próbáld ki!

## Kérdés

Hogyan adhatjuk meg egy ComboBox elemeit? Milyen típusúak ezek az elemek?

## Feladat

Írd be az alábbi táblázat bal oldali oszlopába a megfelelő tárolók neveit:

|  |  |
| --- | --- |
| HBox | Vízszintesen egymás mellé rendezi az elemeket. |
|  | Függőlegesen egymás alá rendezi az elemeket. |
|  | Egymás mellé rendezi az elemeket, és ha nem férnek ki, új sort kezd. |
|  | Öt területre osztja az ablakot (top, bottom, left, right, center), és mindegyikbe helyezhetünk elemet. |
|  | Egymás tetejére helyezi az elemeket. |
|  | Egy táblázat celláiba helyezi az elemeket. |
|  | Húzással helyezhetők el benne elemek, amelyek a tároló széleihez is rögzíthetők. |
|  | Az elemeket füleken helyezi el. |