



December 2

# Szoftverfejlesztés gyakorlat - 14. óra

# 2019

GUI - Graphical User Interface – JFrame JButton JText JLabel - Eseménykezelés

Feladatlap

## Tartalom

1. Feladat - Lottó .....	2
2. Feladat – Számkitalálós játék .....	3
3. Feladat – Stopper óra .....	3
4. Feladat – Növekvő form .....	3
5. Feladat - Színes játék .....	3
6. Feladat - Adat bekérés .....	3
7. Feladat – Adat bekérés .....	3

## 1. Feladat - Lottó

Készíts Lottó grafikus alkalmazást, amelyben meg tudod adni a szelvény típusát, és egyesével ki tudod húzni a nyerőszámokat, majd emelkedő számsorrendben megjeleníted azt. A számok húzását animáció segítségével valósítsd meg. Az animálást a fejlesztés legutolsó fázisában próbáld leprogramozni, amikor már minden egyéb funkció működik!

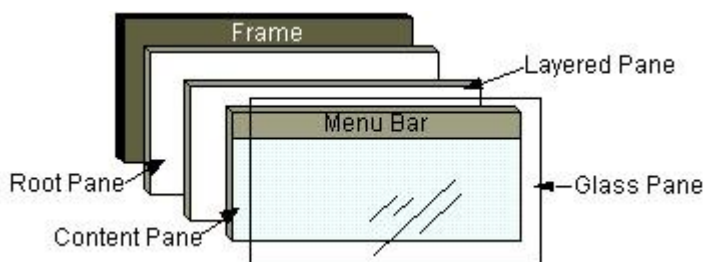
Készíts egy `FrmLotto` osztályt, amelyet a `JFrame` osztályból származtass. Hozz létre egy konstruktort, amelyben csak egy `Init()` metódust fogsz majd meghívni. Az `Init` metódusnak nincs visszatérési értéke, a feladata hogy inicializálja az ablak elkészítéséhez szükséges vezérlőket, beleértve magát a `JFrame` ablakot is.

Az `Init` metódusban állítsd be a következőket.

- Az ablak címe, melyet a bal felső sarokban olvashatsz legyen `Lottó`.
- Az ablak mérete legyen 300 px széles és 400px magas.
- Az ablakot ne lehessen átméretezni.
- Az ablak megjelenésének kezdőpozícióját állítsd be középre.
- Amikor bezárják az ablakot, akkor a program futása is álljon le.
- Az `Init` legvégén állítsd be az ablak láthatóságát.

Futtasd a programot, teszteld, mely elemek kellenek minimálisan az ablak megjelenítéséhez és egyben helyes működéséhez, illetve melyek hagyhatóak el.

Az osztályhoz adj hozzá egy `JPanel` típusú adattagot, melyet inicializálj az `Init` metódusban úgy, hogy implicit konvertáld `JPanel` típusúvá az osztályból lekérhető `getContentPane()` metódust. Az adattag neve `pnlMain` legyen és `Layout` (elrendezés) tulajdonságát állítsd be null értékre, hogy az ne az alapértelmezett `BorderLayout` viselkedjen.



A `RootPane` (gyökér panel) számos réteget tartalmaz, amelyek közül az egyik a `ContentPane` (tartalom panel).

Amikor hozzáadsz valamit egy `Frame`-hez, akkor automatikusan hozzáadódik `ContentPane` ablaktáblához.

Az osztályhoz adj hozzá két `JLabel` típusú adattagot, amelyek az aktuális lottó számot fogja megjeleníteni illetve, a kihúzott számokat fogja tárolni. Az adattagok nevei `lblNumber` és `lblNumbers` legyen. Az `Init` metódusban állítsd be a `lblNumber` tulajdonságait.

- Kezdeti szöveg legyen „1”.
- Betűtípus `Courier`, stílus félkövér és a méret 78 legyen.
- A felirat mérete 90 széles, 70 magas, pozíciója a panel x koordinátájának 20 értéke, illetve y koordinátájának szintén 20 értéke.
- Végül add hozzá `pnlMain` adattaghoz az elkészített feliratot.

Az osztályhoz adj hozzá két  `JButton` típusú adattagot, amelyek a lottó sorsolás megkezdéséért, az egyes számok kihúzásáért illetve a számok rendezéséért fognak felelni. Az adattagok nevei `btnnStart` és `btnnSort` legyen.

Készítsd el a további plusz funkciókat a programhoz

- Tedd lehetővé a generált számok elmentését és visszatöltését!
- A program mentse el a sorsolás pontos idejét is!
- Tedd lehetővé tippek megadását és a tippek sorsolás utáni kiértékelését, azaz a program állapítsa meg, hogy hányas találatunk volt!

Bővítsd a programot a skandináv lottóval, és a kenő játékkal! Nézz utána az interneten a pontos játékszabályoknak!

Érdekesség, hogy 1957 márciusától (10. hetétől) a szerencsejáték honlapján megtalálható az eddigi összes nyerőszám: <https://bet.szerencsejatek.hu/cmsfiles/otos.html>

## **2. Feladat – Számkitalálós játék**

Készíts számkitalálós játékot!

- A program gondoljon egy számot 1-1000 között! (Random)
- A felhasználó tippelhesen a gondolt számra (JTextBox, JButton), és a program írja ki, hogy a tipp helyes-e, illetve, hogy kisebb, vagy nagyobb a gondolt számnál! (JOptionPane)
- Helyes tipp esetén a játék érjen véget gratulációval (JOptionPane), és lehessen újra kezdeni a játékot!
- A program folyamatosan jelezze ki a játékidőt, illetve a tippek számát!
- Minden játék végén a program rögzítse és jelenítse meg az aktuális eredményt (játékidő, tippek száma)!

## **3. Feladat – Stopper óra**

Készíts egy stopperóra alkalmazást.

## **4. Feladat – Növekvő form**

Készítsd el a mellékelt Windows Form programot, amelyben egy form méretét tudod növelni vagy csökkenteni bizonyos határok mentén.

## **5. Feladat - Színes játék**

Készítsd el a mellékelt Windows Form programot, amelyben egy játékot kell megvalósítanod, a cél egyszínűvé varázsolni az alkalmazás minden vezérlő gombját és a hattárt. Teszteld a programot, jöjj rá a logikára, mikor melyik kattintáskor mi történik!

## **6. Feladat - Adat bekérés**

Készíts Windows Form programot, amely

- lehetőséget ad a felhasználónak, hogy megadhassa egy háromszög oldalainak a hosszát (TextBox)
- ellenőrzi a megadott értékeket, és a hiba jellegének megfelelő üzenetet ad, pl. „Nem pozitív valós számot adott meg!” (Try-Catch, MessageBox) vagy pl. „A megadott adatokból nem szerkeszthető háromszög”
- Kiszámolja, és megjeleníti a háromszög kerületét, és területét! (Label) A terület meghatározásához használd a Héron-képletet (nézz utána az interneten)

## **7. Feladat – Adat bekérés**

Készíts Windows Form programot, amely bekér egy szöveget, és elvégzi a következőket!

- Kiírja a szöveg karaktereinek a számát!
- Kiírja a szöveg szavainak a számát!
- Kiírja a szövegben a magánhangzók számát.
- Kiírja a szövegben a mássalhangzók számát.
- Kiírja a szövegben az írásjelek számát.
- Kiírja a szöveg megfordítottját!
- Kiírja a szöveget, csupa kisbetűvel, nagybetűvel és CamelCase formában is.
- Kiírja, hogy a szöveg palindroma-e (ugyanaz, mint megfordítva), ehhez távolítsd el a felesleges írásjeleket, szóközöket és egységeses kisbetűs változatot alkalmazz az ellenőrzés során. pl.: Indul a görög aludni!