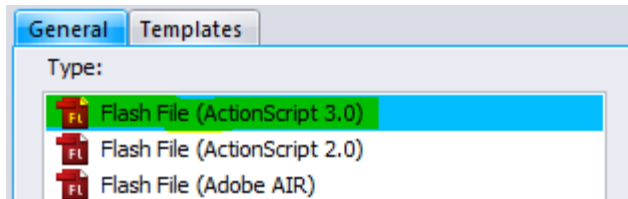


Preloader (előtöltő) készítése Flash CS4 Actionscript 3 –al lépésről-lépésre.

A lecke alatt elkészítünk egy előtöltőt, ami lefoglalja a nézőt, míg a flash lejátszója megjeleníti a betöltendő tartalmat. Jelen példában egy autó képét fogjuk megjeleníteni egy Alpha átmenetes hatással. A leckében megismerkedhetünk a Flash CS4 alapjaival és az Actionscript 3 programozással. A leckét próbáltam minél részletesebben leírni lépésenként képekkel illusztrálva.

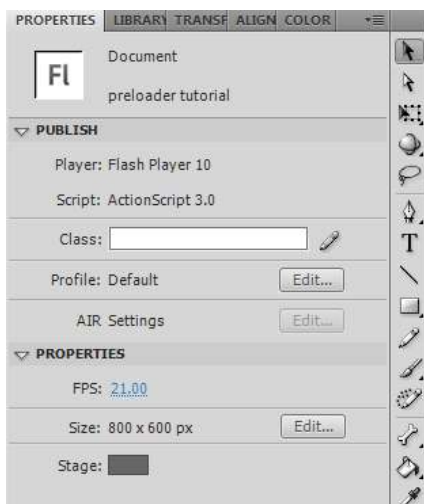
1. Nyissunk meg egy új dokumentumot a flash CS4-el és válaszuk típusként a **Flash File (ActionScript 3.0)**-t. (File- New-New Documentum-General-Type- Flash File (ActionScript 3.0). (1a, kép).

a,

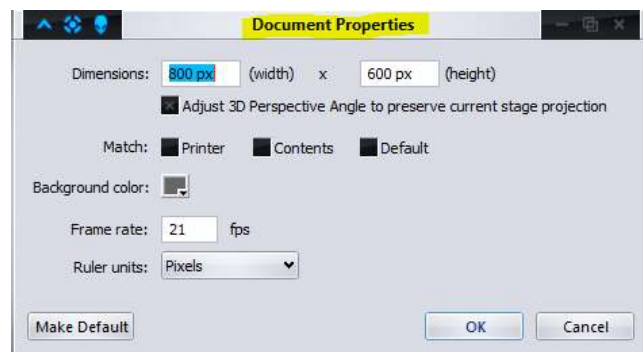


2. A **Properties/Tulajdonságok** ablakban állítsuk be példánk szerint az **Edit/Szerkesztés** lehetőséggel a színpad méretét **800 x 600 px**-re. Az **FPS/Másodpercenkénti** kockák száma legyen 21. A Stage/Színpad színe legyen szürke: **#666666**. Ok.(2a, 2b kép)

a,



b,



Mentsük munkánkat: **File-Save**, névnek a **preloader tutorial**-t adtam.

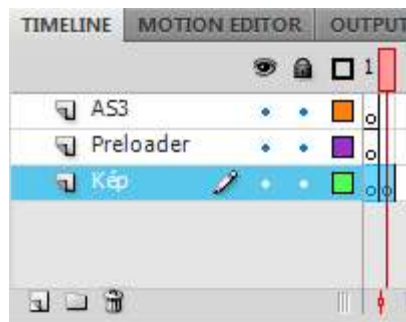
3. A **Timeline/Időszalag**-on készítsünk újabb két réteget a **New Layer** gomb segítségével. A rétegek neveibe duplakattintással belépve nevezzük át fentről lefelé haladva **AS3, Preloader**, és **Kép**. Az AS3-ba csak kódokat fogunk használni. A Preloader rétegre fogjuk majd elkészíteni a betöltőnket. A Kép rétegre elkészítünk egy kisebb animációt, amit a betöltés után szeretnénk megjeleníteni. (3a, kép)

a,



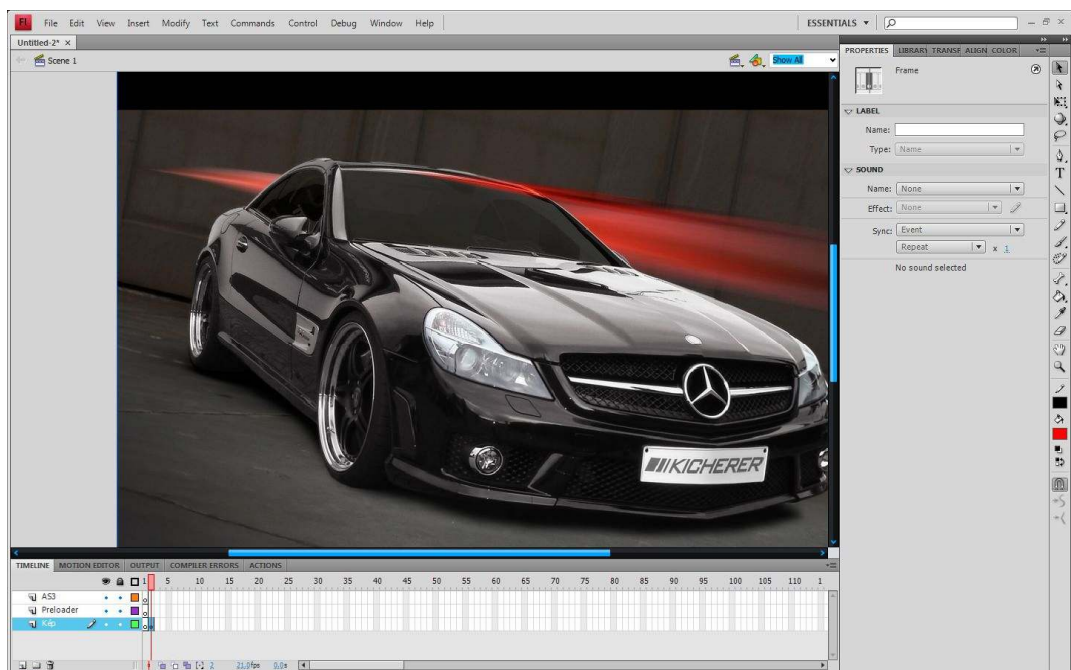
4. A **Kép** rétegen kattintsunk az első kulcskockába, majd az egér **bal** gombját lenyomva **+ shift** gomb folyamatos nyomásával húzzuk el az egér segítségével a kijelölt kulcskockát egy kockával arrébb. Így a kijelölt rétegen ez első kulcskocka után beszúrtunk egy új kulcskockát. Másképp ugyan ezt: jelöljük ki a kulcskockát majd jobb egérgombbal megjelenő ablakból választuk ki az „**Insert Kayframe** „lehetőséget. (4,a kép)

a,



5. Ebbe a beszúrt kulcskockába importáljunk be egy képet. Válaszunk a **File- Import- Import to stage** lehetőséget. Keresünk egy képet merevlemezünkről, kiválasztás után választuk a Megnyitást. Képünk megjelenik a Stage/Színpadon. (5a, kép)

a,

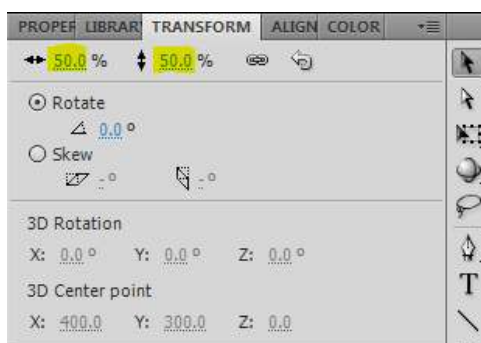


6. A képigazításhoz nyissuk meg a **Window** menüt és választuk ki a **Transform** ablakot. Legyen bekapcsolva a **Constrain** (összekötés, összekapcsolás) kis lánc ikon (6a, kép). A képünk arányos méretezéséhez használjuk a **Scale Width** vagy a **Scale Height** százalékos arány beállítás lehetőségét. Az egérrel menjünk rá valamelyik %-os értékre, majd az egér jobb gombját lenyomva tartva vízszintesen elmozdítva állíthatunk a kép arányain. Én itt 50-50%-ot használtam az eredetihez képest (6b, kép).

a,



b,



Nyissuk meg az **Align** ablakot, kapcsoljuk be a **To stage**/Színpadhoz igazítás, választuk ki az **Align Vertical Center**-t (6c, kép). Így a képet a színpad közepéhez igazítottuk vertikálisan. Igazítsuk a képet horizontálisan is közésre a **Distribute Horizontal Centert**-t választva. (6d, kép)

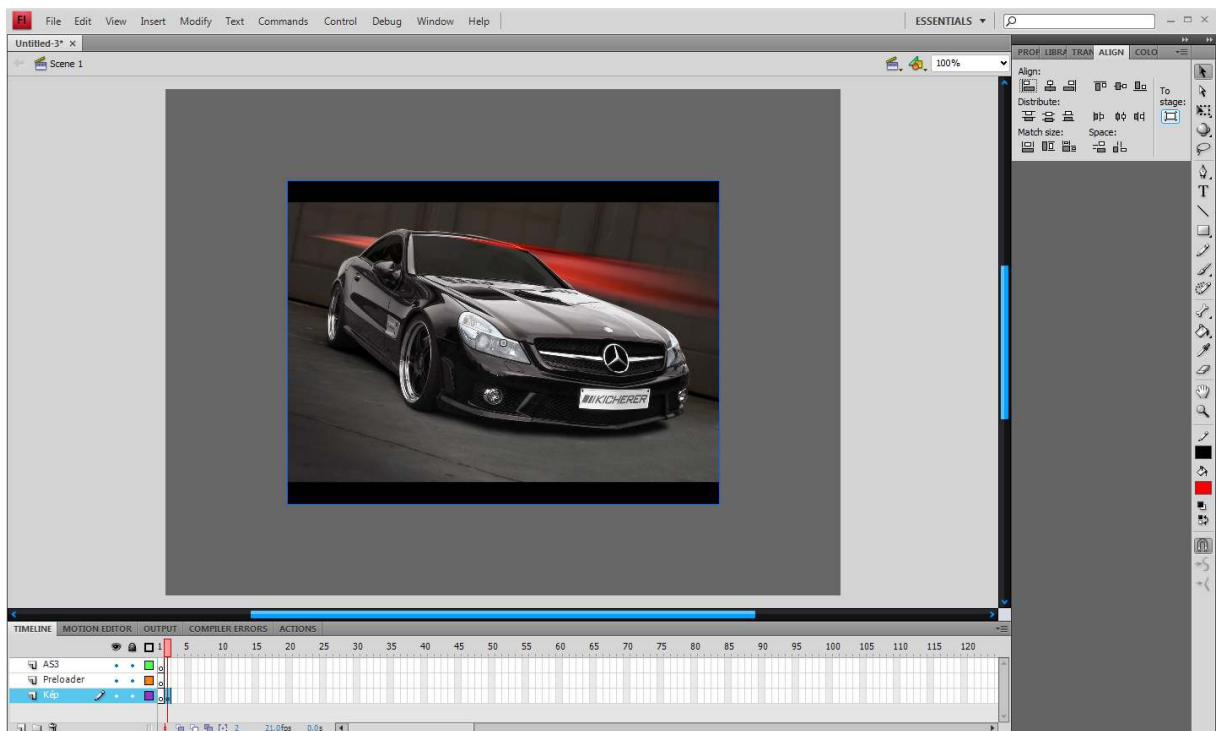
c,



d,

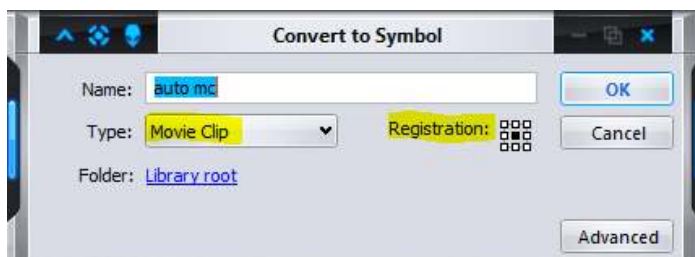


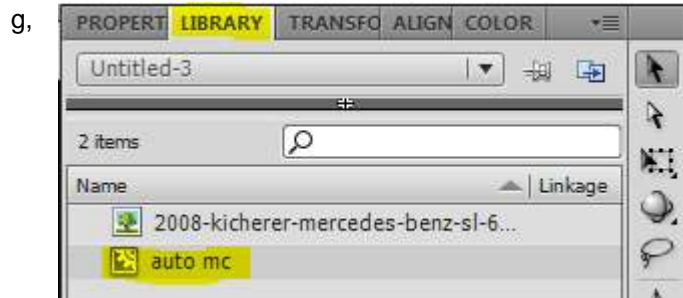
e,



Ok. A képünk beállításával megvagyunk.(6e, kép). A képünket alakítsuk át **Movi Clip** szimbólummá.(6f, kép) Az egérrel kattintsunk képünkre ezzel kijelölve, és nyissuk meg a szerkesztő léceken a **Modify** menüt. Válaszuk ki a **Convert to Symbol**-t (F8). A megjelenő ablakban névnek adjuk az **auto mc**-t. Regisztrációs pontként jelöljük be a középpontot. Majd nyomjunk Ok-t. A **Library**/Könyvtár panelen ellenőrizhetjük a mozi klipünk meglétét.(6,g kép)

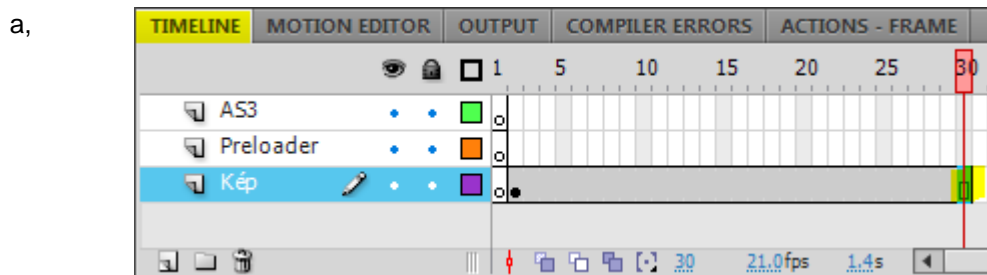
f,



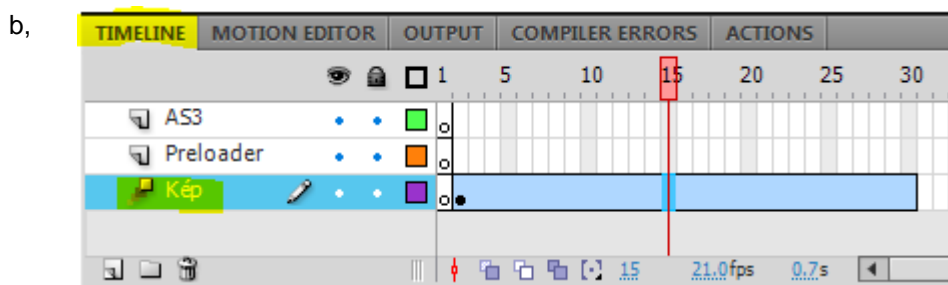


Készítsünk hozzá egy kis animációt.

7. A Timeline-on a Kép rétegen kattintsunk a 30. filmkockára és az **F5** gomb lenyomásával szúrjunk be egy üres filmkockát.(7a, kép)

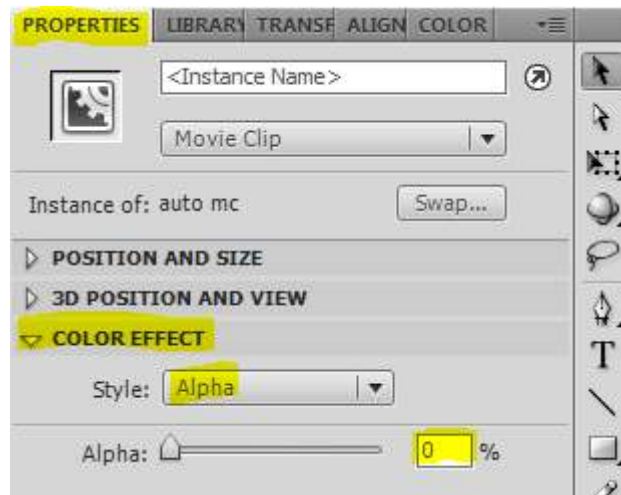


Kattintsunk a Kép réteg idősvábjában a 2. és 30. képkocka közé eső kockák valamelyikére a jobb egér gombbal. A megjelenő ablakban választuk ki a **Creat Moiton Twen-t**. A 2. és 30. filmkocka közötti kockák kékre színeződtek. Ezzel jelzi a flash, hogy azon a rétegen valamilyen mozgás animáció van, ill. lesz. (7b, kép)



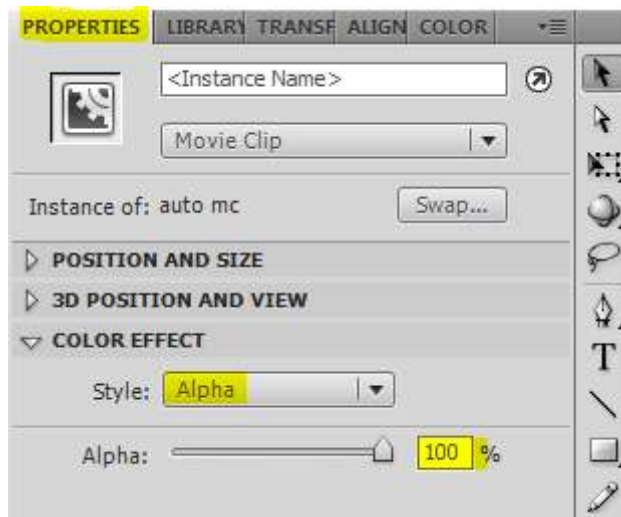
8. Kattintsunk a Kép réteg idősvábjában a 2-es filmkockára, majd az egérrel kattintsunk a színpadon lévő képre. A Properties panelen választuk ki a **Color Effect** menüben található **Style** beállítást, ezen belül választuk az **Alpha** értéket és az értéket állítsuk 0%-ra.(8a,kép)

a,



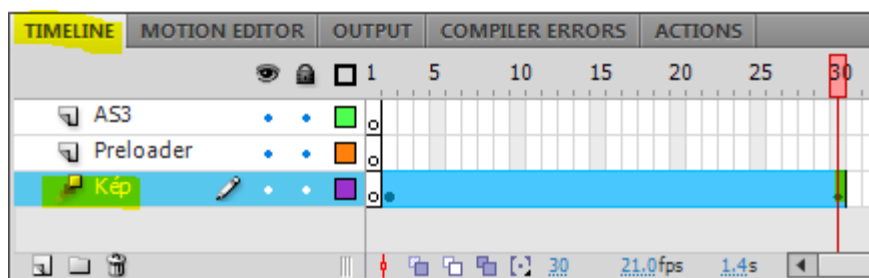
Kattintsunk a Kép idősvábjában a 30. filmkockára, majd kattintsunk az egérrel a képünkre. A Properties panelen az **Alpha** értéket állítsuk 100%-ra.(8,b kép)

b,



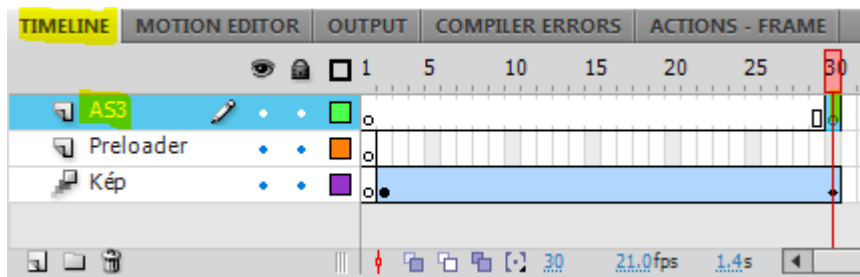
9. Ha mindent jól csináltunk, akkor a 30. képkockában találunk egy apró fekete pontot, amit a flash generált. A Kép réteg melletti kis ikon is megváltozott. Ezzel jelzi a flash, hogy valamilyen animációt tartalmaz az adott réteg. (9a,kép)

a,



10. Ellenőrizzük le, hogy mit is csináltunk eddig. Nyomjuk le a **Ctrl+Entert**, vagy választuk a **Control** menüt és a **Test Movie**-t. A mozink egy alfa értékkel ellátott clipet játszik le. Tehát az autó „átlátszódása” 0 értékről 100-ra fokozatosan erősödik. A mozink pulzálva újra lejátszódik. Ha azt szeretnénk, hogy a teljes lejátszódás (megjelenés) után megálljon, az AS3 rétegre beépítünk egy stop eseményt. Ehhez jelöljük ki az **AS3** réteget, majd kattintsunk a réteg **30.** filmkockájára ahol befejeződik a mozink, és nyomjunk egy **F6**-ot. Ezzel beszúrtunk egy kulcskockát.(10a, kép)

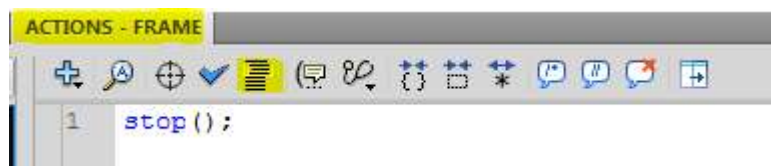
a,



11. Kattintsunk erre a kulcskockára és nyissuk meg az **Actions panelt**. Az első sorba írjuk be a következő kódot: **stop();** (11a, kép)

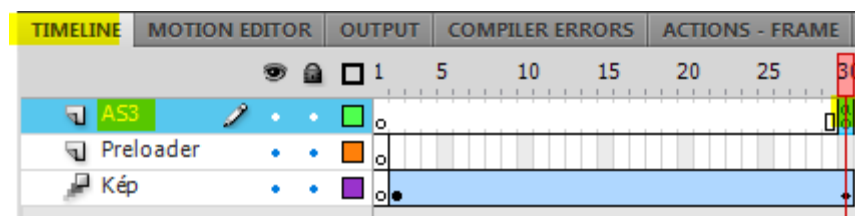
Meg fogunk lepődni, hogy miért nem tudjuk be írni a pontos vessző karaktert. Azért mert a billentyűzetünk alapleosztása magyar, ha átállítjuk angolra, akkor működni fog. De létezik egy gyors megoldás erre a problémára. Az stop szó és a zárójelek begépelése után az action panelen van egy **Auto Format** gomb. Használjuk ezt. A flash ekkor kijavítja a kódunkat. Ebben az esetben lezárta a parancsunkat a pontos vessző karakterrel.

a,



A Timeline-nak pedig így kell kinéznie. Az AS3 réteg 30. kulcskockájában szerepel egy (a) karakter, ami azt jelzi, hogy ott szerepel egy utasítás vagy parancssor (akció), amit a flash lejátszónak végre kell hajtania. Jelen esetben megállítja a lejátszást. (11b, kép)

b,

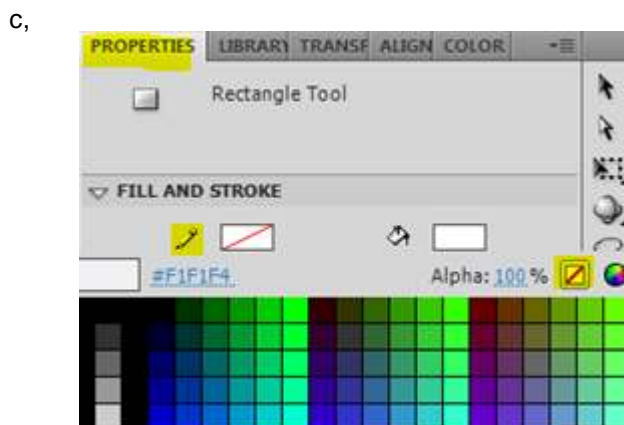
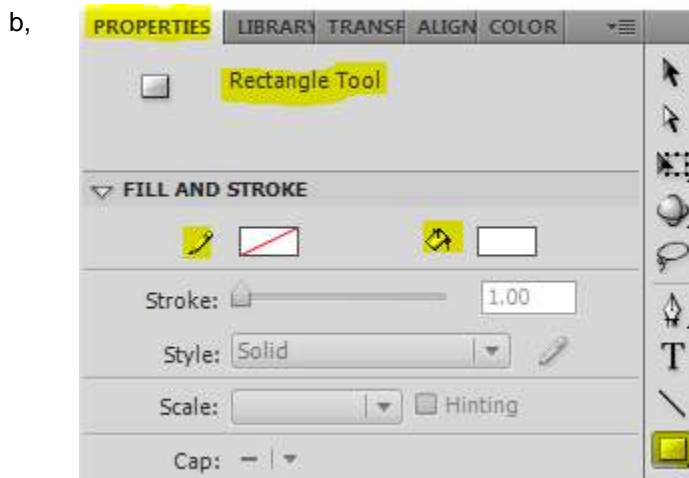
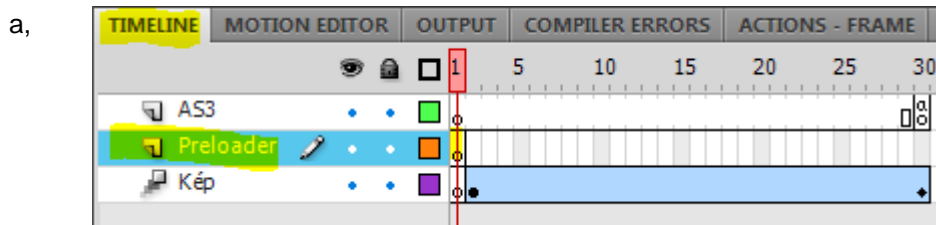


Teszteljük újra a mozinkat: **Ctrl+Enter**. Most már csak egyszer játszódik le a mozink.

Ezt az animációt fogjuk betölteni. Most pedig készítsük el hozzá az előtöltőt.

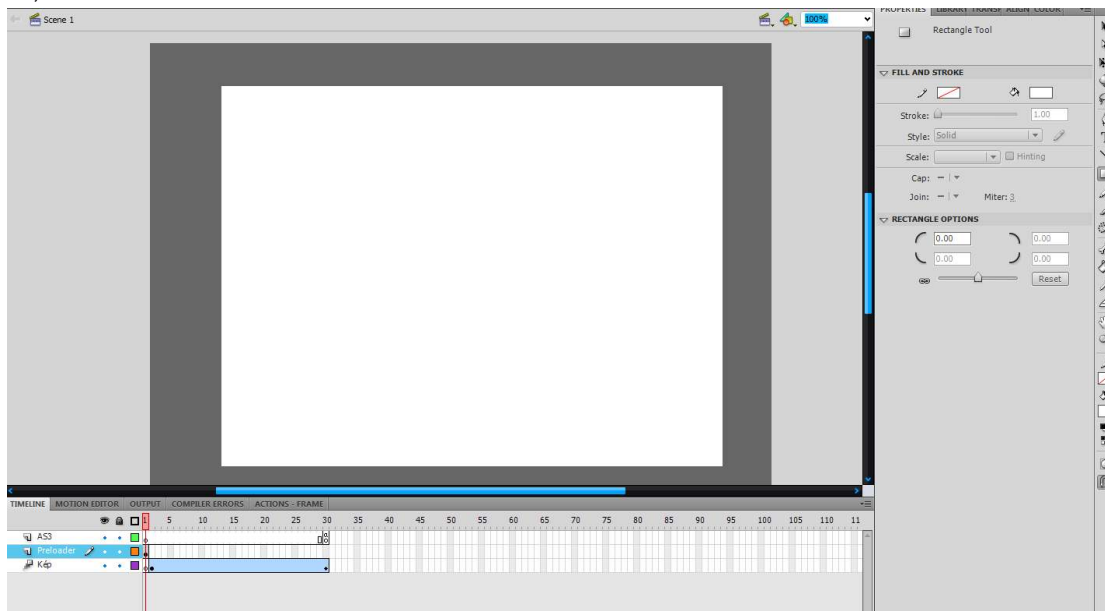
Az előtöltő/preloader készítése.

12. Kattintsunk a Timeline panelen a **Preloader** réteg első kulcskockájába (12a, kép). A Properties panelen választuk ki a rajzeszközök közül a **Rectangle Tool** eszközt (12b, kép). A **Fill And Stroke** beállításnál kattintsunk a körvonalszín választóra és tiltsuk le a körvonalszínt (12c, kép). Kitöltő szín megválasztásához használjuk az előző módszert a festékes vödör melletti színválasztóra kattintva. Én a fehéret választottam.



13. A **Preloader** réteg első kulcskocáját kijelölve a színpadra rajzoljunk egy tetszőleges méretű nagy fehér téglalapot. (13a, kép)

a,



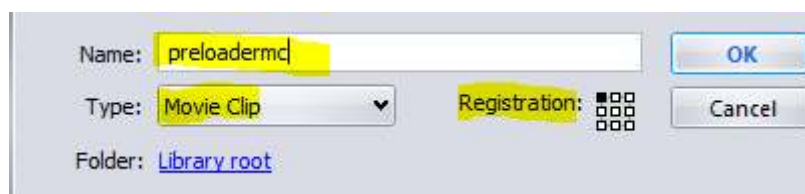
14. Kattintsunk a **Selection Tool** eszközre és jelöljük ki vele a téglalapunkat. (14a,kép)

a,



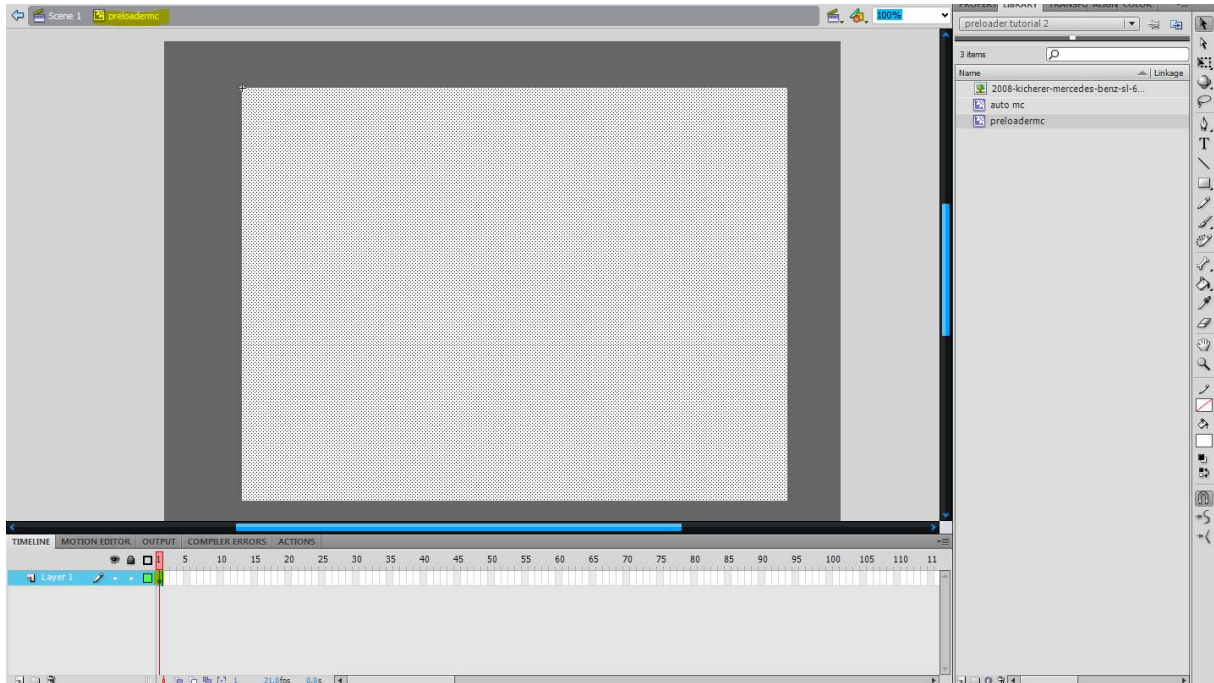
Az **F8**-al készítsünk belőle **Movie Clip** szimbólumot. Neve legyen **preloadermc**. A **regisztrációs pontja a bal felső pont legyen**. Ok. (14b, kép)

b,



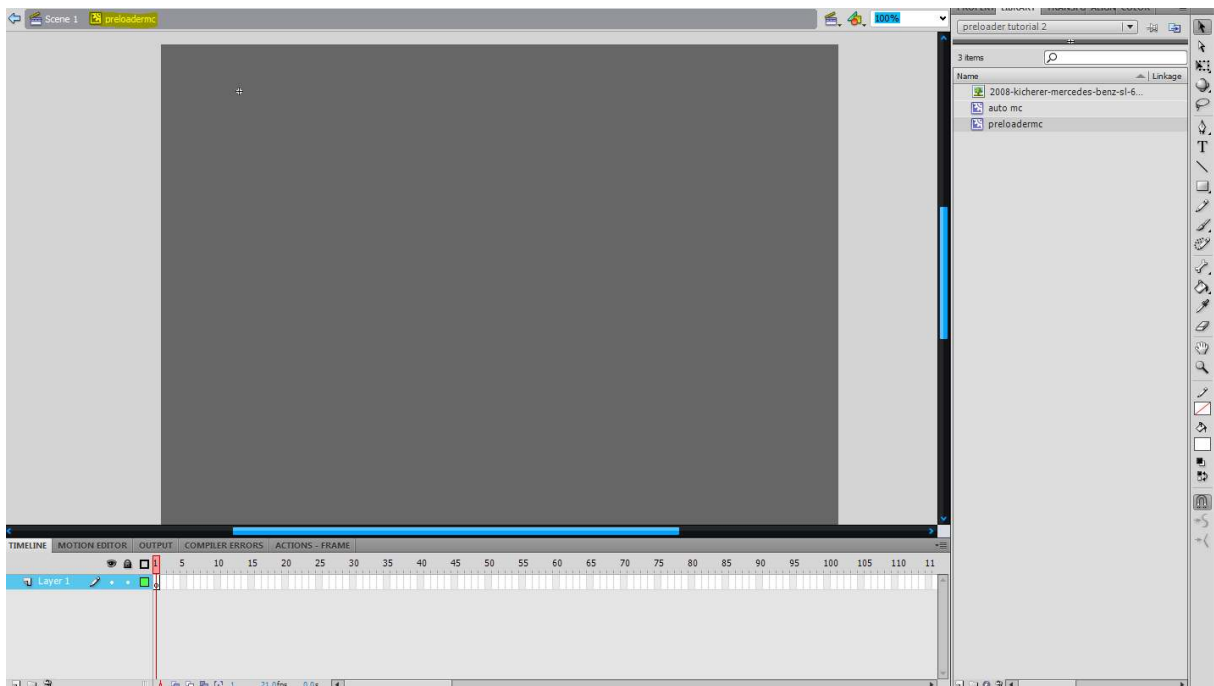
15. Kattintsunk duplán a fehér téglalapon, így a szimbólum belsejébe jutunk így azt kijelöltük szerkesztésre. A Scene 1 felirat mellett megjelenik a **preloadermc** is. A Timeline-on egy réteg szerepel. Az első kulcskockában pedig a preloadermc szerepel **szerkesztő módban**. (15a, kép)

a,

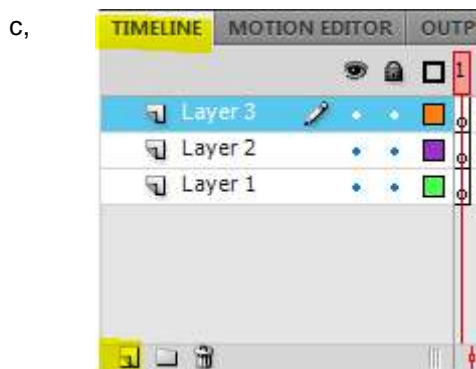
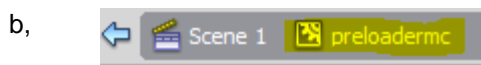


16. Kattintsunk a téglalagra és a **Delete**-vel töröljük a színpadról. (16a, kép)

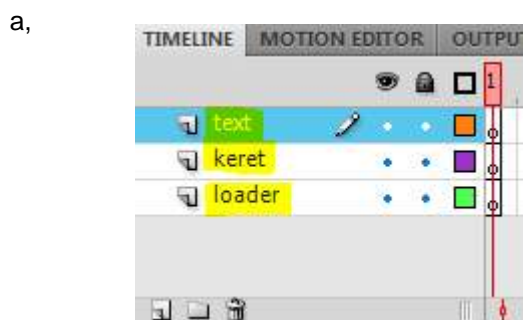
a,



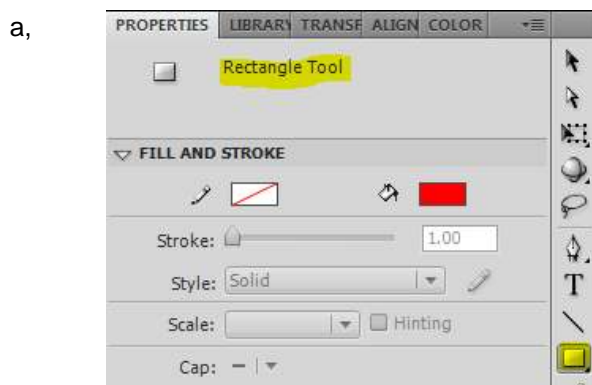
Még a **preloadermc**-ben maradva (16b, kép) a Timeline-on szűrjük be két új réteget a **New Layer** gombbal. (16c, kép)



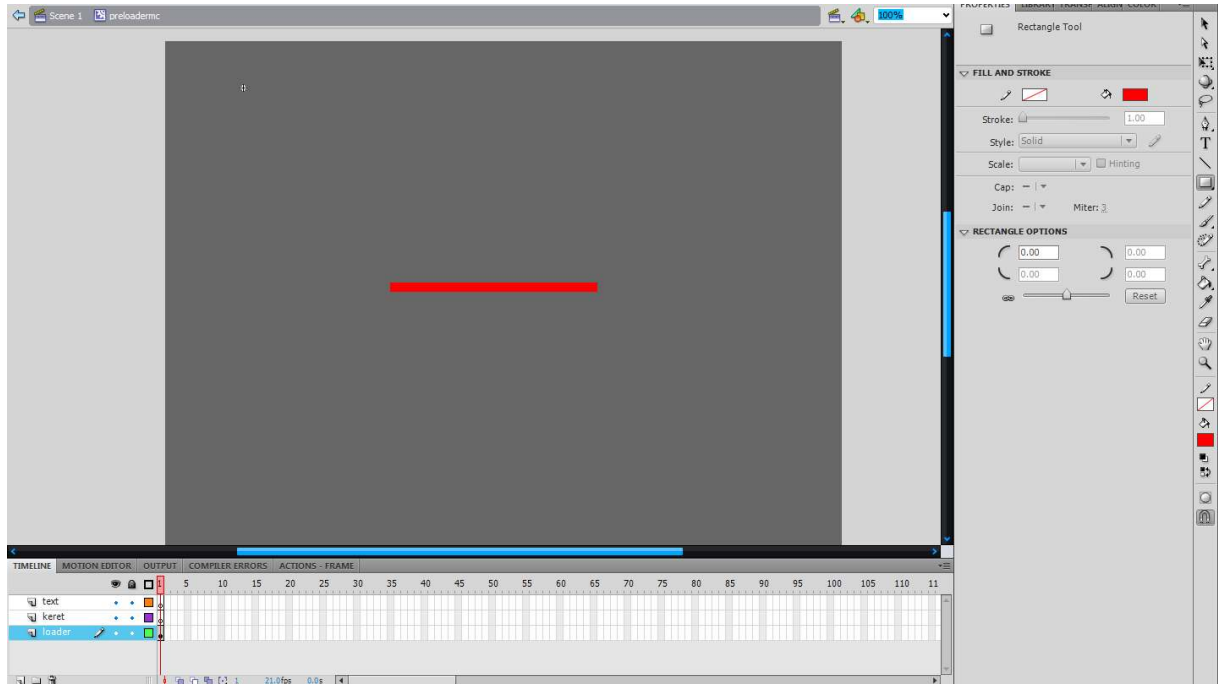
17. A rétegek neveit dupla kattintással írjuk át. Az első legyen **text**, a második **keret**, a harmadik **loader**. (17a, kép)



18. A **loader** réteg első filmkockájába a már ismert módon rajzoljunk egy piros színű körvonal nélküli téglalapot a **Rectangle Tool** eszközzel. (18a és 18b kép)

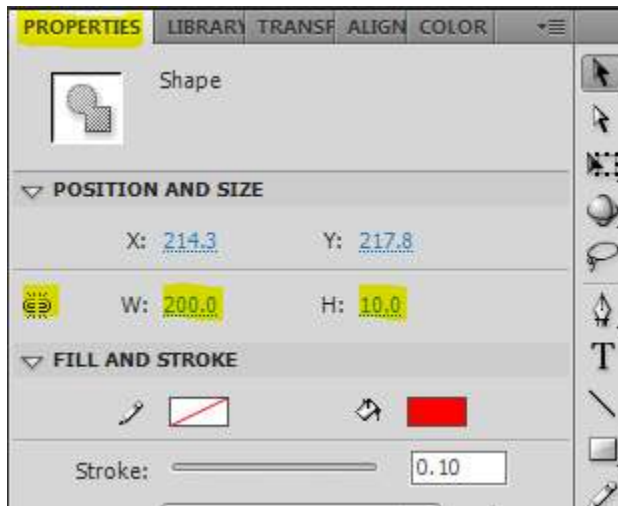


b,



A megrajzolt és kijelölt téglalapon a **Properties** panelen végezzük el az alábbi beállításokat. **Figyeljünk arra, hogy a kis lánc ikon (összekapcsolás) most ne legyen bekapcsolva.** A **Width** (hosszúság) legyen 200, a **Height** (szélesség) legyen 10. A téglalapot az egérrel igazítsuk körülbelül a színpad közepére. (18c,kép)

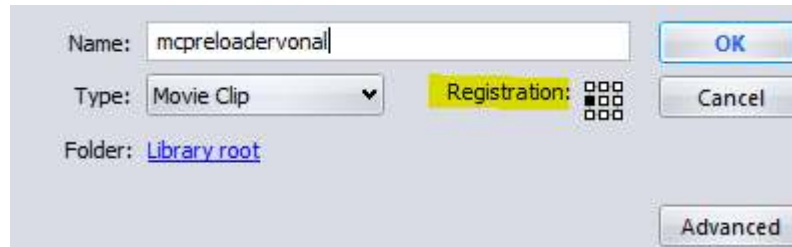
c,



19. A téglalapról készítsünk **Movie Clip** szimbólumot az **F8** gombbal. Neve legyen **mcpreloadervonal**.

A **regisztrációs pont** legyen a **bal középső pont**. Ok. Később kitekerek rá, hogy miért ezt a pontot adtuk meg. (19a, kép)

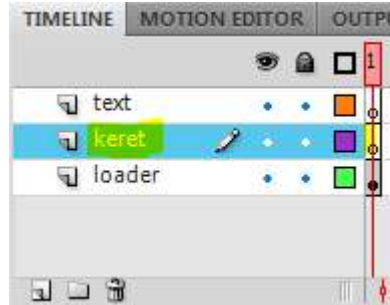
a,



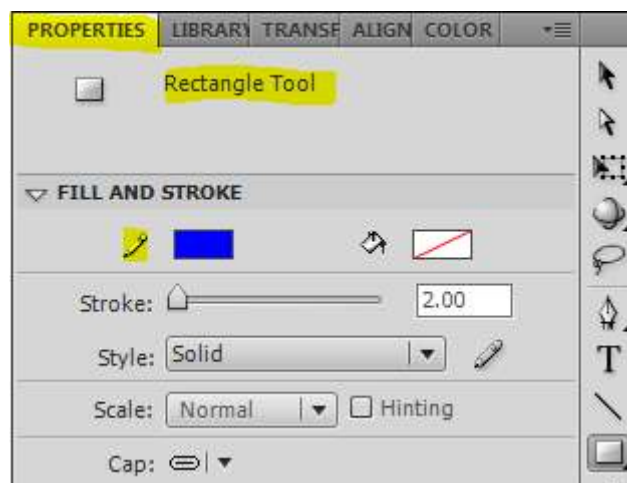
Ezzel megalkottuk a betöltő csíkot.

20. Most rajzoljunk neki egy keretet. Kattintsunk a **Keret** réteg első filmkockájába (20a, kép). A rajzeszközök közül válaszuk ki a **Rectangle Tool** eszközt. A **Stroke Color** (vonal szín)-nek válaszuk sötétkék színt. A **Fill Color** (kitöltő szín) legyen **letiltva**. A **Stroke-t** (vonalvastagság) állítsuk 2-re (20b, kép). Ha megvagyunk, akkor a piros téglalap bal felső sarkától ki indulva húzzuk az egeret a téglalap jobb alsó sarkához (20c, kép), ezáltal rajzolva köré egy kék vonalú téglalapot. Ha ügyesek voltunk úgy néz ki a téglalap mintha adtunk volna neki egy kék körvonalat (20d, kép). A flash amúgy segít a pontos rajzolásban. Jelen esetben a körvonalat szorosan hozzá illeszti a téglalaphoz.

a,



b,



c,



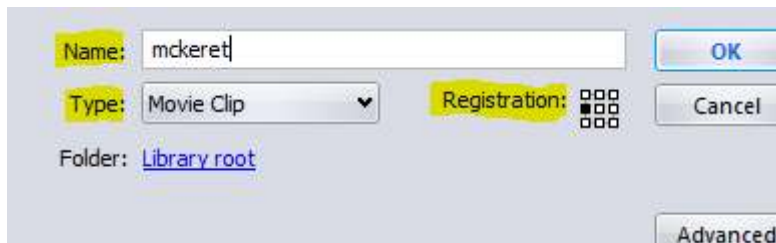
d,



Ez a kék vonalú keret szerepel a Keret réteg első kulcskockájában.

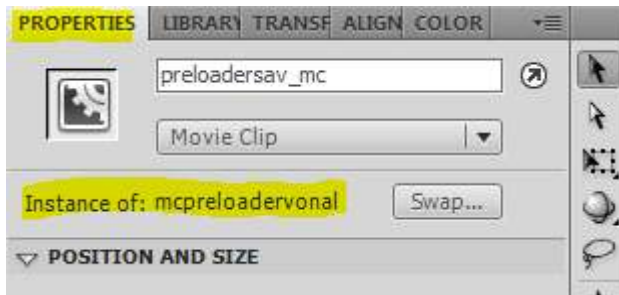
A keretből készítsünk **Movie Clip**-et. (20e, kép) Vigyázunk, hogy csak a keret legyen kijelölve. Kattintsunk a Keret réteg első kulcskockájába, nyomjunk **F8**-at. A Movie Clip szimbólum neve legyen **mkeret**. A **regisztrációs pont** legyen szintén a **bal szélső középpont**. Ok.

e,



21. Térjünk vissza a piros téglalapunkhoz (betöltő csík). Kattintsunk a **Loader** rétegen található piros téglalagra a színpadon. A **Properties** panelen győződjünk meg, hogy az **Instance of**-nál az **mcpreloadervonal** mint Movie Clip szerepel. **Instance Name**-nek (példány név) adjuk meg a következő nevet: **preloadersav_mc** (21a, kép). Az ékezetes betűket itt lehetőleg kerüljük el. Az alulra rögzített kötőjelet a Shift+(- kötőjel) billentyű kombinációval oldjuk meg. Erre a kis apróságra figyeljünk (21b, kép).

a,

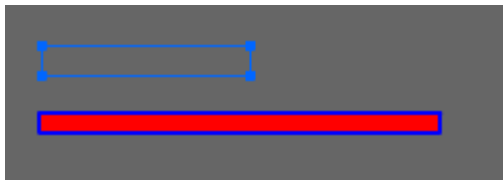


b,

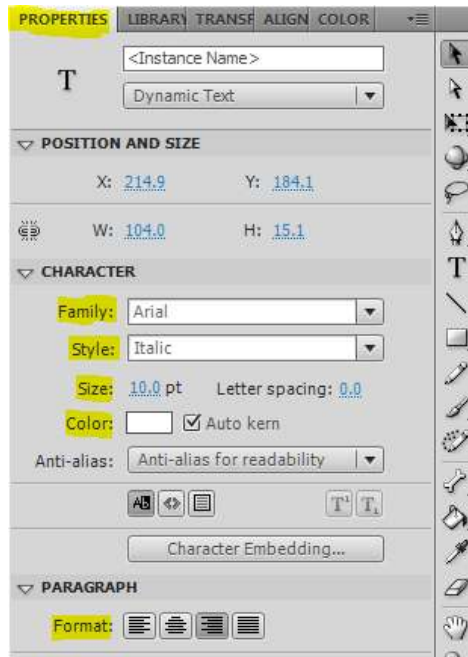


Készítsük el a feliratokat. Kattintsunk a **Text** réteg első kulcskockájába. A rajzeszközöknél válaszunk ki a **Text Tool**-t és kattintsunk vele a színpadon lévő piros betöltőnk fölé (21c, kép). A **Properties** panelen válaszunk ki a **Dynamic Text** beállítást. Formázzuk meg a betűtípust. Állítsuk be a következőkre: Arial, Italic, 10-es méret, fehér szín. Állítsuk be a szöveg helyzetét is, az Align right-al (21d, kép). A **Selection Tool** eszközzel vagy a kurzor billentyűkkel igazítsuk a szövegdobozt a piros téglalapunk közepe felé. Nagyjából, úgy ahogy a képen látjuk (21e, kép).

c,



d,

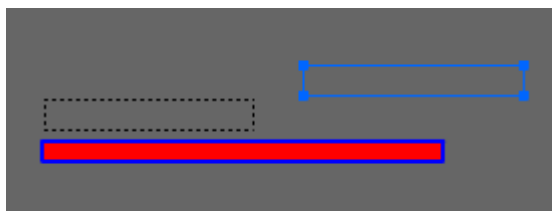


e,

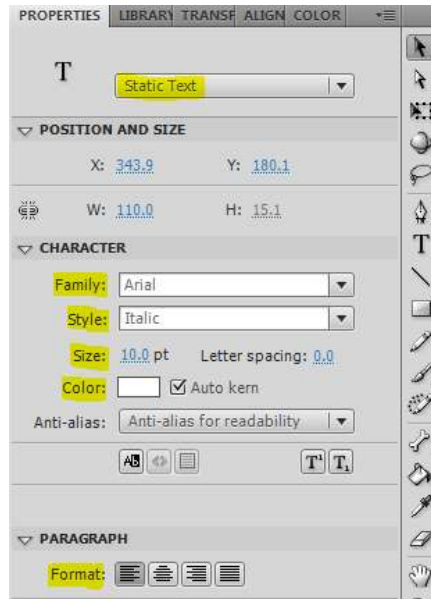


22. Készítsük el a... **loading** feliratot. Még mindig a **Text** réteg legyen kiválasztva. Válaszuk ki újra a **Text Tool** eszközt, majd kattintsunk vele a színpadon lévő piros betöltő jobb felső része fölé. Újra megjelenik a szövegdoboz (22a, kép). A Properties panelen ismét formázzuk a betűtípust. Minden beállítás marad az előző beállításhoz, kivéve az, hogy **nem Dynamic Text** lesz az alapbeállítás, **hanem Static Text**. A szöveg helyzete pedig balra igazított legyen (22b, kép).

a,

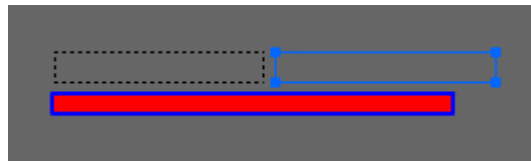


b,



A **Selection Tool** eszközzel vagy a kurzorokkal igazítsuk a szövegdobozt az előző szövegdoboz jobb széle mellé. Szinte érintsék egymást, mint a képen (22c, kép).

c,



Kattintsunk ebbe a szöveg dobozba és írjuk be a... **loading** feliratot. A szöveg beírása után **ne nyomjunk ENTERT**, hanem a **Selection Tool-ra** kattintva lépünk ki belőle (22d, kép).

d,

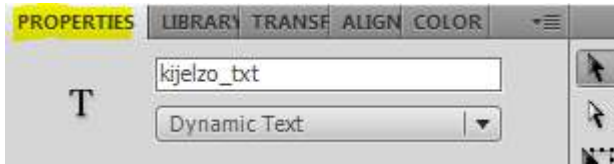


23. Kattintsunk az előző **Dynamic Text** szövegdobozra (23a, kép). Az **Instance Name** (példánynév) legyen **kijelzo_txt**. Figyeljünk az alsó kötőjel beütésére. Valamint kerüljük az ékezetes betűket, ugyanis az Actionscript nem szereti őket (23b, kép). Ha ezzel megvagyunk, térjünk vissza a fő színpadra. Kattintsunk a **Scene 1**-re (23c, kép).

a,



b,



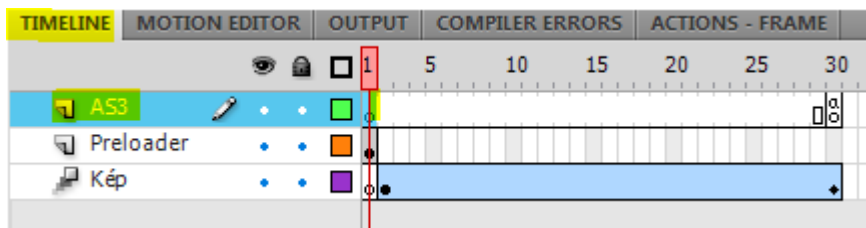
c,



Programozás Actionsript 3-al.

24. Készen vagyunk a grafikai résszel. Most következnek egy kis programozás az Actionsript 3-al. Mielőtt elkezdenénk tisztázzuk, hogy mit is szeretnénk, illetve mire lesz szükségünk. Mindenki ismeri azt a szólást, hogy „a számítógép azt csinálja amire utasítjuk és nem azt amit szeretnénk”. Hiába hadonászunk az egérrel a képernyőnkön addig nem fog történni semmi sem, míg nem kattintunk vele pontosan valahová, hogy bekövetkezzen valamilyen esemény. Valahogy így van az Actionsript-el is és minden programozási nyelvel is. Az Actionsript programozást objektum programozásnak is hívják. Tehát a könyvtárunkban szereplő mozi clipeket vagy a különböző példánynévvel ellátott objektumokat fogjuk programozni. A **Scene 1**-ben (a fő színpad) kattintsunk az **AS3** réteg első kulcskockájába.(24a, kép)

a,



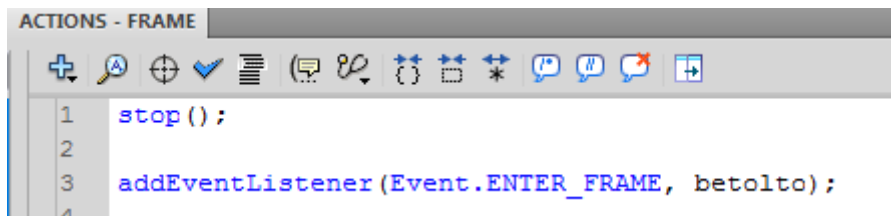
25. Nyissuk meg az **actions panelt**. Ha nem találjuk a **Window** menüből válaszuk ki az **Actions** panelt, vagy nyomjunk **F9**-et.

Az Actions panel első sorába írunk be egy **stop()**-ot. Ez fogja megakadályozni, hogy a flash lejátszó betöltési szakasz nélkül lejátssza az egész mozinkat, illetve megjelenítse a képet. A pontosvesszőt a végén ne felejtsük el. Használjuk erre az **Auto Format** gombot, ha csak nem akarjuk megváltoztatni billentyűzetünk alapnyelv beállítását magyarról-angolra. Helyezzük a kurzort a pontosvessző karakter utáni részre, majd nyomjunk **ENTER**-t. (25a,kép)



Hozzuk létre az **eseményfigyelőt**. Mi is ez? A flash ezt az eseményt fogja lejátszani illetve kiváltani. Az eseményfigyelőben létrehozunk, illetve meghatározunk egy eseményt. Nyomjunk **ENTER**-t és a **3.** sorba írjuk be a következő kódot: **addEventListener(Event.ENTER_FRAME, betoltes);** A **FRAME** szót követő vessző után hagyjunk egy szóközt. Feliratkozunk az eseményfigyelőre, a zárójelek közzé beírjuk az esemény paraméterét és nevét. Az események típusa az **Event** osztályhoz tartozik. Mivel animációról van szó (a betöltő mozink a piros téglalap, ami jelzi a betöltés még hátralévő szakaszát úgy, hogy egyre jobban kitölti a kék színű keretet) így az **ENTER_FRAME** eseményt használjuk. Az **Enter_Frame** lesz az esemény típusa. A kiváltó esemény neve pedig a **betolto**-nevű függvényül esz. Ezt nevezik még **osztálynévnek is**. Az ékezetes betűket kerüljük. A parancssort zárjuk le a pontosvessző karakterrel a már bevált módszerrel. Használjuk az **Auto Format** gombot. (25b, kép)

b,



26. Nyomjuk **kétszer ENTER-t** és az **5.** sorba írjuk be a következő kódot amivel a **betolto** nevű függvényünket fogjuk definiálni, vagyis meghatározzuk a beállításait. Ezt nevezik létrehozó függvénynek is. Minden létrehozó függvény a **function** szóval kezdődik. Közvetlenül utána meg kell adnunk az előzőleg meghatározott függvényünk illetve osztályunk nevét amit **betolto** néven hoztunk létre, ez lesz a konstruktor neve. A név megadása után egy zárójelpár következik, ide fogjuk beírni a konstruktorparamétereket.

```
function betolto(e:Event):void {
```

```
}
```

Adjuk meg az eseményobjektumot aminek az adattípusa az **Event**. A visszatérési típus **void**.

A **{ }** zárójelek közzé írt paramétereket tömbnek nevezzük. Itt sorolhatunk fel elemeket. Változókat fogunk létrehozni benne a „**var**” szóval. (26a, kép)

a,

```
ACTIONS - FRAME
1  stop ();
2
3  addEventListener (Event.ENTER_FRAME, betolto);
4
5  function betolto(e:Event):void {
6
7  }
```

27. A 6. sorban létrehozunk egy „teljesbetolt” nevű változót. **var teljesbetolt:** Ami egy **Number** (szám) osztályú adat lesz. Ezért létrehozunk egy önálló objektumot **loaderInfo**, ami a betöltendő adatt (kép) teljes byte mennyiségét **bytesTotal** tartalmazza. A 7. sorban egy másik változónk a **var folyamatbetolt:** változó lesz, amit szintén egy önálló objektummal **loaderInfo**. –al látunk el. Ez a folyamatos betöltés byte mennyiség lesz **bytesLoaded;**

Éppen mennyinél tart a betöltés byte-ban mérve a teljes betöltéshez képest. Még kell egy utolsó változó ami a „folyamatbetolt” és a „teljesbetolt” változókat elosztja egymással **var totalertek:Number=folyamatbetolt/teljesbetolt;** Ezt az értéket fogjuk szorozni 100-al és ellátjuk egy „%” jelzéssel. (27a,kép)

a,

```
ACTIONS - FRAME
1  stop ();
2
3  addEventListener (Event.ENTER_FRAME, betolto);
4
5  function betolto(e:Event):void {
6      var teljesbetolt:Number=loaderInfo.bytesTotal;
7      var folyamatbetolt:Number=loaderInfo.bytesLoaded;
8      var totalertek:Number=folyamatbetolt/teljesbetolt;
9  }
```

28. Az elosztás után meg kell adnunk egy feltételes utasítást az „if” utasítással. A zárójelek közé beírjuk az állításokat. **if (folyamatbetolt==teljesbetolt)** Ha a „folyamatbetolt” értéke egyenlő a „teljesbetolt” értékével akkor a lejátszó mit tegyen? Kezdjük egy új tömb utasítással a kapcsos zárójel megnyitásával. Megkérjük, hogy a két tömbben lévő utasításokat hajtsa végre. Először is törölje az eseményünket

10. sor **removeEventListener (Event.Enter_Frame, betolto);** (Ha most megnyomjuk az AutoFormat gombot a flash hibát fog jelezni,mivel nem teljes a kódunk. Ezért nem tudjuk használni ezt a módszert a pontos vessző használatára. Így másoljuk azt az egér segítségével.) Kérjük meg, hogy lépjen a 2-es kulcskockára, ahol elkezdődik az animációnk **gotoAndPlay (2);** A 12.sor **}** zárójellel zárjuk be az esemény törlés és továbblépést tartalmazó tömbünket. Az else feltételvizsgáló megadásával újra nyissunk meg a **{** zárójellel

egy tömböt. Ebben megadjuk a lejátszó számára, hogy milyen mozi clipet és példányokat vegyen elő paraméterekkel ellátva a feltétel teljesülése után, és ezekkel mit tegyen. Vegye elő az `elotolto_mc`-ben a `preloadersav_mc`-t, majd azt a regisztrációs pontjánál kezdődően az x tengelyen mozgatva töltsse ki teljes mértékig. Írjuk ezt a 13. sorba

```
elotolto_mc.preloadersav_mc.scaleX=totalertek;
```

A 14. sorba írva szintén vegye elő az `elotolto_mc.kijelzo_txt` dinamikus text példányt aminek típusa `text` lesz. Ez egyenlő lesz a tizedes jegyeket nem tartalmazó számmal `Math.floor`-al azaz a teljes betöltés számával amit megszorozunk 100-al. Ehhez az értékhez pedig hozzáadunk egy százalék karaktert. Végül bezárjuk a nyitva maradt tömböket. Tehát írjuk be a következő kódot.

```
elotolto_mc.kijelzo_txt.text=Math.floor(totalertek*100) + "%";
```

Íme a teljes kód leírva és képben (28a, kép)

```
stop();
```

```
addEventListener(Event.ENTER_FRAME, betolto);
```

```
function betolto(e:Event):void {
```

```
    var teljesbetolt:Number=loaderInfo.bytesTotal;
```

```
    var folyamatbetolt:Number=loaderInfo.bytesLoaded;
```

```
    var totalertek:Number=folyamatbetolt/teljesbetolt;
```

```
    if (folyamatbetolt==teljesbetolt) {
```

```
        removeEventListener(Event.ENTER_FRAME,betolto);
```

```
        gotoAndPlay(2);
```

```
    } else {
```

```
        elotolto_mc.preloadersav_mc.scaleX=totalertek;
```

```
        elotolto_mc.kijelzo_txt.text=Math.floor(totalertek*100)+"%";
```

```
    }
```

```
}
```

