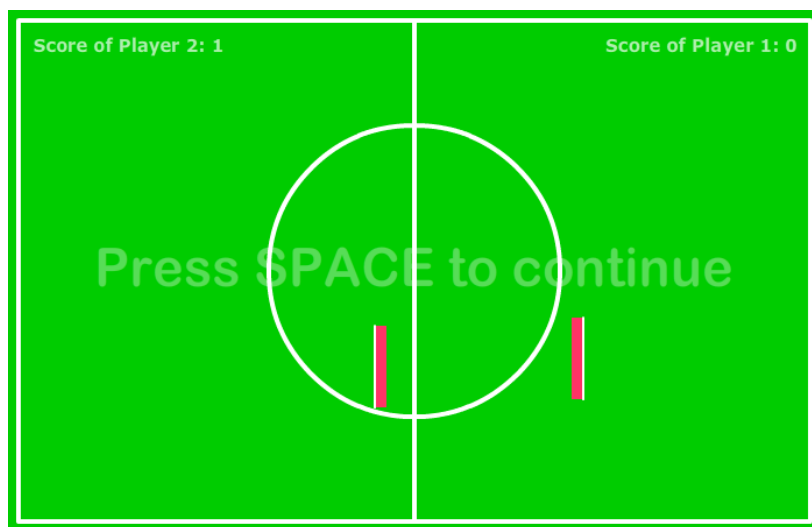


## ZADANIE 1

Zaprojektuj grafikę do popularnej gry Ping-pong



Pola tekstowe wstawiane są ręcznie powinny mieć atrybut „dynamic text” (ustawienie w panelu Właściwości); jeśli pola tekstowe powoływane są dynamicznie, to należy pamiętać o osadzeniu używanej czcionki.  
Gracze (rakiety) powinny mieć postać klipu filmowego (*MovieClip* lub prostszego elementu *Sprite*); podobnie jak cztery ściany pola gry – bok lewy, bok prawy, ściana górna i ściana dolna. Jeśli elementy te wstawiane są ręcznie (jako klipy) to muszą mieć nadane nazwy instancji.  
Zaprojektuj piłeczkę i dodaj ją do projektu (obiekt *MovieClip* lub *Sprite*).

## ZADANIE 2

Zaprojektuj sposób poruszania się graczy.

- I. Jeśli planujemy swobodne poruszanie się graczy w obrębie własnej połowy pola gry to najlepiej jest skorzystać z metody `startDrag()` z parametrem ograniczającym pole poruszania się. Poruszanie odbywa się za pomocą myszy (`MouseEvent`).

Opis metody `startDrag()` wygląda następująco:

```
startDrag(lockCenter:Boolean = false, bounds:Rectangle = null):void
```

Obiekt `Rectangle` reprezentuje obszar prostokątny o podanym lewym górnym punkcie narożnym (`x`, `y`), podanej wysokości i podanej szerokości. Do utworzenia obiektu `Rectangle` można użyć konstruktora `new Rectangle()`.

```
var boundaries1 = new Rectangle(355,12,325,358);  
event.target.startDrag(true,boundaries1);
```

Działanie metody `startDrag()` kończy:

```
stopDrag():void
```

- II. Jeśli gra będzie obsługiwana na urządzeniu z ekranem dotykowym, to odpowiednimi metodami będą:

```
startTouchDrag(touchPointID:int, lockCenter:Boolean = false, bounds:Rectangle = null):void
```

```
stopTouchDrag(touchPointID:int):void
```

touchPointID:[int](#) — Liczba całkowita, która ma być przypisana do punktu dotyku.

Nasłuchiwanymi zdarzeniami rozgłaszanymi są w tym wypadku zdarzenia z klasy [TouchEvent](#):

```
TouchEvent.TOUCH_BEGIN
```

```
TouchEvent.TOUCH_END
```

III. Jeśli gracze będą używać przycisków z klawiatury to należy nasłuchiwać zdarzenia rozgłaszanego klawiatury:

```
stage.addEventListener(KeyboardEvent.KEY_DOWN, ...);
```

```
stage.addEventListener(KeyboardEvent.KEY_UP, ...);
```

Programując dalsze działanie najlepiej posługiwać się kodem ASCII przycisku, np. dla spacji jest to 32:

```
if(event.keyCode==32){...}
```

### Zaprojektuj sposób poruszania się piłki.

Poruszanie się piłki związane jest ze zdarzeniem [ENTER\\_FRAME](#). Piłka musi zmieniać kierunek ruchu w zależności od napotkanej przeszkody lub wypadać (znikać) z pola gry w przypadku błędu zawodnika.

Piłkę dodajemy do listy wyświetlania za pomocą metody [addChild\(\)](#) lub [addChildAt\(\)](#) (gdy dodajemy na określony poziom), a zdejmujemy z listy wyświetlania metodą [removeChild\(\)](#).

Kolizje wykrywane są dzięki metodzie [hitTestObject](#):

```
ball_mc.hitTestObject(player1_mc) ; //metoda zwraca true w przypadku kolizji piłki z graczem.
```

### Zaprojektuj sposób przyznawania punktacji i możliwość wznawiania gry.

