# **Ćwiczenie 12 – Podstawy ActionScript 3.0**

Wykonamy teraz kilka drobnych ćwiczeń pokazujących możliwości sterowania obiektami poprzez ActionScript 3.0. "Action Script" to prosty język skryptowy umożliwiający dodawanie do projektu wielu ciekawych elementów takich jak: linki do stron WWW, opcji drukowania, przewijania tekstu, podmieniania kursora, sterowania obiektem za pomocą klawiatury i wielu innych.

Pobierz przykłady (http://jsekulska.kis.p.lodz.pl/studia.htm).

#### Przykład 1 – Sterowanie listwą czasową

Otwórz nowy plik. Nie zmieniaj właściwości dokumentu. Wstaw nowy symbol typu "przycisk" i określ jego wygląd w klatkach **W górę, Over, Down** i **Hit**.

	9		W g	Over	Down	Н	lit	
🕤 tekst	1.		•	•	•			go+To
√ tło	•	•	•	•	•			

Umieść przycisk na głównej scenie i nazwij go "button1\_btn".

Biblioteka	Ustawienia predefiniow	Właściwości	
N	button1_btn		
Ľ	Przycisk		-
Instanc	ja: p1	Zami	eń
<b>▽ POŁOŻ</b>	ENIE I ROZMIAR		-

W klatce 10 wstaw klatkę kluczową (F6) i usuń z niej zawartość:



W klatce nr 1 i 10 otwórz panel **Operacje** i wstaw akcję stop() ; W klatce nr 1 wstaw dodatkowo kod:

```
function przekierowanie(myevent:MouseEvent):void {
    gotoAndPlay(10);
}
```

button1\_btn.addEventListener(MouseEvent.CLICK, przekierowanie);

	9		5	10
🕤 Warstw 🧷				

Przetestuj film (Ctrl+Enter). Naciśnięcie przycisku powinno nas przekierować do pustej klatki nr 10 i zatrzymać animację w tej klatce.

## Przykład 2 – Przeciąganie obiektów

Do tego ćwiczenia można wykorzystać gotowy już przycisk w stworzonym przed chwilą pliku – będzie to kontynuacja poprzedniego przykładu. Zaznacz klatkę nr 10 i wstaw tam "nową" instancję przycisku z biblioteki. Nazwij wstawiony przycisk "button2\_btn".

W tej samej klatce otwórz panel **Operacje** i poniżej akcji stop () ; wpisz kod:

```
function startDragging(event:MouseEvent):void{
    this.startDrag();
}
function stopDragging(event:MouseEvent):void{
    this.stopDrag();
}
button2_btn.addEventListener(MouseEvent.MOUSE_DOWN, startDragging);
button2_btn.addEventListener(MouseEvent.CLICK, stopDragging);
```

W tym momencie trzymając klawisz myszy można go przenosić, a gdy klawisz zwolnisz to obiekt pozostanie tam gdzie go zostawisz.

#### Przykład 3 – Przemieszczanie klipu

1. Otwórz nowy plik. Narysuj koło i przekonwertuj je do symbolu klip (F8).

	Konwertuj na symbol								
$\frown$	Nazwa: ball	ОК							
	Typ: Klip filmowy   ▼ Odniesienie:	Anuluj							
	Folder: Folder główny biblioteki								
_	Zaawansowane 🕨								

- 2. W oknie Właściwości klipu w polu Nazwa Instancji wpisz "ball\_mc".
- 3. Otwórz okienko **Operacji** i wpisz kod:

```
import flash.events.Event;
function ballMove(event:Event):void{
ball_mc.x++
}
stage.addEventListener(Event.ENTER_FRAME, ballMove);
```

Uruchom film (Ctrl+Enter)

Jak widać coś, co narysowałeś przemieszcza się ze stałą prędkością w lewo. Czy to Ci nie przypomina liniowej animacji, którą do tej pory robiłeś przy pomocy *automatycznej animacji*? Może prędkość nie jest ta sama, ale daje to te same efekty.

Wprowadzimy teraz zmienną o nazwie **speed** i zmodyfikujmy kod:

```
import flash.events.Event;
var speed:Number = 5;
function ballMove(event:Event):void{
        ball_mc.x += speed;
}
stage.addEventListener(Event.ENTER_FRAME, ballMove);
```

### Przykład 4 – Widoczność klipu

ActionScript jest językiem zorientowanym obiektowo. Może zauważyłeś pewną rzecz... Często w kodzie AS używana jest kropka. Służy ona do adresowania symboli oraz do określania właściwości i metod. Jak zmienić właściwość klipu? To proste ćwiczenie to wyjaśni. Będziemy zmniejszać właściwość alpha (widoczność) klipu filmowego.

!. Utwórz nowy plik Flasha z domyślnymi parametrami.

- 2. Narysuj koło i przekonwertuj je do klipu (F8). Nazwij instancję klipu nazwą "ball\_mc".
- 3. Otwórz okienko **Operacji** i wpisz poniższy skrypt:

```
import flash.events.Event;
```

```
function alphaChange(event:Event):void{
    ball_mc.alpha -= 0.01;
}
stage.addEventListener(Event.ENTER FRAME, alphaChange);
```

```
Przetestuj animację (CTRL + ENTER).
```

Jest to dość prosta animacja, która po każdym przejściu klipu do następnej klatki zmniejsza o 0.01 parametr alpha tego konkretnego klipu. Niestety, gdybyśmy chcieli, aby nasz klip "pulsował" musimy już bardzo skomplikować skrypt:

```
import flash.events.Event;
```

```
var switchA:Boolean = true;
function alphaChange(event:Event):void{
     if (ball mc.alpha>0 && switchA == true) {
           ball mc.alpha-=0.05;
                 if (ball mc.alpha<=0) {</pre>
                       switchA=false;
                 }
           }
     if (switchA==false) {
           if(ball mc.alpha<1) {</pre>
           ball mc.alpha+=0.05;
                 if(ball mc.alpha>=1) {
                       switchA=true;
                 }
           }
      }
}
stage.addEventListener(Event.ENTER FRAME, alphaChange);
```

Zauważ, że klip szybciej zmienia swoją widoczność, gdyż zmieniliśmy wartość 0.01 na 0.05.

Właściwości klipów to także wymieniono już współrzędne x i y, a także width i height oraz scaleX i scaleY (dzięki nim możesz manipulować wielkością klipu).

## Przykład 5 – Poruszanie obiektem strzałkami z klawiatury

Teraz obiekt **key**, który reaguje na naciskane klawisze klawiatury (najczęściej wykorzystuje się tu strzałki). Jeśli będziesz chciał zrobić grę we Flashu (np. łapanie czegoś, biegającym stworkiem), to wtedy docenisz obiekt key. Obiekt ten reaguje na naciśniecie jakiegoś przycisku i wykonuje określoną akcję.

1. Utwórz nowy film Flasha. Ustaw 60 kl/s. (to ważne!).

- 2. Narysuj dowolny kształt i przekonwertuj go w klip.
- 3. Nadaj instancji tego klipu nazwę "circle".
- 4. Otwórz okienko Operacji i wpisz:

```
function clipMove(event:KeyboardEvent):void{
```

```
if (event.keyCode==Keyboard.RIGHT) {
    circle.x ++;
}
if (event.keyCode==Keyboard.LEFT) {
    circle.x --;
}
stage.addEventListener(KeyboardEvent.KEY_DOWN, clipMove);
```

5. Ustaw klon w na środku stołu montażowego.

6. I wciśnij Ctrl+Enter.

Działa to dość prosto. Obiekt key sprawdza, czy przycisk został naciśnięty. Jeśli tak się stało zostaje wykonana akcja w nawiasach klamrowych.

#### Przykład 6 – Poruszanie obiektem strzałkami z klawiatury (kombinacja klawiszy)

Wykonanym teraz zbliżone ćwiczenie do poprzedniego, z tym, że wykorzystamy wszystkie strzałki (na boki, do góry i w dół oraz ich kombinację. Klip będzie także zmieniał swoją orientację względem kierunku ruchu oraz wracał na scenę, jeśli wyjdzie poza nią.

Utwórz animację, którą chcesz poruszać, w formie klipu. Nazwij jego instancję po prostu "klip". I wpisz w panel operacji:

```
import flash.events.Event;
var speed:Number = 10;
var hold_U:Boolean = false;
var hold_D:Boolean = false;
var hold_L:Boolean = false;
var hold_R:Boolean = false;
function keyRespond(event:KeyboardEvent):void{
    switch(event.keyCode) {
    case 37:
        hold_L = true;
```

```
break;
        case 38:
          hold U = true;
          break;
        case 39:
          hold R = true;
          break;
        case 40:
          hold D = true;
          break;
      }
}
// adding this new function to clear key states when released
function keyLetGo(event:KeyboardEvent):void
    {
      switch(event.keyCode) {
        case 37:
          hold L = false;
          break;
        case 38:
          hold U = false;
          break;
        case 39:
          hold R = false;
          break;
        case 40:
          hold D = false;
          break;
      }
    }
function clipMove(event:KeyboardEvent):void{
if (hold L==true) {
     klip.x -= speed;
     klip.rotation = 270;
ł
if (hold R==true) {
     klip.x += speed;
     klip.rotation = 90;
if (hold U==true) {
     klip.y -= speed;
     klip.rotation = 0;
3
if (hold D==true) {
     klip.y += speed;
     klip.rotation = 180;
}
// move diagonally
if (hold L==true && hold U==true) {
     klip.rotation = 315;
     }
if (hold R==true && hold U==true) {
     klip.rotation = 45;
}
if (hold_L==true && hold_D==true) {
```

```
klip.rotation = 225;
}
if (hold R==true && hold D==true) {
     klip.rotation = 135;
}
// loop to opposite side of the area when the clip travels off-screen
if (klip.y<0) {</pre>
     klip.y = 400;
}
if (klip.y>400) {
klip.y = 0;
ł
if (klip.x<0) {</pre>
klip.x = 550;
}
if (klip.x>550) {
klip.x = 0;
}
}
stage.addEventListener(KeyboardEvent.KEY_DOWN, keyRespond);
```

stage.addEventListener(KeyboardEvent.KEY\_UP, keyLetGo); stage.addEventListener(KeyboardEvent.KEY\_DOWN, clipMove);