

Lekcja 5 – Animacja

plotkarka.eu (kodujący stwór, lekcja 43)

Na czym polega animacja? Rysujemy obiekt, potem wymazujemy ekran, ustawiamy nowe współrzędne i powtarzamy te trzy instrukcje wiele razy. W JS animację realizujemy rekurencyjnie.

Funkcje **clearTimeout** i **setTimeout** odpowiedzialne są za uruchamianie funkcji **animacja** co określony przedział czasu w milisekundach – zmienna **skok**. Zmienna **czas** jest tzw. „uchwytem” funkcji.

```
var w = c.canvas.width;
var h = c.canvas.height;
var skok = 10;
var czas;
function animacja() {
  c.clearRect(0, 0, w, h);
  //rysowanie i przesuwanie obiektu
  clearTimeout(czas);
  czas = setTimeout(animacja, skok);
}

animacja();
```

Przesuwanie

Aby przesunąć obiekt musimy za każdym zwiększyć jego współrzędne (X,Y) o określony skok (vX,vY). Im większe wartości vX i vY, tym obiekt będzie się szybciej przesuwał.

Odbijanie

Gdy obiekt dociera do brzegu powinien się „odbić”, tzn. zmienić kierunek, skąd przyleciał na przeciwny. Należy uwzględnić również szerokość i wysokość obiektu (tutaj bok kwadratu).

```
//odbicia od brzegów
if (x < 0){vx=-vx;} // lewy
if (x+bok > w){vx=-vx;} // prawy
if (y < 0){vy=-vy;} // górny
if (y+bok > h){vy=-vy;} // dolny
```

```
//początkowe położenie
var x = 0;
var y = 0;
//początkowa prędkość
var vx = 2;
var vy = 1;

//w głównej części programu
x = x + vx;
y = y + vy;
```

Obracanie

Jeśli obiekt ma zdefiniowany kąt obrotu, to postępujemy podobnie jak podczas przesuwania – zwiększamy kąt o jakiś elementarny składnik – dK.

```
//początkowy kąt
var kat = 0;
var vk = 1;

//w animacji
kat = kat + vk;
if (kat>=360) {vk=-vk}
```

```
var x = 100;
var y = 100;
var vx = 3;
var vy = 2;
var bok = 50;
var w = c.canvas.width;
var h = c.canvas.height;
var krok = 10;
var czas;

function animacja() {
  //wymazywanie ekranu
  c.clearRect(0, 0, w, h);
  //rysowanie brzegu
  c.strokeRect(0, 0, w, h);
  //rysowanie obiektu
  c.fillRect(x, y, bok, bok);
  // wyliczanie nowych współrzędnych
  x=x+vx;
  y=y+vy;
  //odbijamy od brzegów
  if (x+bok > w || x < 0){vx=-vx;}
  if (y+bok > h || y < 0){vy=-vy;}
  clearTimeout(czas);
  czas = setTimeout(animacja, krok);
}

// uruchamiamy rekurencyjną animację
animacja();
```

Obrotowy kwadrat

Kwadrat przesuwa się i jednocześnie obraca wokół własnego środka - zmianie muszą ulec warunki odbijania ze względu na inny środek obiektu.

Po odbiciu kwadrat kręci się w drugą stronę.

```
//obrotowy kwadrat
function kwa0(xs,ys,bok,kat,kolor){
  c.save();
  c.translate(xs-bok/2,ys-bok/2);
  c.rotate(Math.PI*kat/180);
  c.fillStyle=kolor;
  c.fillRect(-bok/2,-bok/2,bok,bok);
  c.restore();
}
```

Wywołanie nowej funkcji **KwaO** i zmienione warunki odbijania. Dodatkowo warunek na reset kąta, po wykonaniu pełnego obrotu.

```
//wewnątrz funkcji animacja
kwa0(x,y,bok,kat,"red");
x=x+vx; y=y+vy; kat=kat+vk;
if (x > w || x < bok){vx=-vx; vk=-vk;}
if (y > h || y < bok){vy=-vy; vk=-vk;}
if (kat>=360){kat=0;}
```