

		Autor
Dominik Cisko		
		Grupa laboratoryjna
Lab 1		
		Temat
Konfigurowalne tło, gradient		
		Nazwy plików
GradientBackground.html		
		Typ dokumentu
HTML 5		
		Kodowanie znaków
UTF-8		
		Tytuł strony
Gradient		
		Zawartość strony
<pre> var kontener = document.querySelector(".container") var kontener_gradient = document.getElementById("gradient") var kierunek = document.getElementsByName("kierunek")[0] var ksztalt = document.getElementsByName("ksztalt")[0] var typ = document.getElementsByName("typ")[0] var amount = document.getElementById("amountOfColors"); var kolor1 = document.querySelector(".kolor1") var kolor2 = document.querySelector(".kolor2") var kolor3 = document.querySelector(".kolor3") var kolor4 = document.querySelector(".kolor4") </pre>		

```
var outputRed1 = document.querySelector('#outputRed1');  
var outputGreen1 = document.querySelector('#outputGreen1');  
var outputBlue1 = document.querySelector('#outputBlue1');
```

```
var outputRed2 = document.querySelector('#outputRed2');  
var outputGreen2 = document.querySelector('#outputGreen2');  
var outputBlue2 = document.querySelector('#outputBlue2');
```

```
var outputRed3 = document.querySelector('#outputRed3');  
var outputGreen3 = document.querySelector('#outputGreen3');  
var outputBlue3 = document.querySelector('#outputBlue3');
```

```
var outputRed4 = document.querySelector('#outputRed4');  
var outputGreen4 = document.querySelector('#outputGreen4');  
var outputBlue4 = document.querySelector('#outputBlue4');
```

```
var red1 = document.querySelector('#red1')  
var green1 = document.querySelector('#green1')  
var blue1 = document.querySelector('#blue1')  
var outputRed1 = document.querySelector('#outputRed1')  
var outputGreen1 = document.querySelector('#outputGreen1')  
var outputBlue1 = document.querySelector('#outputBlue1')  
var wartoscHex1;
```

```
var red2 = document.querySelector('#red2')  
var green2 = document.querySelector('#green2')  
var blue2 = document.querySelector('#blue2')
```

```
var outputRed2 = document.querySelector('#outputRed2')
var outputGreen2 = document.querySelector('#outputGreen2')
var outputBlue2 = document.querySelector('#outputBlue2')
var wartoscHex2;
```

```
var red3 = document.querySelector('#red3')
var green3 = document.querySelector('#green3')
var blue3 = document.querySelector('#blue3')
var outputRed3 = document.querySelector('#outputRed3')
var outputGreen3 = document.querySelector('#outputGreen3')
var outputBlue3 = document.querySelector('#outputBlue3')
var wartoscHex3;
```

```
var red4 = document.querySelector('#red4')
var green4 = document.querySelector('#green4')
var blue4 = document.querySelector('#blue4')
var outputRed4 = document.querySelector('#outputRed4')
var outputGreen4 = document.querySelector('#outputGreen4')
var outputBlue4 = document.querySelector('#outputBlue4')
var wartoscHex4;
```

```
document.getElementById("thirdColor").style.display = "none";
document.getElementById("fourthColor").style.display = "none";
```

```
function ustawKolorField(){
    if(outputRed1.value<0){outputRed1.value=0}
    if(outputRed1.value>255){outputRed1.value=255}
```

```
if(outputRed1.value==""){outputRed1.value=0}
```

```
if(outputGreen1.value<0){outputGreen1.value=0}
```

```
if(outputGreen1.value>255){outputGreen1.value=255}
```

```
if(outputGreen1.value==""){outputGreen1.value=0}
```

```
if(outputBlue1.value<0){outputBlue1.value=0}
```

```
if(outputBlue1.value>255){outputBlue1.value=255}
```

```
if(outputBlue1.value==""){outputBlue1.value=0}
```

```
if(outputRed2.value<0){outputRed2.value=0}
```

```
if(outputRed2.value>255){outputRed2.value=255}
```

```
if(outputRed2.value==""){outputRed2.value=0}
```

```
if(outputGreen2.value<0){outputGreen2.value=0}
```

```
if(outputGreen2.value>255){outputGreen2.value=255}
```

```
if(outputGreen2.value==""){outputGreen2.value=0}
```

```
if(outputBlue2.value<0){outputBlue2.value=0}
```

```
if(outputBlue2.value>255){outputBlue2.value=255}
```

```
if(outputBlue2.value==""){outputBlue2.value=0}
```

```
if(outputRed3.value<0){outputRed3.value=0}
```

```
if(outputRed3.value>255){outputRed3.value=255}
```

```
if(outputRed3.value==""){outputRed3.value=0}
```

```
if(outputGreen3.value<0){outputGreen3.value=0}
```

```
if(outputGreen3.value>255){outputGreen3.value=255}
```

```
if(outputGreen3.value==""){outputGreen3.value=0}
```

```
if(outputBlue3.value<0){outputBlue3.value=0}
```

```
if(outputBlue3.value>255){outputBlue3.value=255}
```

```
if(outputBlue3.value==""){outputBlue3.value=0}
```

```
if(outputRed4.value<0){outputRed4.value=0}
```

```
if(outputRed4.value>255){outputRed4.value=255}
```

```
if(outputRed4.value==""){outputRed4.value=0}
```

```
if(outputGreen4.value<0){outputGreen4.value=0}
```

```
if(outputGreen4.value>255){outputGreen4.value=255}
```

```
if(outputGreen4.value==""){outputGreen4.value=0}
```

```
if(outputBlue4.value<0){outputBlue4.value=0}
```

```
if(outputBlue4.value>255){outputBlue4.value=255}
```

```
if(outputBlue4.value==""){outputBlue4.value=0}
```

```
var red1_hex = parseInt(outputRed1.value, 10).toString(16);
```

```
var green1_hex = parseInt(outputGreen1.value, 10).toString(16);
```

```
var blue1_hex = parseInt(outputBlue1.value, 10).toString(16);
```

```
wartoscHex1 = "#" + pad(red1_hex) + pad(green1_hex) + pad(blue1_hex);
```

```
kolor1.value = wartoscHex1;
```

```
red1.value = outputRed1.value;
```

```
green1.value = outputGreen1.value;
```

```
blue1.value = outputBlue1.value;
```

```
var red2_hex = parseInt(outputRed2.value, 10).toString(16);
```

```
var green2_hex = parseInt(outputGreen2.value, 10).toString(16);
```

```
var blue2_hex = parseInt(outputBlue2.value, 10).toString(16);
```

```
wartoscHex2 = "#" + pad(red2_hex) + pad(green2_hex) + pad(blue2_hex);
```

```
kolor2.value = wartoscHex2;
```

```
red2.value = outputRed2.value;
```

```
green2.value = outputGreen2.value;
```

```
blue2.value = outputBlue2.value;
```

```
var red3_hex = parseInt(outputRed3.value, 10).toString(16);
```

```
var green3_hex = parseInt(outputGreen3.value, 10).toString(16);
```

```
var blue3_hex = parseInt(outputBlue3.value, 10).toString(16);
```

```
wartoscHex3 = "#" + pad(red3_hex) + pad(green3_hex) + pad(blue3_hex);
```

```
kolor3.value = wartoscHex3;
```

```
red3.value = outputRed3.value;
```

```
green3.value = outputGreen3.value;
```

```
blue3.value = outputBlue3.value;
```

```
var red4_hex = parseInt(outputRed4.value, 10).toString(16);
```

```
var green4_hex = parseInt(outputGreen4.value, 10).toString(16);
```

```
var blue4_hex = parseInt(outputBlue4.value, 10).toString(16);
```

```
wartoscHex4 = "#" + pad(red4_hex) + pad(green4_hex) + pad(blue4_hex);
```

```
kolor4.value = wartoscHex4;
```

```
red4.value = outputRed4.value;
```

```
green4.value = outputGreen4.value;
```

```
blue4.value = outputBlue4.value;
```

```
if(amount.value == 2)
```

```
{
```

```
if(typ.value == 'linear-gradient')
```

```
{
```

```
kontener_gradient.style.background = "linear-gradient("+kierunek.value+ ", "+ kolor1.value+  
", "+ kolor2.value+ ")";
```

```
kierunek.style.display="block";
```

```
ksztalt.style.display="none";
```

```
}
```

```
else if(typ.value == "radial-gradient")
```

```
{
```

```
kontener_gradient.style.background = "radial-gradient("+ksztalt.value+ ", "+ kolor1.value+  
", "+ kolor2.value+ ")";
```

```
kierunek.style.display="none";
```

```
ksztalt.style.display="block";
```

```
}
```

```
else
```

```
{
```

```
console.log("Zły gradient")
```

```
}
```

```
}
```

```
if(amount.value == 3)

{

if(typ.value == 'linear-gradient')

{

    kontener_gradient.style.background = "linear-gradient("+kierunek.value+ ", "+ kolor1.value+
    ", "+ kolor2.value+ ", "+ kolor3.value+")";

    kierunek.style.display="block";

    ksztalt.style.display="none";

}

else if(typ.value == "radial-gradient")

{

    kontener_gradient.style.background = "radial-gradient("+ksztalt.value+ ", "+ kolor1.value+
    ", "+ kolor2.value+ ", "+ kolor3.value+")";

    kierunek.style.display="none";

    ksztalt.style.display="block";

}

else

{

    console.log("Zły gradient")

}

}

if(amount.value == 4)

{

if(typ.value == 'linear-gradient')

{

    kontener_gradient.style.background = "linear-gradient("+kierunek.value+ ", "+ kolor1.value+
    ", "+ kolor2.value+ ", "+ kolor3.value+", "+ kolor4.value+")";

    kierunek.style.display="block";

    ksztalt.style.display="none";

}
```



```

    }

    else if(typ.value == "radial-gradient")
    {
        kontener_gradient.style.background = "radial-gradient("+ksztalt.value+ " , "+ kolor1.value+
        ", "+ kolor2.value+ " , "+ kolor3.value+ " , "+ kolor4.value+ ")";

        kierunek.style.display="none";

        ksztalt.style.display="block";

    }

    else

    {
        console.log("Zły gradient")
    }

}

}

```

```

function ustawKolorSlider(){

var red1_hex = parseInt(red1.value, 10).toString(16);

var green1_hex = parseInt(green1.value, 10).toString(16);

var blue1_hex = parseInt(blue1.value, 10).toString(16);

wartoscHex1 = "#" + pad(red1_hex) + pad(green1_hex) + pad(blue1_hex);

kontener_gradient.style.backgroundColor = wartoscHex1;

kolor1.value = wartoscHex1;


outputRed1.value = red1.value;

outputGreen1.value = green1.value;

outputBlue1.value = blue1.value;


var red2_hex = parseInt(red2.value, 10).toString(16);

```

```
var green2_hex = parseInt(green2.value, 10).toString(16);  
var blue2_hex = parseInt(blue2.value, 10).toString(16);  
wartoscHex2 = "#" + pad(red2_hex) + pad(green2_hex) + pad(blue2_hex);  
kontener_gradient.style.backgroundColor = wartoscHex2;  
kolor2.value = wartoscHex2;
```

```
outputRed2.value = red2.value;  
outputGreen2.value = green2.value;  
outputBlue2.value = blue2.value;
```

```
var red3_hex = parseInt(outputRed3.value, 10).toString(16);  
var green3_hex = parseInt(outputGreen3.value, 10).toString(16);  
var blue3_hex = parseInt(outputBlue3.value, 10).toString(16);  
wartoscHex3 = "#" + pad(red3_hex) + pad(green3_hex) + pad(blue3_hex);  
kolor3.value = wartoscHex3;
```

```
outputRed3.value = red3.value;  
outputGreen3.value = green3.value;  
outputBlue3.value = blue3.value;
```

```
var red4_hex = parseInt(outputRed4.value, 10).toString(16);  
var green4_hex = parseInt(outputGreen4.value, 10).toString(16);  
var blue4_hex = parseInt(outputBlue4.value, 10).toString(16);  
wartoscHex4 = "#" + pad(red4_hex) + pad(green4_hex) + pad(blue4_hex);  
kolor4.value = wartoscHex4;
```

```
outputRed4.value = red4.value;
```

```
outputGreen4.value = green4.value;
```

```
outputBlue4.value = blue4.value;
```

```
if(amount.value == 2)
```

```
{
```

```
if(typ.value == 'linear-gradient')
```

```
{
```

```
kontener_gradient.style.background = "linear-gradient("+kierunek.value+ ", "+kolor1.value+ ", "+  
kolor2.value+ ")";
```

```
kierunek.style.display="block";
```

```
ksztalt.style.display="none";
```

```
}
```

```
else if(typ.value == "radial-gradient")
```

```
{
```

```
kontener_gradient.style.background = "radial-gradient("+ksztalt.value+ ", "+kolor1.value+ ", "+  
kolor2.value+ ")";
```

```
kierunek.style.display="none";
```

```
ksztalt.style.display="block";
```

```
}
```

```
else
```

```
{
```

```
console.log("Zły gradient")
```

```
}
```

```
}
```

```
if(amount.value == 3)
```

```
{
```

```
if(typ.value == 'linear-gradient')
```

```
{

kontener_gradient.style.background = "linear-gradient("+kierunek.value+ ", "+ kolor1.value+ ", "+
kolor2.value+ ", "+ kolor3.value+")";

kierunek.style.display="block";

ksztalt.style.display="none";

}

else if(typ.value == "radial-gradient")

{

kontener_gradient.style.background = "radial-gradient("+ksztalt.value+ ", "+ kolor1.value+ ", "+
kolor2.value+ ", "+ kolor3.value+")";

kierunek.style.display="none";

ksztalt.style.display="block";

}

else

{

console.log("Zły gradient")

}

}


if(amount.value == 4)

{

if(typ.value == 'linear-gradient')

{

kontener_gradient.style.background = "linear-gradient("+kierunek.value+ ", "+ kolor1.value+ ", "+
kolor2.value+ ", "+ kolor3.value+ ", "+ kolor4.value+")";

kierunek.style.display="block";

ksztalt.style.display="none";

}

else if(typ.value == "radial-gradient")
```

```
{

kontener_gradient.style.background = "radial-gradient("+ksztalt.value+ " , "+kolor1.value+ " , "+
kolor2.value+ " , "+kolor3.value+ " , "+kolor4.value+" )";

kierunek.style.display="none";

ksztalt.style.display="block";

}

else

{

console.log("Zły gradient")

}

}

}

function pad(n){ <!-- wstawia zera między elementami liczby hex, gdy je trzeba

if(n.length<2)

{

return "0"+n

}

else{

return n;

}

}

function selectAmountOfColors()

{

ustawKolorField();

var amount = document.getElementById("amountOfColors").value;
```

```
if(amount==2)

{

    document.getElementById("thirdColor").style.display ="none";

    document.getElementById("fourthColor").style.display ="none";

}


if(amount==3)

{

    document.getElementById("thirdColor").style.display ="block";

    document.getElementById("fourthColor").style.display ="none";

}


if(amount==4)

{

    document.getElementById("thirdColor").style.display ="block";

    document.getElementById("fourthColor").style.display ="block";

}

}
```

Założenia projektu

Aplikacja służy do zmiany stylu CSS całego dokumentu HTML, dokładniej tła. Zmiana dokonuje się poprzez przesuwanie sliderów, lub zmianę wartości RGB liczbowo.

Można wybrać ilość kolorów, typ gradientu(linearny, lub radialny), a także podtypy tych gradientów(kierunek dla linearnego, i elipsa/okrąg dla radialnego).

W polach do wprowadzania liczb, wartość nie może przekraczać 255, i być mniejsza niż 0

var kontener_gradient– pobiera kontener 'gradient' z HTML.

var kierunek– pobiera wartość selektora kierunek gradientu.

var kształt– kształt gradientu z selektora, linearny, lub radialny

var typ - typ gradientu radialnego, z selektora

var amount - wartość selektora amountsOfColors, oznacza ile kolorów znajdzie się w gradiencie

var kolor1,2,3,4 – przechowuje odpowiednio kontenery klasy .kolor

var outputRed1 ,2,3,4 – wartość liczbowa koloru czerwony dla odpowiednich kolorów

var outputGreen1 ,2,3,4 - wartość liczbowa koloru zielony dla odpowiednich kolorów

var outputBlue1,2,3,4 - wartość liczbowa koloru niebieski dla odpowiednich kolorów

var red1,2,3,4 – wartość ze slidera, dla koloru czerwony dla odpowiednich kolorów

var green1,2,3,4 - wartość ze slidera, dla koloru zielony dla odpowiednich kolorów

var blue1,2,3,4 - wartość ze slidera, dla koloru niebieski dla odpowiednich kolorów

var wartoscHex1 ,2,3,4 – wartość RGB w postaci hexadecymetralnej dla odpowiednich kolorów

var red1_hex - wartość koloru czerwonego w postaci hexadecymetralnej

var green1_hex - wartość koloru zielonego w postaci hexadecymetralnej

var blue1_hex - wartość koloru niebieskiego w postaci hexadecymetralnej

function ustawKolorSlider() – zmienia tło HTML, gdy zmieniona jest wartość slidera

function ustawKolorField() – zmienia tło HTML, gdy zmieniona jest wartość pola tekstowego

function pad(n) – funkcja przyjmuje liczbę, i zwraca liczbę. Jeżeli liczba ma długość mniejszą od 2, to dopisuje 0 na jej początku, tak aby po złączeniu np., trzech wartości kolorów, powstał kod w języku szesnastkowym

function selectAmountOfColors() – funkcja chowa i pokazuje elementy, odpowiednie za modyfikację koloru 3 i 4, w zależności od tego ile kolorów użytkownik chce mieć w swoim gradiencie.



