

PROGRAMMATION WEB AVANCEE

TP3 SUITE

Canvas

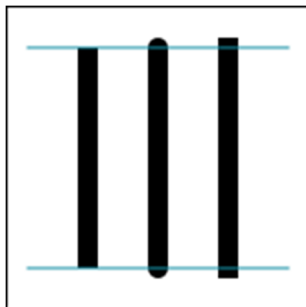
Exercice 1 : lignes et dégradés

Les styles de lignes

Les extrémités

La propriété **lineCap** permet de définir la façon dont les extrémités des chemins sont affichées. Trois valeurs sont admises : **butt** (par défaut), **round** et **square**.

Voici trois lignes, chacune avec un lineCap différent :



Code source

```
<!DOCTYPE html><html>
<head>
<meta charset="utf-8" />
<title>Partie V - Chapitre 3 - lineCap</title>
<style>body { background: black; } canvas { background: white; }</style>
</head>

<body>

<canvas id="canvas" width="150" height="150">
<p>Désolé, votre navigateur ne supporte pas Canvas. Mettez-vous à jour</p>
```

</canvas>

<script>

```
window.onload = function() {  
var canvas = document.querySelector('#canvas');  
var context = canvas.getContext('2d');
```

```
context.lineWidth = 10;
```

```
context.beginPath();  
context.moveTo(40, 20);  
context.lineTo(40, 130);  
context.stroke();
```

```
context.beginPath();  
context.lineCap = 'round';  
context.moveTo(75, 20);  
context.lineTo(75, 130);  
context.stroke();
```

```
context.beginPath();  
context.lineCap = 'square';  
context.moveTo(110, 20);  
context.lineTo(110, 130);  
context.stroke();
```

```
context.beginPath();  
context.lineWidth = 1;  
context.strokeStyle = "#1791a7";  
context.moveTo(10, 20);  
context.lineTo(140, 20);  
context.moveTo(10, 130);  
context.lineTo(140, 130);  
context.stroke();
```

```
};
```

</script>

```
</body>
</html>
```

lineCap s'utilise de la même façon que **lineWidth**, exemple :

```
context.beginPath();

context.lineCap = 'round';

context.moveTo(75, 20);

context.lineTo(75, 130);

context.stroke();
```

Les intersections

Comment gérer la façon dont les angles des chemins sont affichés ?

C'est avec **lineJoin**. Cette propriété reçoit elle aussi trois valeurs différentes : **round**, **bevel** et **miter**(valeur par défaut).



```
<!DOCTYPE html>
```

```
<html>
<head>
<meta charset="utf-8" />
<title>Partie V - Chapitre 3 - lineJoin</title>
<style>body { background: black; } canvas { background: white; }</style>
```

</head>

<body>

<canvas id="canvas" width="150" height="150">

<p>Désolé, votre navigateur ne supporte pas Canvas. Mettez-vous à jour</p>

</canvas>

<script>

```
window.onload = function() {
```

```
var canvas = document.querySelector('#canvas');
```

```
var context = canvas.getContext('2d');
```

```
var y, top = 20, left = 15;
```

```
context.lineWidth = 10;
```

```
y = top;
```

```
context.beginPath();
```

```
context.lineJoin = 'round';
```

```
context.moveTo(left + 0, y + 0);
```

```
context.lineTo(left + 30, y + 30);
```

```
context.lineTo(left + 60, y + 0);
```

```
context.lineTo(left + 90, y + 30);
```

```
context.lineTo(left + 120, y + 0);
```

```
context.stroke();
```

```
y = top + 40;
```

```
context.beginPath();
```

```
context.lineJoin = 'bevel';
```

```
context.moveTo(left + 0, y + 0);
```

```
context.lineTo(left + 30, y + 30);
```

```
context.lineTo(left + 60, y + 0);
```

```
context.lineTo(left + 90, y + 30);
```

```
context.lineTo(left + 120, y + 0);
```

```
context.stroke();
```

```
y = top + 80;
```

```
context.beginPath();
```

```
context.lineJoin = 'miter';
```

```
context.moveTo(left + 0, y + 0);
context.lineTo(left + 30, y + 30);
context.lineTo(left + 60, y + 0);
context.lineTo(left + 90, y + 30);
context.lineTo(left + 120, y + 0);
context.stroke();
};
```

```
</script>
</body>
</html>
```

Les dégradés

Canvas propose deux types de dégradés : linéaire et radial.

Pour créer un dégradé, on commence par créer un objet **canvasGradient** que l'on va assigner à **fillStyle**.

Pour créer un tel objet, on utilise au choix **createLinearGradient()** ou **createRadialGradient()**.

Dégradés linéaires

On a besoin de quatre paramètres pour créer un dégradé linéaire :

```
createLinearGradient(debutX, debutY, finX, finY)
```

debutX et debutY sont les coordonnées du point de départ du dégradé, et finX et finY sont les coordonnées de fin. Faisons un dégradé :

```
var linear = new context.createLinearGradient(0, 0, 150, 150);

context.fillStyle = linear;
```

On va ajouter les couleurs avec **addColorStop(position, couleur)**.

Le premier paramètre, position, est une valeur comprise entre 0 et 1. C'est la position relative de la couleur par rapport au dégradé. Si on met 0.5, la couleur commencera au milieu :

```
var linear = context.createLinearGradient(0, 0, 0, 150);
```

```
linear.addColorStop(0, 'white');  
  
linear.addColorStop(1, '#1791a7');  
  
context.fillStyle = linear;  
  
context.fillRect(20, 20, 110, 110);
```

Pour modifier l'inclinaison du dégradé, il faut modifier les paramètres de **createLinearGradient()**. Par exemple, si on met `createLinearGradient(0, 0, 150, 150)`, la fin du dégradé sera dans le coin inférieur droit, et donc incliné à 45 degrés.

Il est possible de mettre plus de deux **addColorStop()**. Voici un exemple avec quatre :

```
var linear = context.createLinearGradient(0, 0, 0, 150);  
  
linear.addColorStop(0, 'white');  
  
linear.addColorStop(0.5, '#1791a7');  
  
linear.addColorStop(0.5, 'orange');  
  
linear.addColorStop(1, 'white');  
  
context.fillStyle = linear;  
  
context.fillRect(10, 10, 130, 130);
```



Un dégradé linéaire avec Canvas

Code source

```
<!DOCTYPE html>  
<html>  
<head>  
<meta charset="utf-8" />  
<title>Partie V - Chapitre 3 - Exemple 11</title>
```

```
<style>body { background: black; } canvas { background: white; }</style>
</head>

<body>

<canvas id="canvas" width="150" height="150">
<p>Désolé, votre navigateur ne supporte pas Canvas. Mettez-vous à jour</p>
</canvas>

<script>

window.onload = function() {
var canvas = document.querySelector('#canvas');
var context = canvas.getContext('2d');

var linear = context.createLinearGradient(0, 0, 0, 150);
linear.addColorStop(0, 'white');
linear.addColorStop(0.5, '#1791a7');
linear.addColorStop(0.5, 'orange');
linear.addColorStop(1, 'white');

context.fillStyle = linear;
context.fillRect(10, 10, 130, 130);
};

</script>
</body>
</html>
```

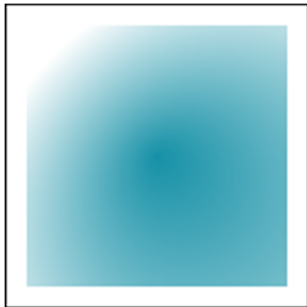
Dégradés radiaux

Du côté des dégradés radiaux, il faut six paramètres :

createRadialGradient(centreX, centreY, centreRayon, finX, finY, finRayon)

Un dégradé radial est défini par deux choses : un premier cercle (le centre) qui fait office de point de départ et un second qui fait office de fin. Ce qui est pratique, c'est que les deux cercles n'ont pas besoin d'avoir la même origine, ce qui permet d'orienter le dégradé :

```
var radial = context.createRadialGradient(75, 75, 0, 130, 130, 150);  
  
radial.addColorStop(0, '#1791a7');  
  
radial.addColorStop(1, 'white');  
  
context.fillStyle = radial;  
  
context.fillRect(10, 10, 130, 130);
```



Un dégradé radial avec Canvas

Code source :

```
<!DOCTYPE html>  
  
<html>  
  
<head>  
  
<meta charset="utf-8" />  
  
<title>Partie V - Chapitre 3 - Exemple 12</title>  
  
<style>  
body { background: black; } canvas { background: white; }  
</style>  
  
</head>  
<body>  
<canvas id="canvas" width="150" height="150">  
  
<p>Désolé, votre navigateur ne supporte pas Canvas. Mettez-vous à jour</p>  
  
</canvas>
```



```
<script>

window.onload = function() { var canvas = document.querySelector('#canvas');

var context = canvas.getContext('2d');
var radial = context.createRadialGradient(75, 75, 0, 130, 130, 150);

radial.addColorStop(0, '#1791a7');

  radial.addColorStop(1, 'white');
context.fillStyle = radial; context.fillRect(10, 10, 130, 130); };
</script>

</body>

</html>
```

Ici, le cercle du centre est... au centre du canvas, et celui de fin en bas à droite. Grâce aux dégradés radiaux, il est possible de créer des bulles assez facilement. La seule condition est que la couleur de fin du dégradé soit transparente, ce qui nécessite l'utilisation d'une couleur RGBA ou HSLA :

```
var radial1 = context.createRadialGradient(0, 0, 10, 100, 20, 150); // fond

radial1.addColorStop(0, '#ddf5f9');

radial1.addColorStop(1, 'ffffff');

var radial2 = context.createRadialGradient(75, 75, 10, 82, 70, 30); // bulle orange

radial2.addColorStop(0, '#ffc55c');

radial2.addColorStop(0.9, '#ffa500');

radial2.addColorStop(1, 'rgba(245,160,6,0)');
```

```
var radial3 = context.createRadialGradient(105, 105, 20, 112, 120, 50); // bulle turquoise

radial3.addColorStop(0, '#86cad2');

radial3.addColorStop(0.9, '#61aeb6');
```

```
radial3.addColorStop(1, 'rgba(159,209,216,0)');

context.fillStyle = radial1;

context.fillRect(10, 10, 130, 130);

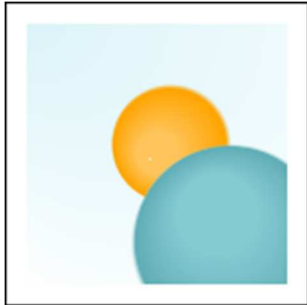
context.fillStyle = radial2;

context.fillRect(10, 10, 130, 130);

context.fillStyle = radial3;

context.fillRect(10, 10, 130, 130);
```

Ce qui donne un dégradé de fond avec deux bulles de couleur :



Deux bulles créées grâce au dégradé radial

Code source

```
<!DOCTYPE html><html>
<head>
<meta charset="utf-8" />
<title>Partie V - Chapitre 3 - Exemple 13</title>
<style>body { background: black; } canvas { background: white; }</style>
</head>

<body>

<canvas id="canvas" width="150" height="150">
<p>Désolé, votre navigateur ne supporte pas Canvas. Mettez-vous à jour</p>
</canvas>

<script>
```

```
window.onload = function() {  
var canvas = document.querySelector('#canvas');  
var context = canvas.getContext('2d');  
  
var radial1 = context.createRadialGradient(0, 0, 10, 100, 20, 150);  
radial1.addColorStop(0, '#ddf5f9');  
radial1.addColorStop(1, '#ffffff');  
  
var radial2 = context.createRadialGradient(75, 75, 10, 82, 70, 30);  
radial2.addColorStop(0, '#ffc55c');  
radial2.addColorStop(0.9, '#ffa500');  
radial2.addColorStop(1, 'rgba(245,160,6,0)');  
  
var radial3 = context.createRadialGradient(105, 105, 20, 112, 120, 50);  
radial3.addColorStop(0, '#86cad2');  
radial3.addColorStop(0.9, '#61aeb6');  
radial3.addColorStop(1, 'rgba(159,209,216,0)');  
  
context.fillStyle = radial1;  
context.fillRect(10, 10, 130, 130);  
context.fillStyle = radial2;  
context.fillRect(10, 10, 130, 130);  
context.fillStyle = radial3;  
context.fillRect(10, 10, 130, 130);  
};  
  
</script>  
</body>  
</html>
```