
La balise <canvas>

3T61U1EW – savoir rédiger pour le multimédia

Jérôme Landré – jerome.landre@univ-reims.fr
bureau C203 (GMP) – bureau H110 (MMI)

Plan

- I- Présentation & organisation
- II- Rappels Javascript
 - II-1) Syntaxe de Javascript
 - II-2) Déplacement d'objets à l'écran en Javascript
- III- <canvas> ?
- IV- Dessiner dans un <canvas>
- V- Traitement d'images avec <canvas>



I- Présentation et organisation

Présentation personnelle

- Jérôme Landré
- Enseignant-chercheur en traitement d'images
- <http://jerome.landre.pagesperso-orange.fr/>



Organisation

- 9h de travaux pratiques (TP)
- 1 évaluation sur machine (1h30)
- Objectifs du cours :
 - Connaître la balise <canvas>
 - Dessiner dans les <canvas>
 - Traiter des images dans des <canvas>

Quelques règles

- Travail attendu :
 - Lire le cours précédent avant d'arriver dans la salle
 - Apprendre par cœur les parties signalées par ♥
- En cours :
 - Pas de téléphone, pas de musique (donc pas d'écouteurs)
 - On ne parle pas, on chuchote
 - On fait le travail demandé
- Conventions :

♥ Apprendre par coeur
english text

Code
Javascript

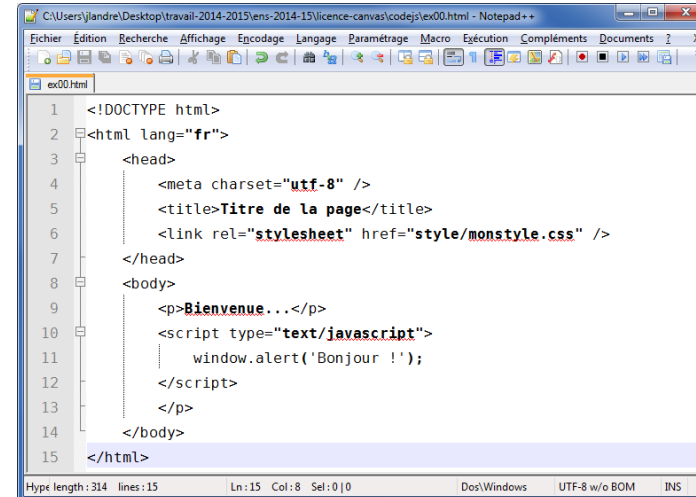
```
window.alert('salut !');
```

```
<script>  
    window.alert('salut !');  
</script>
```

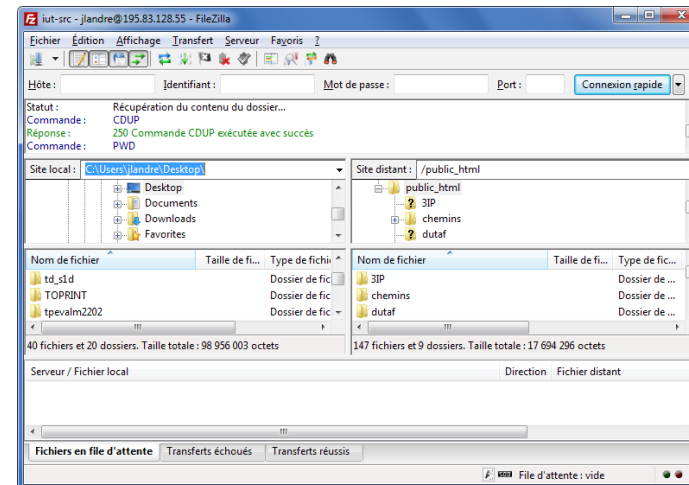
Outils pour javascript

- Pour programmer avec PHP, il faut :

- Un éditeur de textes (notepad++)
- Un client ftp (filezilla)
- Un navigateur

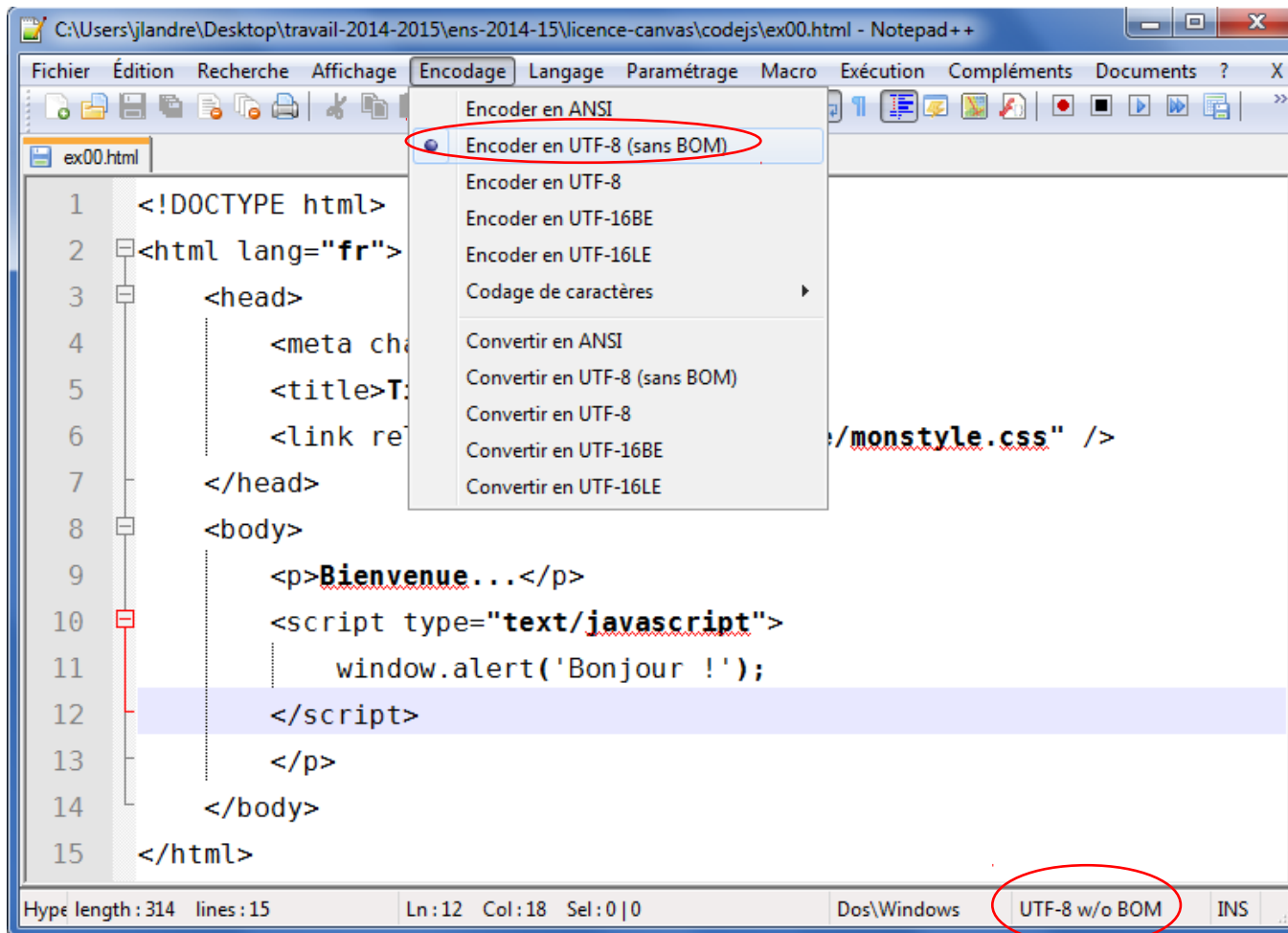
A screenshot of the Notepad++ text editor. The window title is 'C:\Users\jlandre\Desktop\travail-2014-2015\ens-2014-15\licence-canvas\codejs\ex00.html - Notepad++'. The menu bar includes 'Fichier', 'Édition', 'Recherche', 'Affichage', 'Encodage', 'Langage', 'Paramétrage', 'Macro', 'Exécution', 'Compléments', 'Documents', and '?'. The toolbar contains various icons for file operations and editing. The text area shows an HTML document with the following code:

```
1 <!DOCTYPE html>
2 <html lang="fr">
3   <head>
4     <meta charset="utf-8" />
5     <title>Titre de la page</title>
6     <link rel="stylesheet" href="style/monstyle.css" />
7   </head>
8   <body>
9     <p>Bienvenue...</p>
10    <script type="text/javascript">
11      window.alert('Bonjour !');
12    </script>
13    <p>
14  </body>
15 </html>
```

The status bar at the bottom indicates 'Type length: 314 lines: 15', 'Ln: 15 Col: 8 Sel: 0 | 0', 'Dos/Windows', 'UTF-8 w/o BOM', and 'INS'.

Encodage en UTF-8

- Une remarque en passant :
 - Quel que soit l'éditeur de texte choisi, il faudra encoder vos programmes en UTF-8 (sans BOM)





II- Rappels Javascript

C'est quoi Javascript ?

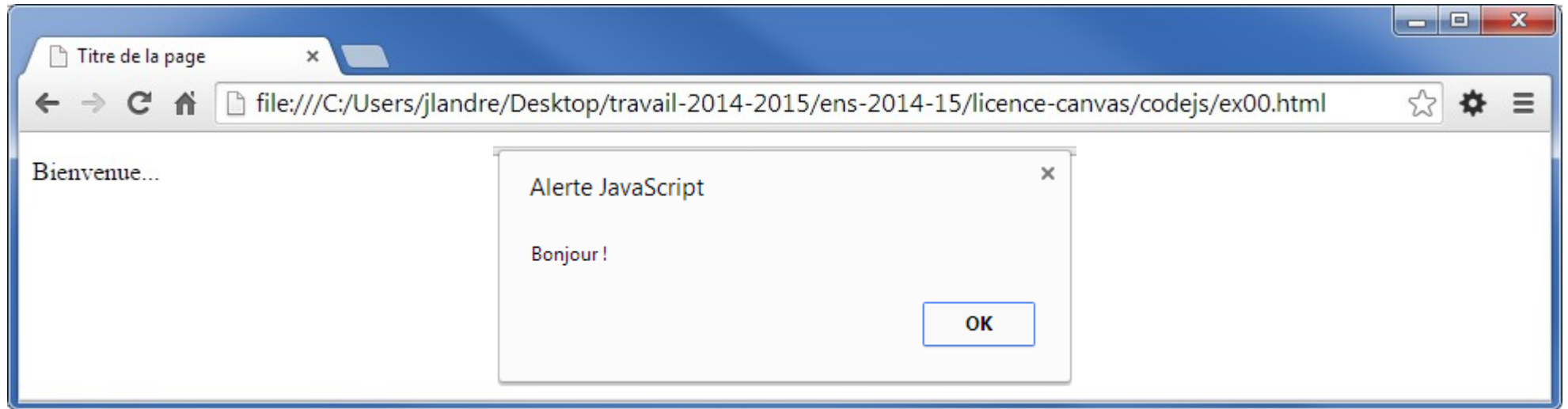
- Langage de programmation orienté objet
- Développé pour le web
- Côté **client**
- Créé en 1995 par Brendan Eich
- Javascript est normalisé sous le nom ECMAScript : « <http://www.ecma-international.org> »

Exemple de code Javascript

```
<!DOCTYPE html>
<html lang="fr">
  <head>
    <meta charset="utf-8" />
    <title>Javascript</title>
    <link rel="stylesheet" href="style/monstyle.css" />
  </head>
  <body>
    <p>Bienvenue...</p>
    <script type="text/javascript">
      window.alert('Bonjour !');
    </script>
  </body>
</html>
```

- Remarque : l'attribut type est optionnel depuis HTML 5.1, par défaut, la balise <script> considère que ce qui suit est du Javascript

Résultat

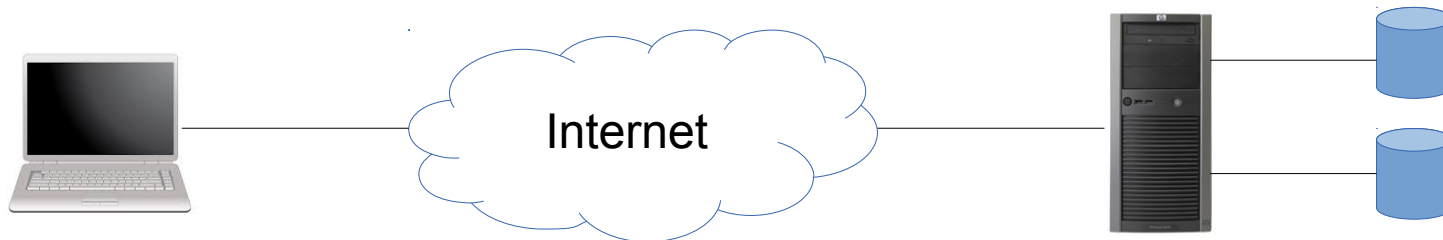


Utiliser Javascript

- On peut utiliser Javascript :
 - En local sur son ordinateur (développement)



- A distance sur un serveur (production)





II-1) Syntaxe de Javascript

Syntaxe Javascript

```
<script type="text/javascript">
  note1=12;
  note2=15;
  somme=note1+note2;
  moyenne=somme/2;
  window.alert('La moyenne vaut : '+moyenne);
  document.write('<p>');
  for (i=10; i>=0; i--) {
    document.write(i+'<br />');
  }
  document.write('</p>');
  if (moyenne>=10) {
    document.write('<p>Félicitations !</p>');
  } else {
    document.write('<p>Il faut faire mieux !</p>');
  }
</script>
```



II-2) Déplacement d'objets à l'écran en Javascript

Déplacer un objet à l'écran

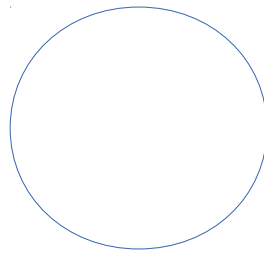
- Un objet positionné en position « absolute » sur l'écran peut être déplacé grâce à Javascript
- On va déplacer une image sur une page web :
 - 1) Télécharger le fichier :
<http://195.83.128.55/~jlandre/mim/lic-bouge-vide.html>
 - 2) Remplir la fonction `bouge()` pour faire bouger le logo vers la droite

Déplacer un objet à l'écran

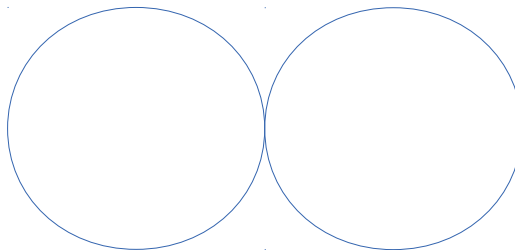
- On va déplacer une image sur une page web :
 - 3) Modifier la fonction `bouge()` pour faire bouger le logo vers la droite puis vers la gauche quand il a atteint le bord droit de l'écran (utiliser `screen.width` pour connaître la largeur de l'écran)
 - 4) Modifier la fonction `bouge()` pour faire bouger le logo vers la droite puis vers la gauche puis vers la droite
 - 5) Modifier la fonction `bouge()` pour faire bouger le logo vers la droite et vers le bas
 - 6) Modifier la fonction `bouge()` pour faire bouger le logo vers la droite et vers le bas puis vers la gauche et vers le haut quand il a atteint le bord en bas à droite de l'écran (`screen.height`)

Déplacer un objet à l'écran

- On va déplacer une image sur une page web :
 - 7) Modifier la fonction `bouge()` pour faire bouger le logo selon un cercle centré sur la page



- 8) Modifier la fonction `bouge()` pour faire bouger le logo selon la lemniscate de Bernouilli centrée sur la page





III- <canvas> ?

W3C

- « The canvas element provides scripts with a resolution-dependent bitmap canvas, which can be used for rendering graphs, game graphics, art, or other visual images on the fly. »
- Spécification HTML 5.1 Nightly du W3C
- <http://www.w3.org/html/wg/drafts/html/master/scripting-1.html#the-canvas-element>

<canvas>

```
<!DOCTYPE html>
<html lang="fr">
  <head>
    <meta charset="utf-8" />
    <title>Mon premier canvas</title>
    <link rel="stylesheet" href="style/monstyle.css" />
  </head>
  <body>
    <p>Voici mon premier canvas</p>

    <canvas id="moncanvas" width="200" height="200">
      Votre navigateur ne supporte pas canvas !
    </canvas>

  </body>
</html>
```



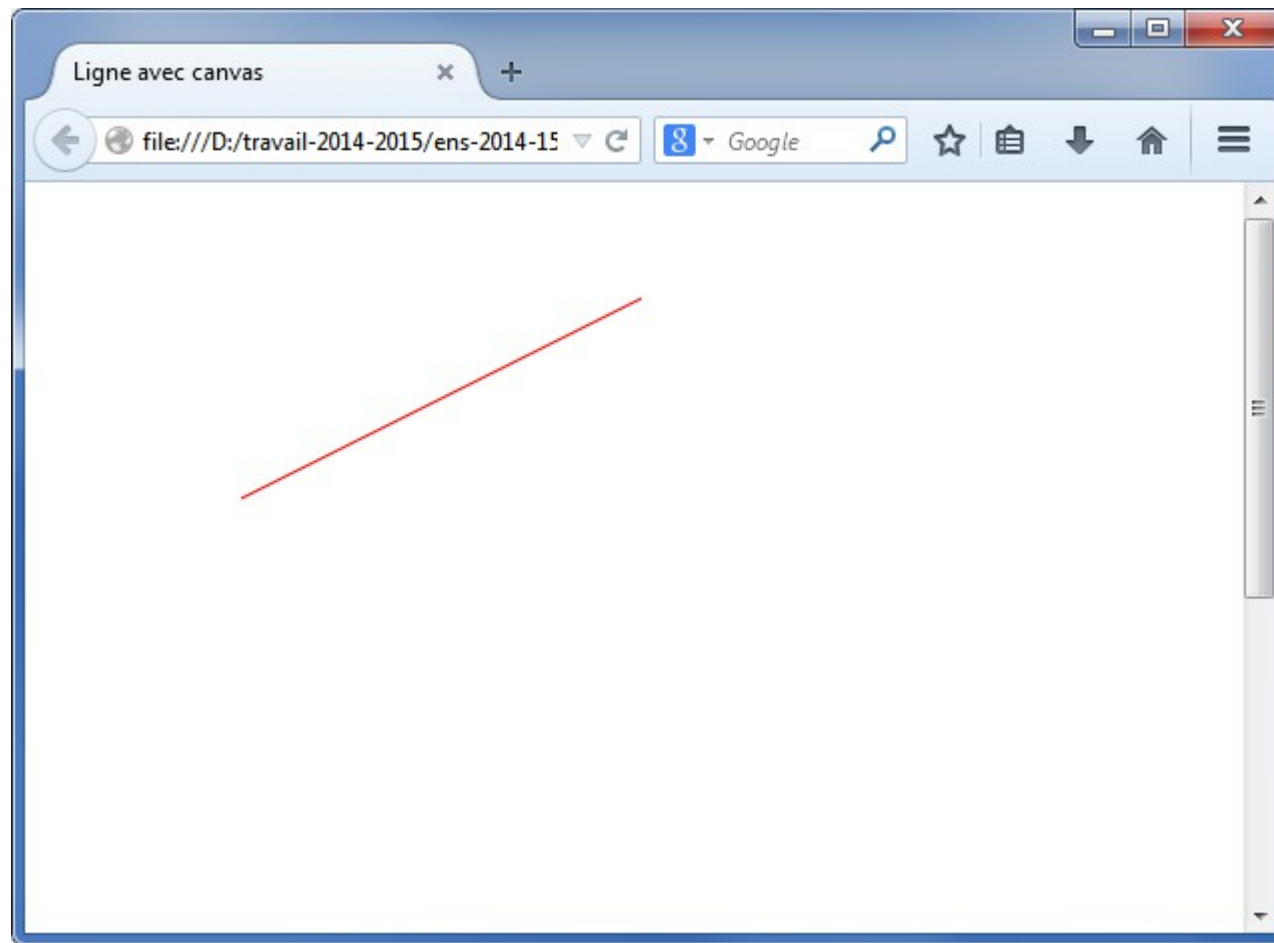
IV- Dessiner dans un <canvas> ?

Ligne dans un <canvas>

```
<!DOCTYPE html>
<html lang="fr">
  <head>
    <meta charset="utf-8" />
    <title>Ligne avec canvas</title>
    <link rel="stylesheet" href="style/monstyle.css" />
  </head>
  <body>
    <canvas id="monCanvas" width="600" height="600"></canvas>

    <script type="text/javascript">
      var canvas = document.getElementById('monCanvas');
      var ctx = canvas.getContext('2d');
      ctx.beginPath();
      ctx.moveTo(100, 150);
      ctx.lineTo(300, 50);
      ctx.strokeStyle = '#ff0000';
      ctx.stroke();
    </script>
  </body>
</html>
```

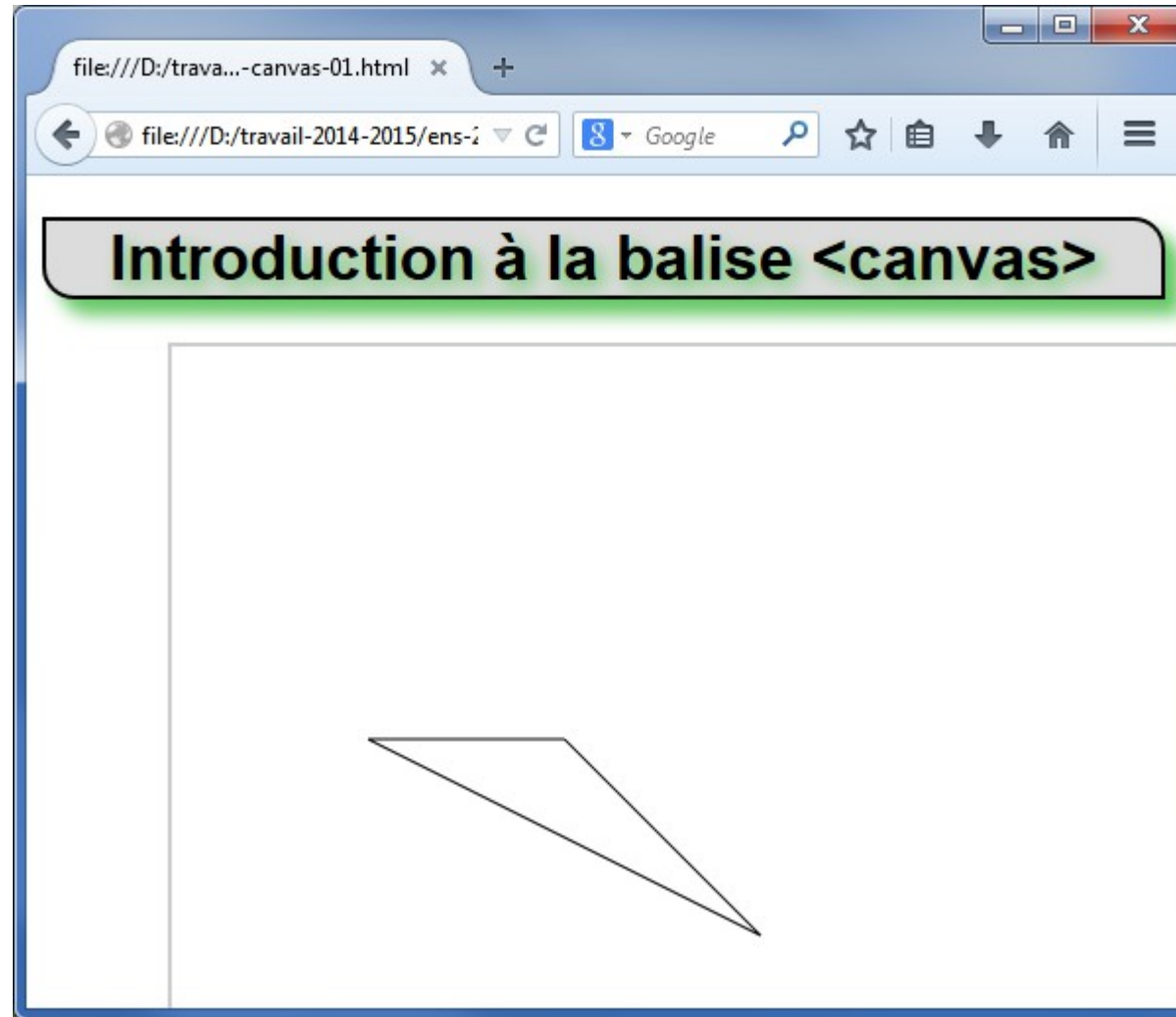

Ce qui donne :



Lignes dans un <canvas>

```
<!DOCTYPE html>
<html lang="fr">
  <head>
    <meta charset="utf-8" />
    <title>Ligne avec canvas</title>
    <link rel="stylesheet" href="style/monstyle.css" />
  </head>
  <body>
    <canvas id="monCanvas" width="600" height="600"></canvas>
    <script type="text/javascript">
      var canvas = document.getElementById('monCanvas');
      var ctx = canvas.getContext('2d');
      ctx.beginPath();
      ctx.moveTo(200,200);
      ctx.lineTo(300,300);
      ctx.moveTo(300,300);
      ctx.lineTo(100,200);
      ctx.moveTo(100,200);
      ctx.lineTo(200,200);
      ctx.closePath();
      ctx.stroke();
    </script>
  </body>
</html>
```

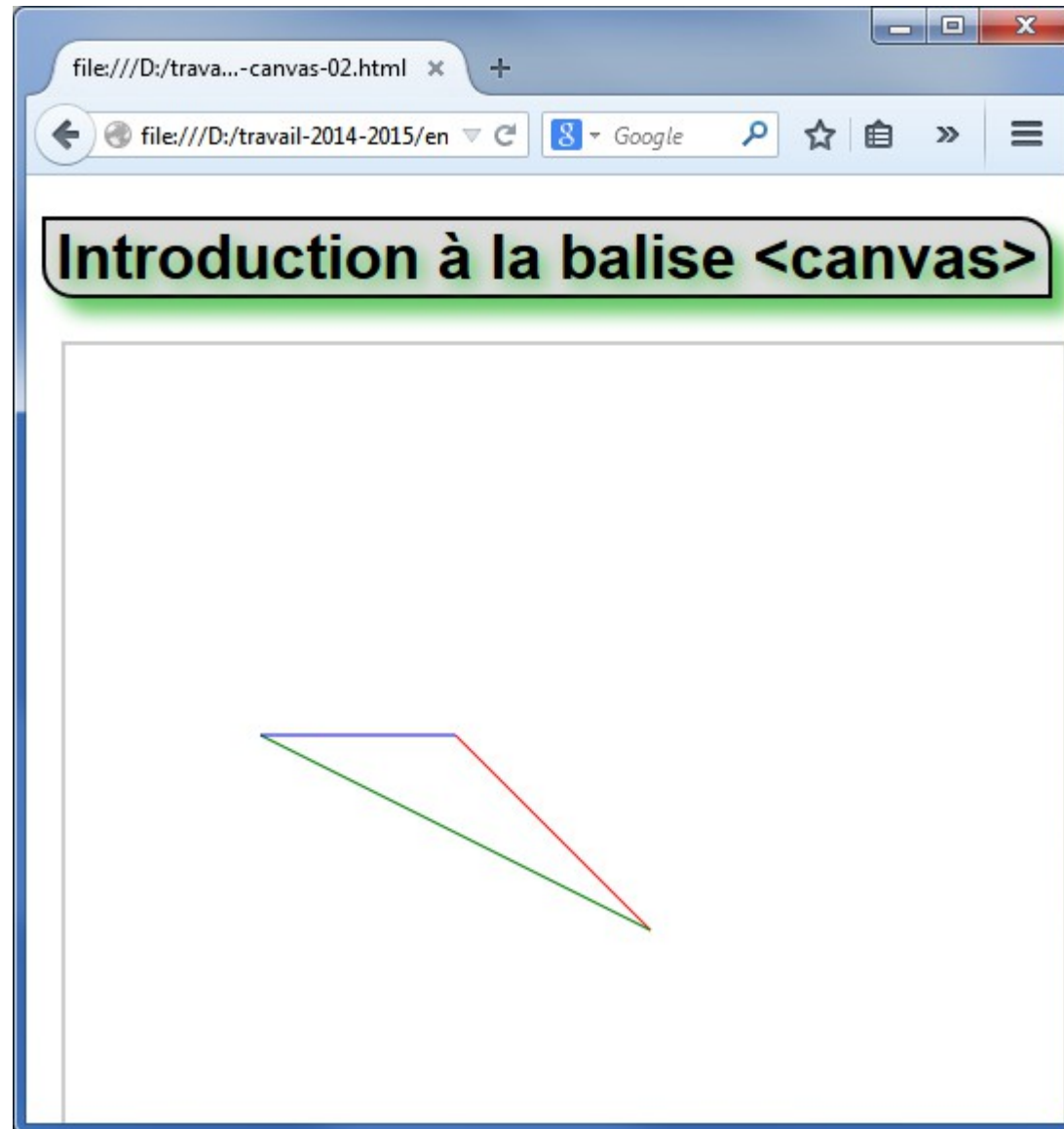
Ce qui donne :



Lignes colorées dans un <canvas>

```
<!DOCTYPE html>
<html lang="fr">
  <head>
    <meta charset="utf-8" />
    <title>Ligne avec canvas</title>
    <link rel="stylesheet" href="style/monstyle.css" />
  </head>
  <body>
    <canvas id="monCanvas" width="600" height="600"></canvas>
    <script type="text/javascript">
      var canvas = document.getElementById('monCanvas');
      var ctx = canvas.getContext('2d');
      ctx.beginPath(); ctx.strokeStyle="red";
      ctx.moveTo(200,200); ctx.lineTo(300,300);
      ctx.closePath(); ctx.stroke();
      ctx.beginPath(); ctx.strokeStyle="green";
      ctx.moveTo(300,300); ctx.lineTo(100,200);
      ctx.stroke(); ctx.closePath();
      ctx.beginPath(); ctx.strokeStyle="blue";
      ctx.moveTo(100,200); ctx.lineTo(200,200);
      ctx.closePath(); ctx.stroke();
    </script>
  </body>
</html>
```

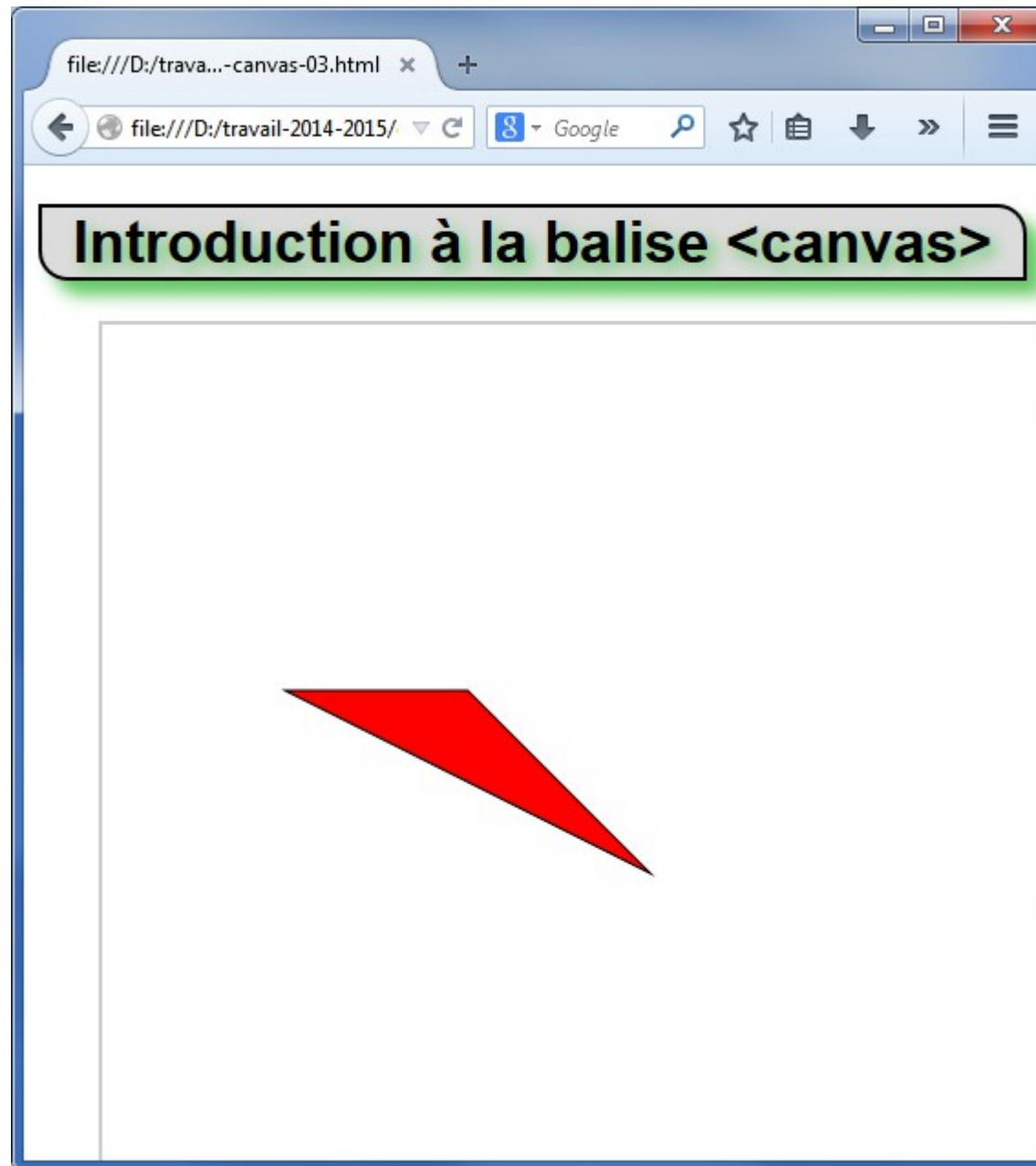
Ce qui donne :



Triangle plein dans un <canvas>

```
<!DOCTYPE html>
<html lang="fr">
  <head>
    <meta charset="utf-8" />
    <title>Ligne avec canvas</title>
    <link rel="stylesheet" href="style/monstyle.css" />
  </head>
  <body>
    <canvas id="monCanvas" width="600" height="600"></canvas>
    <script type="text/javascript">
      var canvas = document.getElementById('monCanvas');
      var ctx = canvas.getContext('2d');
      ctx.beginPath();
      ctx.fillStyle="red";
      ctx.moveTo(200,200);
      ctx.lineTo(300,300);
      ctx.lineTo(100,200);
      ctx.lineTo(200,200);
      ctx.closePath();
      ctx.fill();
      ctx.stroke();
    </script>
  </body>
</html>
```

Ce qui donne :

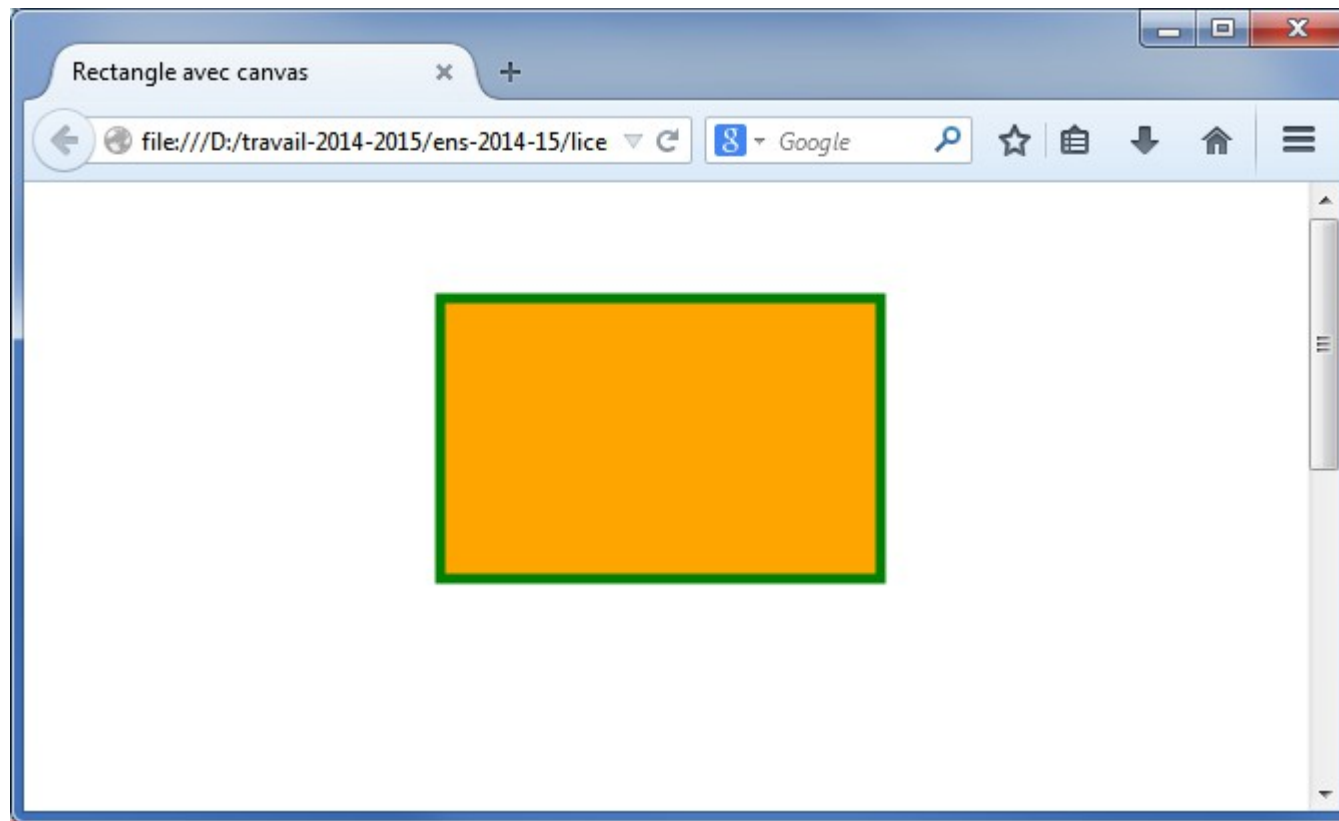


Rectangle dans un <canvas>

```
<!DOCTYPE html>
<html lang="fr">
  <head>
    <meta charset="utf-8" />
    <title>Rectangle avec canvas</title>
    <link rel="stylesheet" href="style/monstyle.css" />
  </head>
  <body>
    <canvas id="monCanvas" width="600" height="600"></canvas>

    <script type="text/javascript">
      var canvas = document.getElementById('monCanvas');
      var ctx = canvas.getContext('2d');
      ctx.beginPath();
      ctx.rect(200, 50, 220, 140);
      ctx.fillStyle = 'orange';
      ctx.fill();
      ctx.lineWidth = 5;
      ctx.strokeStyle = 'green';
      ctx.stroke();
    </script>
  </body>
</html>
```


Ce qui donne :

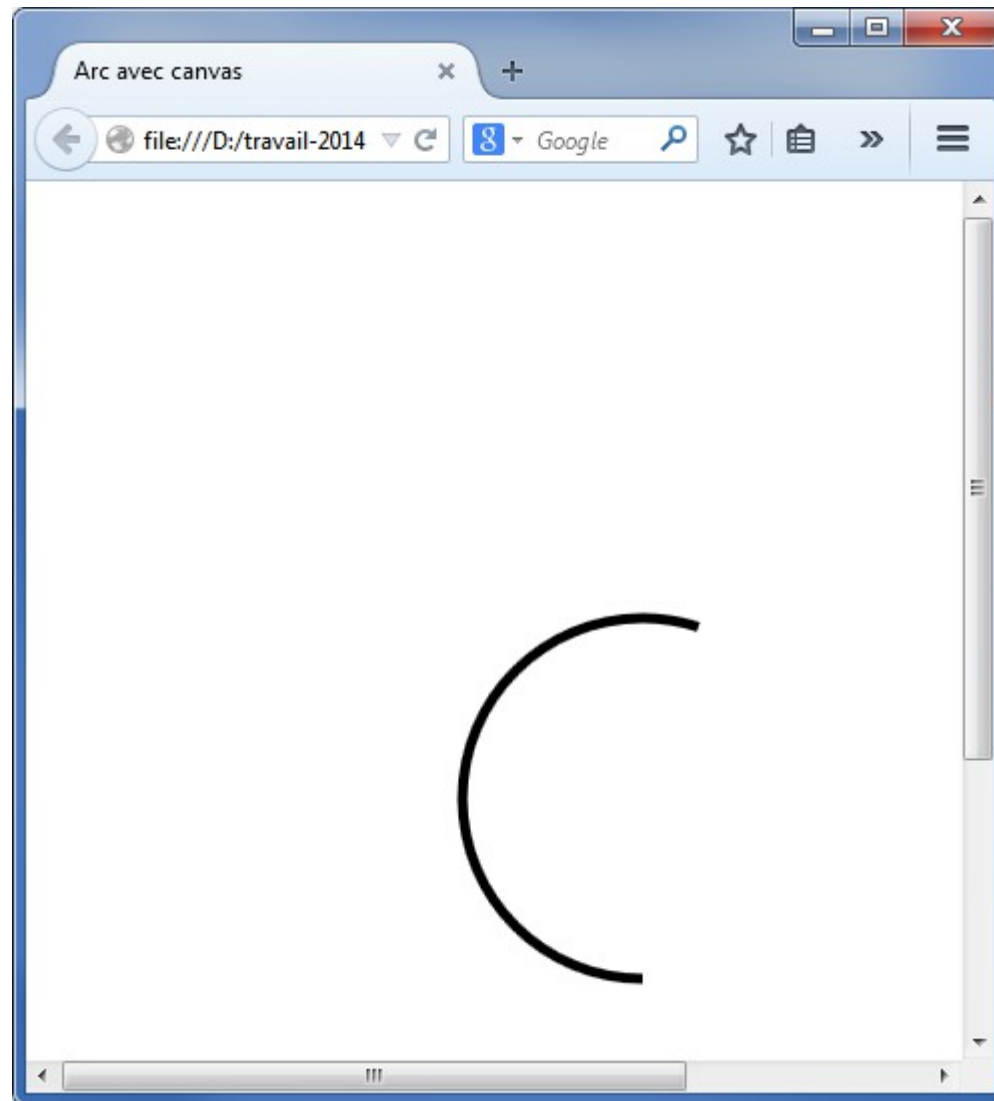


Arc dans un <canvas>

```
<!DOCTYPE html>
<html lang="fr">
  <head>
    <meta charset="utf-8" />
    <title>Arc avec canvas</title>
    <link rel="stylesheet" href="style/monstyle.css" />
  </head>
  <body>
    <canvas id="monCanvas" width="600" height="600"></canvas>

    <script type="text/javascript">
      var canvas = document.getElementById('monCanvas');
      var ctx = canvas.getContext('2d');
      var x = canvas.width / 2; var y = canvas.height / 2;
      var rayon = 90; var sensTrigo = false;
      var angleDebut = 0.5 * Math.PI; var angleFin = 1.6 *
Math.PI;
      ctx.beginPath();
      ctx.arc(x, y, rayon, angleDebut, angleFin, sensTrigo);
      ctx.lineWidth = 5;
      ctx.stroke();
    </script>
  </body>
</html>
```

Ce qui donne :



Exercices

- Ecrire un programme qui affiche 100 cercles colorés au hasard sur tout le <canvas>
- Ecrire un programme qui affiche 100 rectangles colorés au hasard sur tout le <canvas>
- Ecrire un programme qui affiche des bulles à gauche du <canvas> avec une taille de plus en plus petite du bas vers le haut

```
Math.round(Math.random()*255)  
Math.sin(angle)
```

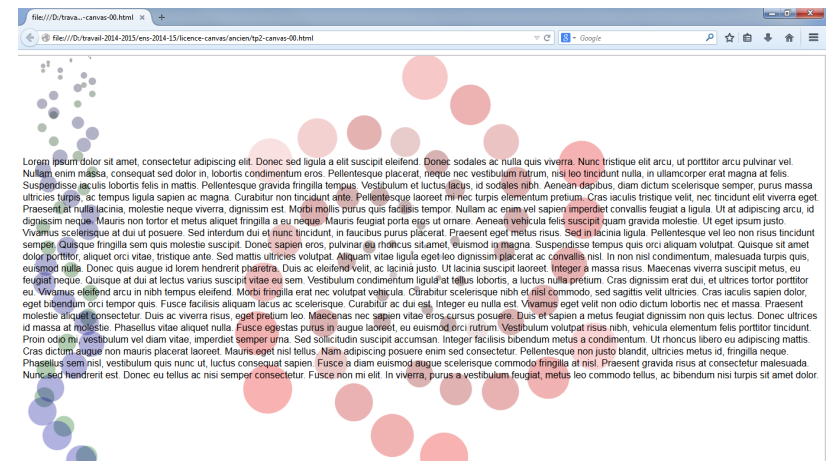
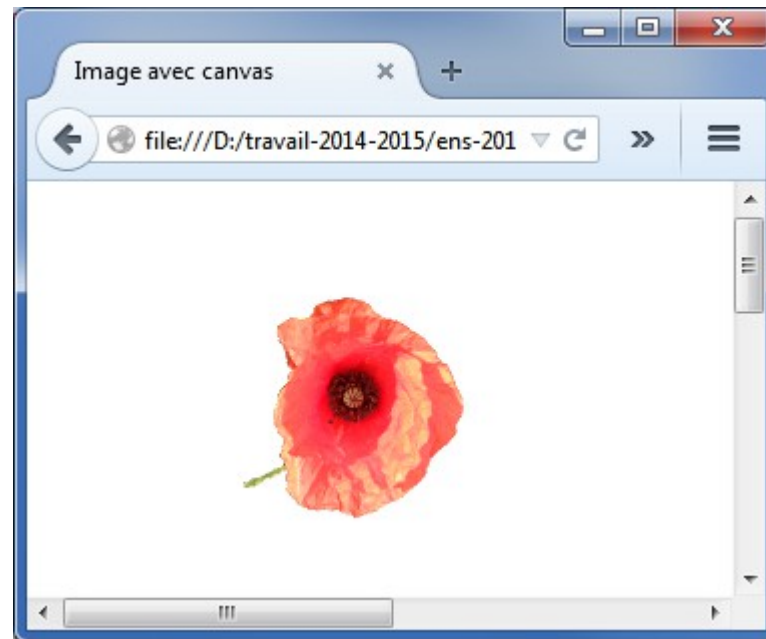


Image dans un <canvas>

- Récupérer :
 - <http://195.83.128.55/~jlandre/mim/coquelicot-02.png>

```
<!DOCTYPE html>
<html lang="fr">
  <head>
    <meta charset="utf-8" />
    <title>Image avec canvas</title>
    <link rel="stylesheet" href="style/monstyle.css" />
  </head>
  <body>
    <canvas id="monCanvas" width="600" height="600"></canvas>
    <script type="text/javascript">
      var canvas = document.getElementById('monCanvas');
      var ctx = canvas.getContext('2d');
      var img = new Image();
      img.src = "coquelicot-02.png";
      img.onload= function() {
        ctx.drawImage(img, 100, 50);
      }
    </script>
  </body>
</html>
```

Ce qui donne :



Images dans un <canvas>

- Ecrire un programme javascript qui affiche une diagonale d'une dizaine d'images de coquelicot du coin en haut à gauche vers le coin en bas à droite
- Ecrire un programme javascript qui affiche un cercle d'une dizaine de coquelicots
- Ecrire un programme javascript qui affiche une lemniscate de Bernouilli composée d'une dizaine de coquelicots

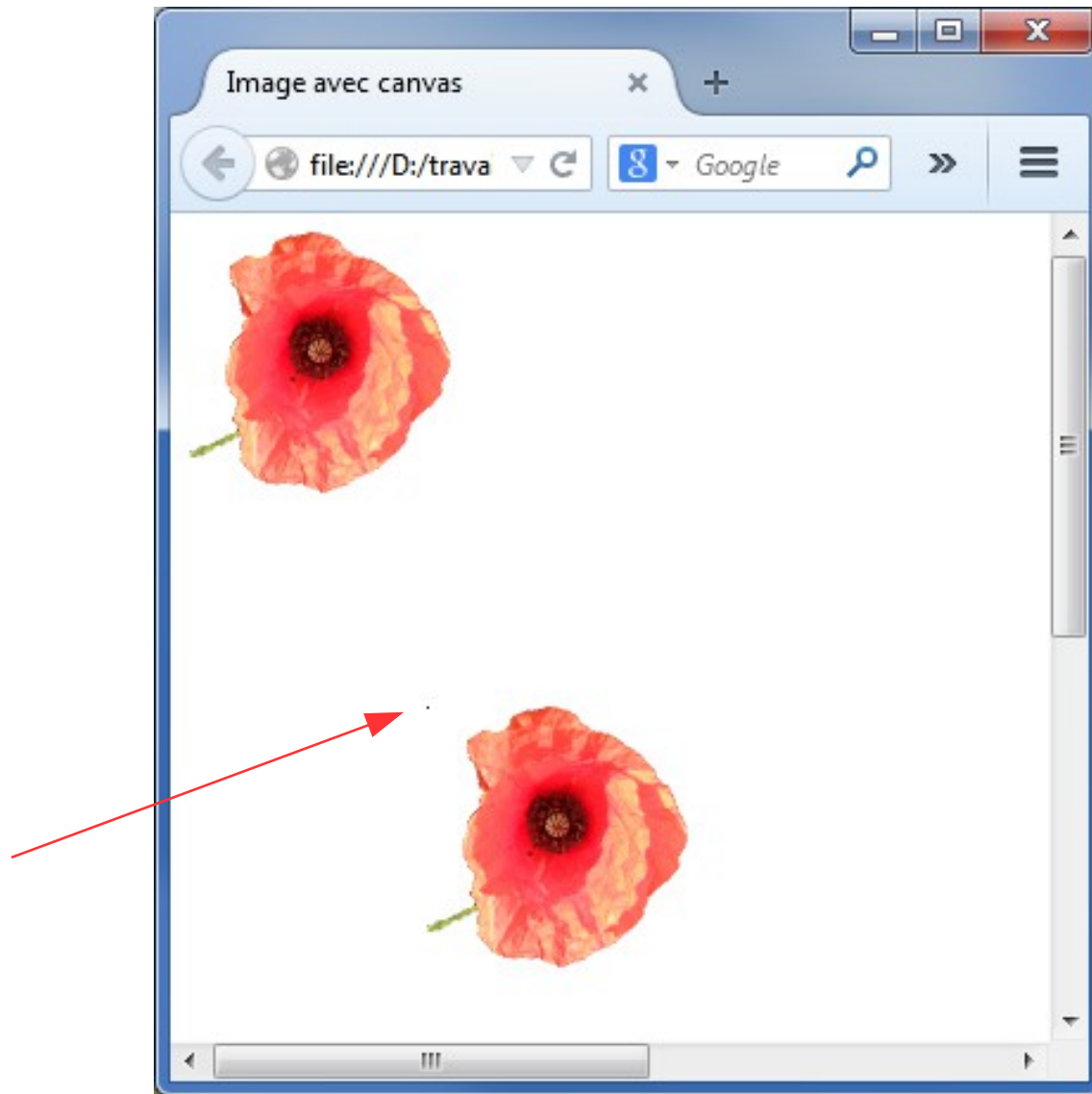


V- Traitement d'images avec canvas ?

Accès aux pixels dans un <canvas>

```
<!DOCTYPE html>
<html lang="fr">
  <head>
    <meta charset="utf-8" /> <title>Image avec canvas</title>
  </head>
  <body>
    <canvas id="monCanvas" width="600" height="600"></canvas>
    <script type="text/javascript">
      var canvas = document.getElementById('monCanvas');
      var ctx = canvas.getContext('2d');
      var img = new Image(); img.src = "coquelicot-02.png";
      img.onload = function() {
        ctx.drawImage(img, 0, 0);
        var pixels = ctx.getImageData(0, 0, img.width,
img.height);
        pixels.data[0] = 0; // r
        pixels.data[1] = 0; // v
        pixels.data[2] = 0; // b
        pixels.data[3] = 255; // alpha
        ctx.putImageData(pixels, 200, 300);
      }
    </script>
  </body>
</html>
```

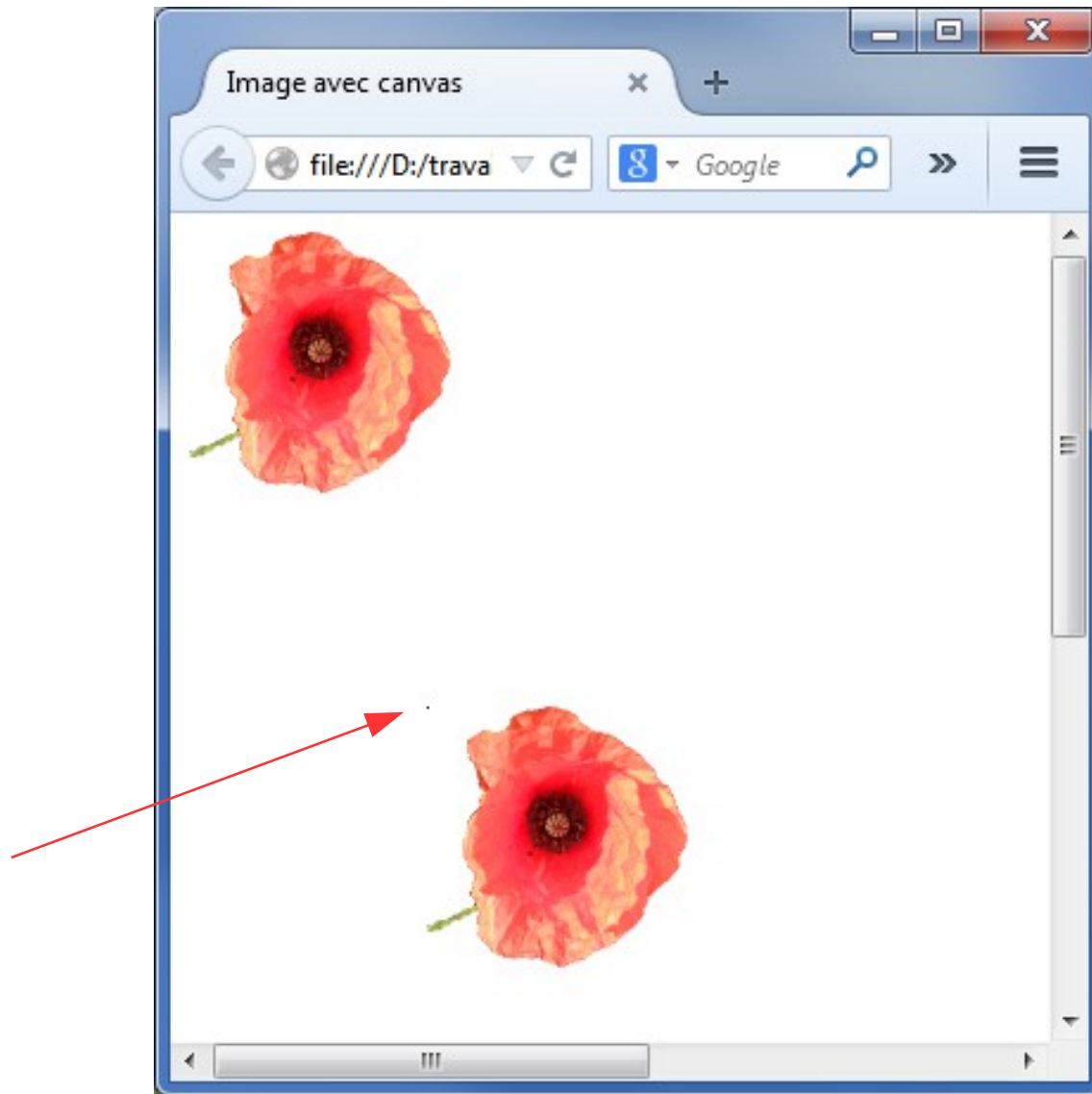
Ce qui donne :



Accès aux pixels dans un <canvas>

```
<!DOCTYPE html>
<html lang="fr">
  <head>
    <meta charset="utf-8" /> <title>Image avec canvas</title>
  </head>
  <body>
    <canvas id="monCanvas" width="600" height="600"></canvas>
    <script type="text/javascript">
      var canvas = document.getElementById('monCanvas');
      var ctx = canvas.getContext('2d');
      var img = new Image(); img.src = "coquelicot-02.png";
      img.onload = function() {
        ctx.drawImage(img, 0, 0);
        var pixels = ctx.getImageData(0, 0, img.width,
img.height);
        pixels.data[0] = 0; // r
        pixels.data[1] = 0; // v
        pixels.data[2] = 0; // b
        pixels.data[3] = 255; // alpha
        ctx.putImageData(pixels, 200, 300);
      }
    </script>
  </body>
</html>
```

Ce qui donne :

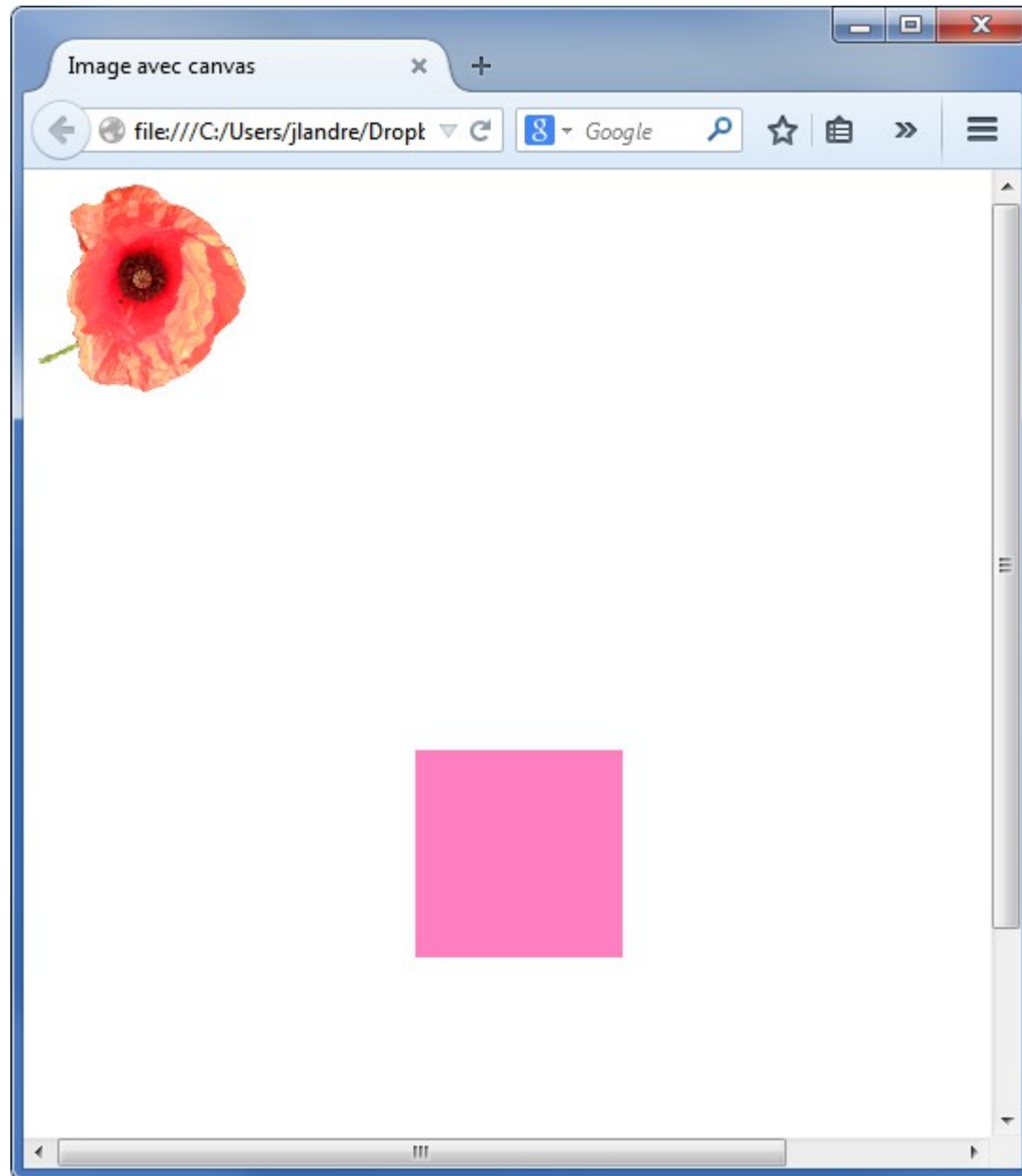


Parcours des pixels d'un <canvas>

...

```
<canvas id="monCanvas" width="600" height="600"></canvas>
<script type="text/javascript">
  var canvas = document.getElementById('monCanvas');
  var ctx = canvas.getContext('2d');
  var img = new Image(); img.src = "coquelicot-02.png";
  img.onload = function() {
    ctx.drawImage(img, 0, 0);
    var pixels = ctx.getImageData(0, 0, img.width,
img.height);
    var numBytes = pixels.data.length;
    for (var i=0; i<numBytes; i+=4) {
      pixels.data[i] = 255; // r
      pixels.data[i+1] = 0; // v
      pixels.data[i+2] = 128; // b
      pixels.data[i+3] = 255; // alpha
    }
    ctx.putImageData(pixels, 200, 300);
  }
</script>
</body>
</html>
```

Ce qui donne :





Avez-vous des questions ?