

Sujet du TP de prise en main de l'API Canvas

Objectif : se familiariser avec la balise <canvas> et l'API de dessin Javascript

1. Exercice 1

Créer un nouveau document HTML, dans lequel vous insérerez un canvas de 500px de large et de 500px de haut.

```
<!DOCTYPE html>
<html>
  <head>
    <title>Mon projet canvas</title>
    <meta charset="UTF-8" />
  </head>
  <body>
    <h1>L'API de dessin</h2>
    <canvas id="mon_canvas" width="500" height="500">
      Message pour les navigateurs ne supportant pas encore canvas.
    </canvas>
  </body>
</html>
```

- ✓ Que se passe-t-il si on enlève les attributs width et height à l'élément d'id "mon_canvas" ?
- ✓ Créer un nouveau fichier « canvas.js », et le lier au fichier HTML

Dans le fichier canvas.js

- ✓ Au chargement de la page, exécuter une action init() qui va initialiser le canvas.
 - Cette fonction init() va récupérer l'objet canvas, puis, dessiner le contexte 2d à l'aide de la méthode getContext(). Afficher un message d'erreur si le contexte ne peut pas être récupéré.
 - Faites afficher dans la console le résultat retourné par *object.getContext('2d')* et regardez ses éléments.

Dessiner

- ✓ Faites afficher un rond bleu de 10 pixels de rayon, en haut, à gauche du canvas.
- ✓ Faites afficher un rond jaune de 5 pixels de rayon, à une position choisie aléatoirement dans le canvas.
- ✓ Faites afficher 50 cercles verts de 2 pixels de rayon, à des positions aléatoires.

Animer

- ✓ Faites bouger le rond bleu : toutes les secondes, le rond descend de 5 pixels sur la diagonale. Rappel : la méthode *window.setInterval(traiter, nb)* permet d'effectuer la fonction *traiter()* toutes les nb millisecondes. Laisser les ronds bleus précédents.
- ✓ Arrêtez le déplacement quand le rond est arrivé dans le coin en bas, à droite
- ✓ Modifier le code précédent pour faire avancer le rond bleu, mais en effaçant au fur et à mesure le rond bleu précédent. Pour cela, placer les positions aléatoires des cercles verts dans un tableau a deux dimensions. Rappel : pour effacer *clearRect(positionX, positionY, largeur, hauteur)*;

Exercice 2 : Un bateau et des vagues

Créez un nouveau fichier HTML, dans lequel vous insérerez un canvas de 500px de haut et 500px de large.

Création d'un bateau

Dessin du mât

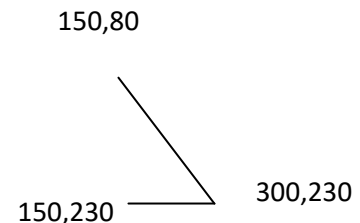
Dessiner un trait, de couleur « sienna ».

Position X = 140, Position Y=50, longueur=200, épaisseur du trait=10

Création de la voile :

Dessiner une forme pleine, de couleur lighthblue.

Fermer le tracé.

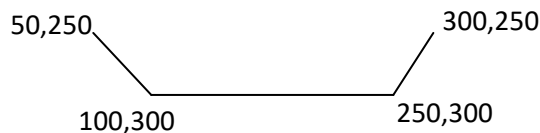


Dessin de la coque :

Il s'agit de dessiner cette forme :

Couleur de fond : #CD 85 3F

Couleur de trait : #A0 52 2D



Dessin des vagues

Par défaut, les arcs de cercle se dessinent dans le sens des aiguilles d'une montre, les angles sont définis en radians.

Les vagues sont en creux.

- ✓ Ecrire une fonction qui dessine une ligne de vagues(demi arc de cercle de 50px de diamètre).
- ✓ Ecrire une ligne de vagues de couleur #00bfff, et une autre de couleur #1e90ff, décalée de 50px.
- ✓ Ecrire un texte en bleu « bateau sur l'eau » avec une ombre grise #eee, décalage horizontal 1px, vertical 2px et flou de 3px.
- ✓ Ajouter en haut à gauche, un soleil en dégradé de jaune 45,45 et rayon 10, pour le cercle central, et 50,50,40 pour le second.



Exercice 3 :

- ✓ Créez un nouveau document avec un canvas dans lequel un rond s'affiche toutes les secondes à une position aléatoire. Quand l'utilisateur clique sur le canvas, s'il touche le rond, ce dernier s'efface.
- ✓ Ajoutez un bouton « compteur » qui s'incrémente à chaque rond touché.
- ✓ Ajouter des boutons de démarrage et de pause

Exercice 4 :

- ✓ Reprendre l'exercice 2 et animer les vagues.