
Fiche Guidée Pour La Réalisation Du Jeu Osu

Préliminaires

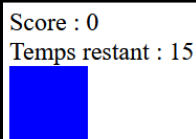
Assurez-vous d'avoir le fichier osu.html semblable à celui-ci :

```
<!DOCTYPE html>
<html>
<head>
  <title>Mini-jeux</title>
  <link type="text/css" rel="stylesheet" href="../style.css">
</head>
<body>
  <div id="monScore" class="score">Score : 0</div>
  <div id="monTimer" class="timer">Temps restant : 15</div>
  <div id="maBalle" class="balle"></div>
  <script src="osu.js"></script>
</body>
</html>
```

Dans votre fichier style.css vous devez avoir la classe balle écrite comme ci-dessous :

```
.balle{
  width: 50px;
  height: 50px;
  background-color: blue;
  position: absolute;
}
```

Si vous avez tout bien mis comme attendu, vous devriez avoir l'affichage suivant :



Score : 0
Temps restant : 15

A blue square representing the ball is visible below the text.

Modification de style.css

Pour modifier la classe balle dans style.css vous pouvez consulter le lien suivant : https://www.w3schools.com/cssref/css3_pr_border-radius.php

Exemple :



Modification de osu.js

1) `maBalle.addEventListener("click", function() {})`

Pour l'évènement « click » associé à votre variable maBalle, vous devez :

- ☐ Stocker dans une variable la largeur maximale de votre écran :
https://www.w3schools.com/jsref/prop_win_innerwidth.asp et
https://www.w3schools.com/jsref/prop_element_offsetwidth.asp
- ☐ Faire de même pour la hauteur maximale de votre écran
- ☐ Générer un x aléatoire entre 0 et votre largeur d'écran :
https://www.w3schools.com/js/js_random.asp
- ☐ Faire de même pour y
- ☐ Assigner les nouvelles coordonnées aléatoires à maBalle :
https://www.w3schools.com/jsref/prop_style_left.asp et
https://www.w3schools.com/jsref/prop_style_top.asp

Au début de votre programme (en dessous de `const maBalle`), vous devez :

- ☐ Ajouter variable `monScore` (`const`) comme `maBalle`
- ☐ Ajouter variable globale `score` (`var`).

Puis retournez dans votre évènement « click » pour :

- ☐ Incrémenter la variable `score` de 1.
- ☐ Changer le texte de `monScore` : https://www.w3schools.com/jsref/prop_node_textcontent.asp

Exemple :



2) `function updateTimer() {}`

- ☐ Ajoutez variable `monTimer` (const) comme `maBalle`.
- ☐ Ajoutez variable `tempsRestant` (var) comme score.
- ☐ Changer le texte de `monScore` : https://www.w3schools.com/jsref/prop_node_textcontent.asp

3) `function startGame() {}`

Dans `timerInterval = setInterval(function() {},` vous devez :

- ☐ Décrémenter de 1 le `tempsRestant`.
- ☐ Appeler votre fonction `updateTime`.
 - Si `tempsRestant` vaut zéro il faut alors :
 - Remettre à 1000 le timer : `clearInterval(timerInterval);`
 - Interpeller le joueur de son score :
https://www.w3schools.com/jsref/met_win_alert.asp
 - Appeler la fonction `resetGame`.

4) `function resetGame() {}`

- ☐ Remettre le timer à 1000 : `clearInterval(timerInterval);`
- ☐ Remettre `tempsRestant` à 15 et l'afficher
- ☐ Remettre le score à 0 et l'afficher
- ☐ Remplacer la balle à un endroit précis (au centre par exemple)
- ☐ Appeler la fonction `startGame`

Exemple :

Score : 7
Temps restant : 1

file://

Temps écoulé ! Votre score : 7

OK

Inspecteur Console Débugueur Réseau

Erreurs Avertissements Journaux Informations Débogage CSS XHR Requêtes

463.56439593218397	osu.js:60:11
362.2633295310795	osu.js:61:11
393.21663917918266	osu.js:60:11
379.63404958192046	osu.js:61:11
238.6352966585588	osu.js:60:11
380.8961223237514	osu.js:61:11
368.08099441929943	osu.js:60:11
304.4722369609026	osu.js:61:11
356.13231206601677	osu.js:60:11
445.61232915735934	osu.js:61:11
427.11896874985524	osu.js:60:11
485.6042966723165	osu.js:61:11
479.9433982521966	osu.js:60:11
304.5869353807195	osu.js:61:11