

```
<h1>Jeu du morpion</h1>
```

```
<div class="container">
```

```
  <div class="cell" data-index="0"></div>
```

```
  <div class="cell" data-index="1"></div>
```

```
  <div class="cell" data-index="2"></div>
```

```
  <div class="cell" data-index="3"></div>
```

```
  <div class="cell" data-index="4"></div>
```

```
  <div class="cell" data-index="5"></div>
```

```
  <div class="cell" data-index="6"></div>
```

```
  <div class="cell" data-index="7"></div>
```

```
  <div class="cell" data-index="8"></div>
```

```
</div>
```

```
<h2 class="info"></h2>
```

Permet de vérifier qui à gagné ou pas

Class qui permet de dire c'est a qui de joué et qui à gagné ou qui à perdu

```
*, ::before, ::after {  
  box-sizing: border-box;  
  margin: 0;  
  padding: 0;  
}
```

```
body {  
  font-family: Arial, Helvetica, sans-serif;  
  background: linear-gradient(to right, #283048, #859398);  
  color: #f1f1f1;
```

```
h1, h2 {  
  text-align: center;  
  font-size: 50px;  
  margin: 50px auto 0;  
}
```

```
.container {  
  display: grid;  
  grid-template: repeat(3,100px) / repeat(3,100px);  
  width: 300px;  
  height: 300px;  
  margin: 50px auto;  
}
```

```
.cell {  
  width: 100px;  
  height: 100px;  
  color: #f1f1f1;  
  border: 1px solid #f1f1f1;  
  cursor: pointer;  
  font-size: 60px;  
  display: flex;  
  justify-content: center;  
  align-items: center;  
}
```

Utilisation CSS GRID

1) 3 rangées de 100 px

2) 3 colonnes de 100 px

Une grille de 9 cases de 100 px de haut.

Width: 300px de large

Height: 300 px de haut

Margin 50px auto: centrer

Styliser les cellules

Cursor pointer: on peut cliquer dessus

```
const info = document.querySelector('.info');
```



Texte d'informations, qui à joué, qui à gagné etc...

```
const cellules = document.querySelectorAll('.cell');
```



On prend toutes les cellules

```
let verouillage = true;
```



Quand on a gagné on fige la partie

```
let joueurEnCours = "X";
```



C'est le joueur qui à la croix qui démarre

```
const alignementsGagnants = [  
  [0, 1, 2],  
  [3, 4, 5],  
  [6, 7, 8],  
  [0, 3, 6],  
  [1, 4, 7],  
  [2, 5, 8],  
  [0, 4, 8],  
  [2, 4, 6]  
]
```



Tableau des alignements gagnants

```
let partieEnCours = ["", "", "", "", "", "", "", "", ""];
```



Les endroits ou on aura coché

```
cellules.forEach(cell => {  
  cell.addEventListener('click', clicSurCase);  
})
```



Pour chaque cellule quand on va cliquer dessus, on va envoyer la fonction clique sur case

```
function clicSurCase(e){
```

```
  const caseClique = e.target;
```

```
  const caseIndex = caseClique.getAttribute('data-index');
```

```
  if(partieEnCours[caseIndex] !== "" || !verouillage){
```

```
    return;
```

```
}
```

```
  changementDeJoueur();
```

```
}
```

```
function changementDeJoueur(){
```

```
  joueurEnCours = joueurEnCours === "X" ? "O" : "X";
```

```
  info.innerText = `Au tour de ${joueurEnCours}`;
```

```
}
```

D'abord la fonction **Validation Résultats**

```
function validationResultats(){
```

```
  let finDePartie = false;
```

```
  for(let i = 0; i < alignementsGagnants.length; i++){
```

```
    const checkWin = alignementsGagnants[i];
```

```
    // [0, 1, 2],
```

```
    let a = partieEnCours[checkWin[0]];
```

```
    let b = partieEnCours[checkWin[1]];
```

```
    let c = partieEnCours[checkWin[2]];
```

```
    if(a === "" || b === "" || c === ""){
```

```
      continue;
```

```
    }
```

```
    if( a === b && b === c){
```

```
      finDePartie = true;
```

```
      break;
```

```
    }
```

```
}
```

(e):objet événement.

Case sur laquelle on va cliquer

Index de la case

Si case coché, verrouillage.

Si le joueur en cours est la
croix alors prochain
joueur devient O

**Elle sera appelé
uniquement si la partie
est finie ou pas**

Itérer à travers les 8 combinaisons possible

Première itération alignement de zéro,
Par exemple ca va checker si on a 3 X les uns à la suite des autres

A sera = à X ou O. La même chose pour B et C. On cherche si c'est égal 3 fois
à X ou 3 fois égal à O.

Permet la vérification

La c'est une victoire

```
if(finDePartie){  
  info.innerText = `Le joueur ${joueurEnCours} a gagné !`  
  verouillage = false;  
  return;  
}
```

Fin de partie, alors verrouillage

Et c'est la seulement que le changement de joueur intervient

```
changementDeJoueur();  
}  
  
function changementDeJoueur(){  
  joueurEnCours = joueurEnCours === "X" ? "O" : "X";  
  info.innerText = `Au tour de ${joueurEnCours}`;  
}
```

Si le joueur en cours est la croix alors prochain joueur devient O
Elle sera appelé uniquement si la partie est finie ou pas

```
// si il n'y pas de chaine de caracteres vides dans partie en cours  
let matchNul = !partieEnCours.includes('');  
if(matchNul){  
  info.innerText = 'Match nul !';  
  verouillage = false;  
  return;  
}
```

Vérifie que tout est rempli ou prend en compte qu'on ne plus rien remplir

Inclus les caractères de chaînes vides