**Ce code représente un jeu de tic-tac-toe en JavaScript. Voici un résumé et une explication technique du code :**

**Résumé**

**Le code implémente un jeu de tic-tac-toe où deux joueurs, "X" et "O", alternent pour remplir une grille de 3x3. Chaque cellule de la grille est cliquable et, une fois cliquée, elle affiche le symbole du joueur courant. Le jeu vérifie après chaque coup si un joueur a gagné ou si la partie est nulle, et il gère le changement de joueur.**

**Explication Technique**

1. **Initialisation du joueur courant :**

**javascript**

**Copy code**

**let currentPlayer = "X";**

1. **Mise à jour de l'affichage :**

**javascript**

**Copy code**

**const info = document.querySelector(".info") info.textContent = `Au tour de ${currentPlayer}`**

1. **Sélection des cellules du jeu :**

**javascript**

**Copy code**

**const cells = document.querySelectorAll(".cell")**

1. **Ajout d'un gestionnaire d'événements pour chaque cellule :**

**javascript**

**Copy code**

**cells.forEach(cell => cell.addEventListener("click", handleClick))**

1. **Initialisation de l'état du jeu et du verrou :**

**javascript**

**Copy code**

**const currentGame = ["","","","","","","","","",""]; let lock = false;**

1. **Gestionnaire d'événements handleClick :**

**javascript**

**Copy code**

**function handleClick(e) { const clickedBox = e.target; const boxIndex = clickedBox.getAttribute("data-index"); if(currentGame[boxIndex] !== "" || lock){ return; } currentGame[boxIndex] = currentPlayer; clickedBox.textContent = currentPlayer; verification(); }**

* + **Vérifie si la case est déjà occupée ou si le jeu est verrouillé.**
  + **Met à jour l'état de la case cliquée avec le symbole du joueur courant.**
  + **Appelle la fonction verification pour vérifier l'état du jeu.**

1. **Combinaisons gagnantes :**

**javascript**

**Copy code**

**const winningCombinations = [ [0, 1, 2], [3, 4, 5], [6, 7, 8], [0, 3, 6], [1, 4, 7], [2, 5, 8], [0, 4, 8], [2, 4, 6] ]**

1. **Fonction verification :**

**javascript**

**Copy code**

**function verification (){ for(let i = 0; i < winningCombinations.length; i++) { const combinationToCheck = winningCombinations[i]; let a = currentGame[combinationToCheck[0]] let b = currentGame[combinationToCheck[1]] let c = currentGame[combinationToCheck[2]] if(a === "" || b === "" || c === ""){ continue; } else if (a === b && b === c){ info.textContent = `Le joueur ${currentPlayer} a gagné ! Appuyez sur F5 pour recommencer.` lock = true; return; } } if (!currentGame.includes('')){ info.textContent = `Match Nul ! Appuyez sur F5 pour recommencer.` lock = true; return; } switchPlayer() }**

* + **Vérifie chaque combinaison gagnante.**
  + **Déclare un joueur gagnant si une combinaison est remplie par le même symbole.**
  + **Déclare un match nul si toutes les cases sont remplies sans gagnant.**
  + **Appelle switchPlayer si le jeu continue.**

1. **Changement de joueur :**

**javascript**

**Copy code**

**function switchPlayer(){ currentPlayer = currentPlayer === "X" ? "O" : "X"; info.textContent = `Au tour de ${currentPlayer}` }**

* + **Alterne le joueur courant entre "X" et "O".**
  + **Met à jour le message affiché pour indiquer le joueur courant.**

**Conclusion**

**Le code est structuré pour gérer un jeu de tic-tac-toe simple et efficace, en vérifiant les conditions de victoire et de match nul après chaque coup et en alternant entre les joueurs. Les cellules de la grille sont mises à jour dynamiquement en fonction des interactions des utilisateurs.**