Informationen zur Herangehensweise

Um aus dem Tik Tak To-Spiel in ein Vier Gewinnt-Spiel zu machen, müssten einige grundlegende Änderungen vorgenommen werden. Hier sind einige Vorschläge, wie du das anpassen könntest:

1. Ändere die Anzahl der Felder und die Anordnung der Gewinnmöglichkeiten entsprechend dem Vier Gewinnt-Spiel. Statt 3x3 Feldern benötigst du 7 Spalten und 6 Reihen für Vier Gewinnt.

2. Passe die Logik der Gewinnüberprüfung an, um nach Vier-in-einer-Reihe zu suchen, sei es horizontal, vertikal oder diagonal.

3. Aktualisiere die Funktionen wie "update\_feld", "gewinner\_ermitteln" usw., um die neuen Anforderungen des Vier Gewinnt-Spiels zu erfüllen.

4. Passe das HTML und CSS entsprechend an, um das Spielfeld und die Benutzeroberfläche für Vier Gewinnt anzupassen.

Diese Änderungen erfordern eine umfassendere Anpassung des Codes, um die Spiellogik und die visuelle Darstellung für das Vier Gewinnt-Spiel anzupassen.

Tabelle um Index schneller zu finden.

(0, 1, 2, 3, 4, 5, 6)

(7, 8, 9, 10, 11, 12, 13)

(14, 15, 16, 17, 18, 19, 20)

(21, 22, 23, 24, 25, 26, 27)

(28, 29, 30, 31, 32, 33, 34)

(35, 36, 37, 38, 39, 40, 41)

(42, 43, 44, 45, 46, 47, 48)

(49, 50, 51, 52, 53, 54, 55)

**Projektplanung:**

* **Allgemein:**
  + Überschrift sollte vorhanden sein
  + Punktesystem (wenn genug Zeit da ist)
  + Tabellarisch Spieler „X“ und „O“ auflisten und Punkte rechts daneben zuordnen
* Spielfeld erstellen (7x8 Felder)
* Am besten einen Status der Spielsituation
  + soll anzeigen wer dran ist,
  + wer Gewonnen hat,
  + wenn Unentschieden ist
* Einen Restart Button erstellen
  + Soll das game Aktualisieren, um eine neue Runde zu starten
  + Wenn möglich dann auch versuchen Punktesystem mit nicht einbinden, so dass die Punkte bleiben, aber das Spielfeld Aktualisiert wird.

**Im Detail:**

* HTML-Gerüst:
  + Ein großes Div mit der id=“game\_container“
  + Darin enthalten das h1
  + Dann 2 weitere divs am besten eins für das Punktesystem und eins für das Spielfeld
  + 1) Div id=“punkte\_tabelle“
  + 2) Div id=“felder\_container“
  + Außerhalb der 2 divs aber innerhalb des großen divs ein h2 id=“text\_status“
  + Darunter innerhalb des großen divs ein button id=“restart\_button“
* CSS: Design und Farben Random suchen und anpassen je nachdem was mir gefällt
* Javascript:
  + Erst alle Notwendigen Variablen erstellen und definieren
    - Z.B.: felder=document.querySelectorAll(„.(definierte class)“), um dadurch bspw alle Felder aufzurufen
    - Variable für den Status, wer dran ist, wer gewonnen hat usw.
    - Variable für den button, aus dem HTML Gerüst ziehen
  + Am besten Gewinnmöglichkeit als Array festlegen (sind nicht viele deswegen)
    - Variable für die Gewinnmöglichkeiten als Array [];
    - Variable für den Zustand des Spielfeldes, bspw.: dass es am Anfang leer sein muss
    - Variable für den Spieler, ihn zu definieren bspw.: als „X“
    - Variable bzw boolea dass das Spiel laufen soll oder sich erstmal auf „false“ setzen
  + Funktionen festlegen, am besten alle erst auflisten und dann erst die Logik coden, BSP.:
    - function spiel\_laden(){} usw
    - Vorhandene Funktionen sollten sein:
    - Game Initialisieren,
    - Klick-Funktion, dass „X“ || „O“ gesetzt werden können,
    - Spieler wechsel-Funktion,
    - Spieler Punkte
    - Gewinnermittlung-Funktion und zu guter Letzt die …
    - Aktualisierung des Buttons-Funktion
    - Ggf. neue feat. Hinzufügen zb.:

. sound für gamer „X“ && „O“

. sound für win

. sound für unentschieden

. button für sound „on/off“

. ggf. button als Png/ico darstellen

* + Darauf achten das ggf. einige Funktionen in andere wieder aufgerufen werden müssen