**Game Catch The Box**

* **INDEX.HTML**

<!DOCTYPE html>

<html lang="en">

<head>

  <meta charset="UTF-8" />

  <meta name="viewport" content="width=device-width, initial-scale=1.0" />

  <title>Catch the Box Game</title>

  <link rel="stylesheet" href="style.css" />

</head>

<body>

  <h1>Catch the Box!</h1>

  <p id="score">Score: 0</p>

  <div id="gameArea">

    <div id="box"></div>

  </div>

  <script src="game.js"></script>

</body>

</html>

* **STYLE.CSS**

<!DOCTYPE html>

<html lang="en">

<head>

  <meta charset="UTF-8" />

  <meta name="viewport" content="width=device-width, initial-scale=1.0" />

  <title>Catch the Box Game</title>

  <link rel="stylesheet" href="style.css" />

</head>

<body>

  <h1>Catch the Box!</h1>

  <p id="score">Score: 0</p>

  <div id="gameArea">

    <div id="box"></div>

  </div>

  <script src="game.js"></script>

</body>

</html>

* **GAME.JS**

const box = document.getElementById("box");

const scoreDisplay = document.getElementById("score");

let score = 0;

function getRandomPosition() {

  const gameArea = document.getElementById("gameArea");

  const maxX = gameArea.clientWidth - 50;

  const maxY = gameArea.clientHeight - 50;

  const x = Math.random() \* maxX;

  const y = Math.random() \* maxY;

  return { x, y };

}

function moveBox() {

  const { x, y } = getRandomPosition();

  box.style.left = `${x}px`;

  box.style.top = `${y}px`;

}

box.addEventListener("click", () => {

  score += 1;

  scoreDisplay.textContent = `Score: ${score}`;

  moveBox();

});

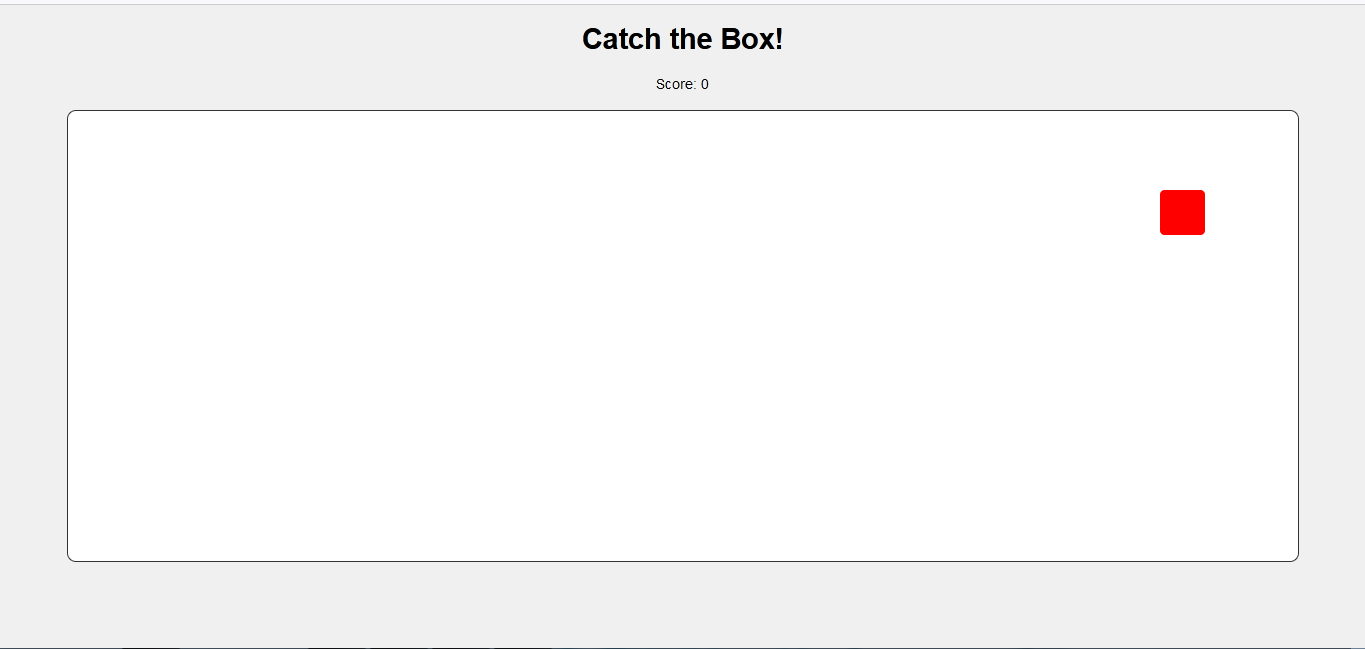
setInterval(() => {

  moveBox();

}, 1000); // move every 1 second

// Initial move

moveBox();

****

**Game Tebak Angka**

* **INDEX.HTML**

<!DOCTYPE html>

<html lang="id">

<head>

  <meta charset="UTF-8" />

  <meta name="viewport" content="width=device-width, initial-scale=1.0"/>

  <title>Game Tebak Angka</title>

  <link rel="stylesheet" href="style.css" />

</head>

<body>

  <div class="game-container">

    <h1>Tebak Angka 1 - 10</h1>

    <p>Masukkan angka tebakan kamu:</p>

    <input type="number" id="guessInput" min="1" max="10"/>

    <button onclick="checkGuess()">Tebak</button>

    <p id="message"></p>

  </div>

  <script src="script.js"></script>

</body>

</html>

* **STYLE.CSS**

body {

    font-family: Arial, sans-serif;

    background-color: #f4f4f4;

    display: flex;

    justify-content: center;

    align-items: center;

    height: 100vh;

    margin: 0;

  }

  .game-container {

    background-color: white;

    padding: 30px;

    border-radius: 12px;

    box-shadow: 0 0 15px rgba(0,0,0,0.2);

    text-align: center;

  }

  input {

    padding: 8px;

    width: 60px;

    text-align: center;

  }

  button {

    padding: 8px 12px;

    margin-left: 10px;

    cursor: pointer;

  }

* **Script.Js**

// Angka acak dari 1 sampai 10

let randomNumber = Math.floor(Math.random() \* 10) + 1;

function checkGuess() {

  const guess = parseInt(document.getElementById('guessInput').value);

  const message = document.getElementById('message');

  if (isNaN(guess) || guess < 1 || guess > 10) {

    message.textContent = "Tolong masukkan angka antara 1 dan 10.";

    return;

  }

  if (guess === randomNumber) {

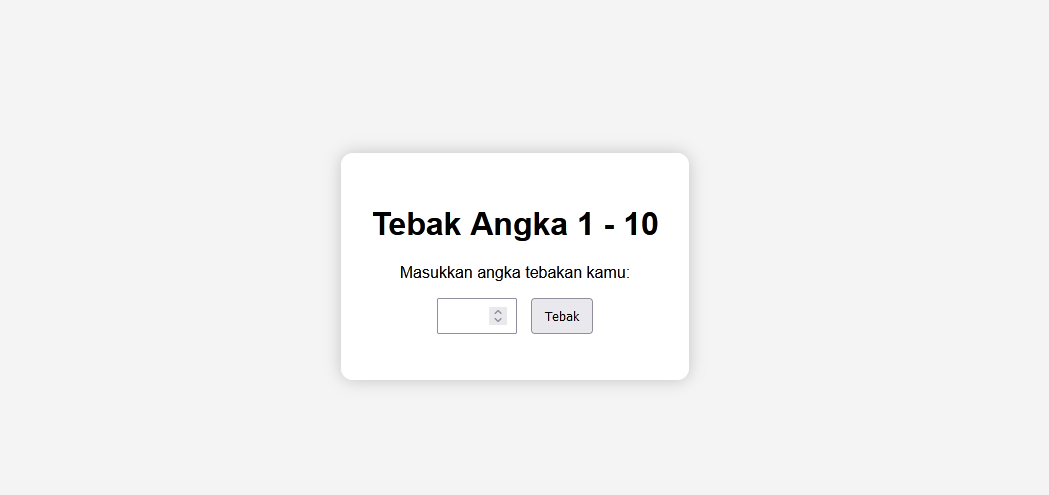
    message.textContent = "🎉 Selamat! Tebakan kamu benar.";

  } else {

    message.textContent = "❌ Salah. Coba lagi!";

  }

}



**CATUR**

* **INDEX.HTML**

<!DOCTYPE html>

<html lang="en">

<head>

  <meta charset="UTF-8" />

  <meta name="viewport" content="width=device-width, initial-scale=1.0"/>

  <title>Chess Game</title>

  <link rel="stylesheet" href="style.css" />

</head>

<body>

  <h1>Simple Chess Game</h1>

  <div id="chessboard"></div>

  <script src="catur.js"></script>

</body>

</html>

* **STYLE CSS**

body {

    background: #2e2e2e;

    font-family: sans-serif;

    color: white;

    text-align: center;

    margin: 0;

    padding: 0;

  }

  h1 {

    margin: 20px 0;

  }

  #chessboard {

    display: grid;

    grid-template-columns: repeat(8, 60px);

    grid-template-rows: repeat(8, 60px);

    width: 480px;

    margin: auto;

    border: 4px solid #444;

  }

  .square {

    display: flex;

    justify-content: center;

    align-items: center;

    font-size: 32px;

    cursor: pointer;

    user-select: none;

  }

  .light {

    background: #f0d9b5;

  }

  .dark {

    background: #b58863;

  }

  .piece:hover {

    transform: scale(1.1);

  }

* CATUR JS

const board = document.getElementById("chessboard");

const pieces = {

  r: "♜", n: "♞", b: "♝", q: "♛", k: "♚", p: "♟",

  R: "♖", N: "♘", B: "♗", Q: "♕", K: "♔", P: "♙",

};

const initialBoard = [

  ["r","n","b","q","k","b","n","r"],

  ["p","p","p","p","p","p","p","p"],

  ["","","","","","","",""],

  ["","","","","","","",""],

  ["","","","","","","",""],

  ["","","","","","","",""],

  ["P","P","P","P","P","P","P","P"],

  ["R","N","B","Q","K","B","N","R"],

];

let selected = null;

function drawBoard() {

  board.innerHTML = "";

  for (let row = 0; row < 8; row++) {

    for (let col = 0; col < 8; col++) {

      const square = document.createElement("div");

      square.className = "square " + ((row + col) % 2 === 0 ? "light" : "dark");

      square.dataset.row = row;

      square.dataset.col = col;

      const piece = initialBoard[row][col];

      if (piece) {

        square.textContent = pieces[piece];

        square.classList.add("piece");

      }

      square.addEventListener("click", () => handleClick(row, col));

      board.appendChild(square);

    }

  }

}

function handleClick(row, col) {

  if (selected) {

    const [selRow, selCol] = selected;

    initialBoard[row][col] = initialBoard[selRow][selCol];

    initialBoard[selRow][selCol] = "";

    selected = null;

    drawBoard();

  } else {

    if (initialBoard[row][col] !== "") {

      selected = [row, col];

    }

  }

}

drawBoard();

