



15puzzle

Nick Clark, Derek Malikian,
Anthony Thompson

15puzzle

Goal: In this project we attempt to make a 15puzzle game using HTML,CSS, and JAVASCRIPT.

Extra Features

- Move counter
- Timer
- Multiple backgrounds
- Tile animations

The Fifteen Puzzle

Which image would you like to use?



The Fifteen Puzzle



Shuffle

Reset

Moves: 0

THE FIFTEEN PUZZLE

10	12	4	13
1		2	7
5	11	14	3
6	15	8	9

Shuffle

Reset

Moves: 0
Time: 8

Basic Functions;

The timer() increments giving us the time we have been playing. We have a resetTimer() and reset() .

```
<div class = "place_content">
  <p id="shuffle">
    <button id="shuffle_btn" onclick=
      ="shufflePuzzle()">Shuffle</
    button>
    <button onclick="reset()" value=
      "Reset">Reset</button>
  </p>

  <div class = "moves_time">Moves: <span
    id="counter">0</span>
  <br/>
    Time: <span id="timer">0</span>

</div>
```

```
function timer() {
  timer = setInterval(function() {
    seconds++;
    document.getElementById("timer").innerText = seconds;
  }, 1000);
}
//Function to restart the timer
function resetTimer() {
  seconds = 0;
  timer();
}
//Function to completely reload game
function reset() {
  location.reload();
}
```


Load Elements

This is to set the puzzle [] to the 15 divs and make them puzzle pieces

```
window.onload = function() {  
    puzzle = $("#gameboard div"); //Set puzzle[] to the div's inside of the gameboard  
    var row = 0;  
    var right = 0;  
    var top = 0;  
    for (var i = 0; i < puzzle.length; i++) {  
        puzzle[i].addClassName("puzzlepiece"); //mark divs as puzzlepiece class  
        puzzle[i].style.float = "left";  
        puzzle[i].style.backgroundColor = "400px 400px";  
  
        blank[i] = []; //Fill blank[] with empty []'s  
        blank[i][0] = right;  
        blank[i][1] = top;  
  
        puzzle[i].style.backgroundColor = "-" + blank[i][0] + "px -" + blank[i][1] + "px";  
        row++;  
        if (row == 4) { //Track which row is being displayed  
            top += 100;  
            right = 0;  
            row = 0;  
        } else {  
            right += 100;  
        }  
    }  
}
```

Shuffle()

This function allows us to shuffle the piece about the board.

```
function shufflePuzzle() {  
    var numArray = [0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15];  
    for (var i = puzzle.length; i > 0; i) {  
        var j = Math.floor(Math.random() * i);  
        var x = numArray[--i];  
        var test = numArray[j];  
        if (test == "0") {  
            puzzle[i].addClassName("puzzlepiece");  
            blankP(puzzle[i]);  
            puzzle[i].innerHTML = "";  
        } else {  
            puzzle[i].innerHTML = numArray[j];  
            regularP(puzzle[i]);  
            background_Position(puzzle[i], test);  
        }  
        numArray[j] = x;  
    }  
    moves = 0; //Reset moves after the board is shuffled  
    document.getElementById("counter").innerHTML = (moves); //Print the current moves  
    mopiece();  
    resetTimer(); //Restart the timer that was running for the previous shuffle  
}
```

Movepiece

This checks to see whether the piece is movable or not and allows us to move it if it is movable.

```
var movepiece = function() {  
  var move = this.innerHTML;  
  var can_move = this.hasClassName('movablepiece'); //boo  
  var blank = 0;  
  if (can_move) { //if yes, move piece  
    for (var i = 0; i < puzzle.length; i++) {  
      blank = puzzle[i].innerHTML;  
      if (puzzle[i].innerHTML == "") {  
        puzzle[i].innerHTML = move;  
        this.innerHTML = blank;  
        regularP(puzzle[i]);  
        blankP(this);  
        mopiece();  
        background_Position(puzzle[i], move);  
      }  
    }  
  }  
  moves++; //increment moves  
  document.getElementById("counter").innerHTML = (moves);  
};
```

MOPIECE

The function allows us to find and label the game pieces.

```
var mopiece = function() {  
  for (var i = 0; i < puzzle.length; i++) {  
    puzzle[i].removeClassName("movablepiece");  
  }  
  for (var i = 0; i < puzzle.length; i++) {  
    if (puzzle[i].innerHTML == "") {  
      puzzle[i].removeClassName("movablepiece");  
  
      switch (i) {  
        case 0:  
          movePA(i + 1);  
          movePA(i + 4);  
          break;  
        case 1:  
        case 2:  
          movePA(i - 1);  
          movePA(i + 1);  
          movePA(i + 4);  
          break;  
        case 3:  
          movePA(i - 1);  
          movePA(i + 4);  
          break;  
        case 4:  
          movePA(i - 4);  
          movePA(i + 4);  
          movePA(i + 1);  
          break;  
      }  
    }  
  }  
}
```


Links

- <http://codd.cs.gsu.edu/~nclark20/15puzzle/home.html>
- <https://github.com/nclark20/15puzzle>

Project info

Total Time Spent: 12 hours

Programming concepts: HTML, CSS, JavaScript, JQuery

Total lines of JavaScript: 210 lines

Total lines of CSS: 144 lines

Total lines of HTML: 77 lines