

Web Development – Mr. Turner

Project – Slider Puzzle

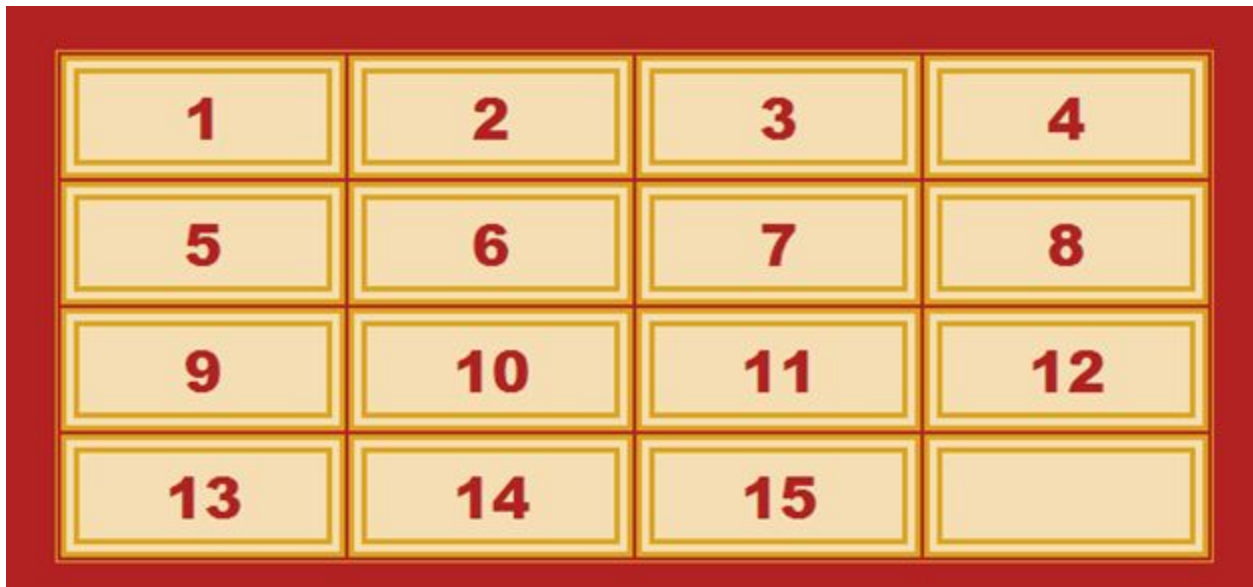
Project Overview

Write a simple slider game where the user puts numbers in order by moving “tiles” around.

Display

The list below includes the essential elements of the page.

- The page should be dressed up using CSS.
- Instructions.
- A start button.
- A reset button.
- An output area that shows the number of moves the player has made toward solving the puzzle.
- There should be a board with 16 spaces and 15 numbered tiles. You may use any style or colors you like. You may use a table.



Functionality

The page opens with the puzzle in its solved state. The user will have to shuffle it manually.

Note that a puzzle like this cannot be randomly shuffled. If the pieces are simply taken off the board and replaced, they may be done so in a way that makes it impossible to solve the puzzle.

The user clicks on one of the numbers adjacent to the empty tile and that number moves to the empty space, leaving its original space vacant.

Clicking on tiles that are **not** adjacent to the empty space will have no effect.

After the user is finished shuffling, they should click the start button. The number of moves will be set to 0. At that point, the program will start counting moves. Every time the user clicks and **successfully** moves a number into a blank spot, the number of moves will increase.

When the user clicks reset, the page will stop counting moves. Remember that the number of moves does not reset to 0 until the start button is clicked.

Update the display accordingly.

The game should tell the user when (s)he wins.

Enhancements

- Create a shuffling function which will only be available when the puzzle is completely solved. It will shuffle the board by making 1000 random moves.

Necessary Programming Skills

- Comprehension of the specifications sheet.
- Planning
 - Figure out the information you need to keep track of.
 - This information will become your global variables.
 - Plan out the individual tasks your program must perform.
 - Think through the steps for each task.
 - Think through the information your task needs (where does it come from?).
 - These will become your functions.
 - Plan out the user interface.
 - You can start with the barest interface, but you should have an idea what you want the final product to look like.
- Managing your variables
 - What's global, local, and passed through as parameters (hint - this program can use all three)?
 - Do you have your back end separated from your front end?
 - This program almost begs you keep track of everything through the display. Your program should hold the puzzle in the background and have the background inform the display.
- Assigning functions as tasks
 - Does your program sort out the different tasks into their own functions?
- Sequencing

- Does your program sequence from the user interaction into the necessary functions?
 - Is there an efficiency to your code that flows from the design document?
- An intuitive user experience
 - Is your display appropriate to the program (what's viewable and what scrolling has to be done)?
 - Is your display adaptable to other resolutions?
 - Is the interface intuitive?