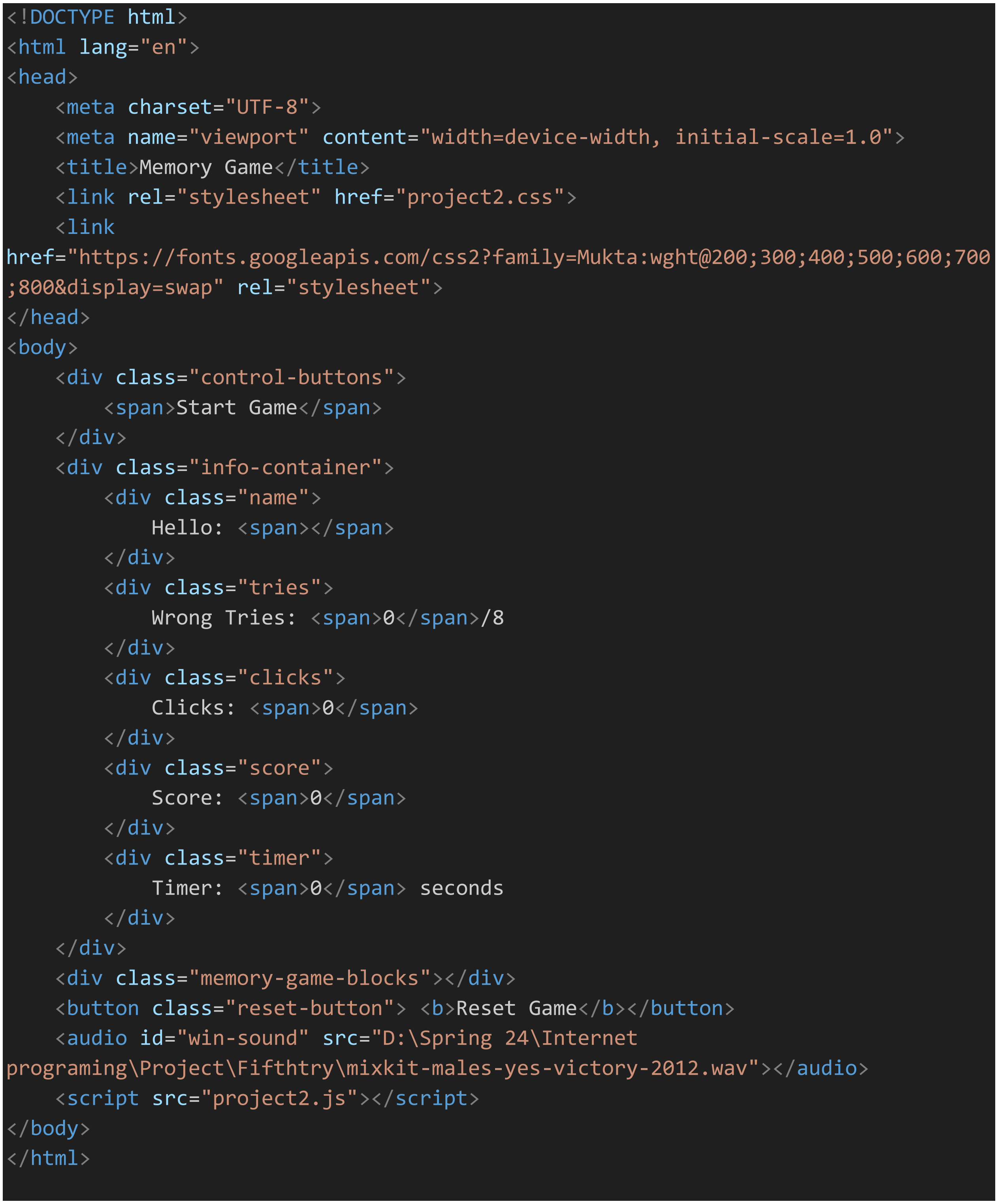
CARD MATCHING GAME

Internet programing project

OMAR ASHRAF MOSTAFA SAMY
20p1542

Html code:

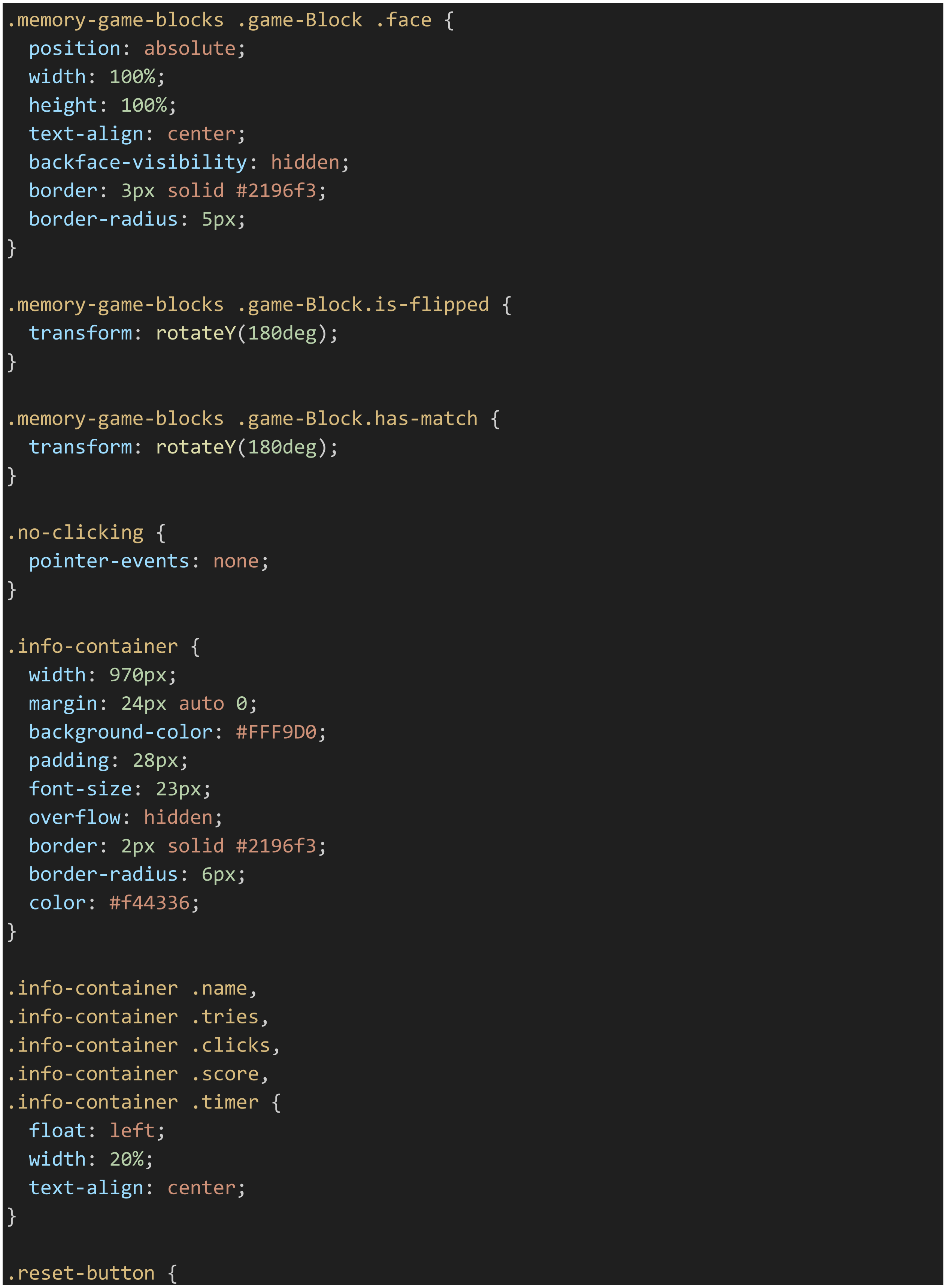


```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Memory Game</title>
    <link rel="stylesheet" href="project2.css">
    <link
href="https://fonts.googleapis.com/css2?family=Mukta:wght@200;300;400;500;600;700
;800&display=swap" rel="stylesheet">
</head>
<body>
    <div class="control-buttons">
        <span>Start Game</span>
    </div>
    <div class="info-container">
        <div class="name">
            Hello: <span></span>
        </div>
        <div class="tries">
            Wrong Tries: <span>0</span>/8
        </div>
        <div class="clicks">
            Clicks: <span>0</span>
        </div>
        <div class="score">
            Score: <span>0</span>
        </div>
        <div class="timer">
            Timer: <span>0</span> seconds
        </div>
    </div>
    <div class="memory-game-blocks"></div>
    <button class="reset-button"> <b>Reset Game</b></button>
    <audio id="win-sound" src="D:\Spring 24\Internet
programing\Project\Fifthtry\mixkit-males-yes-victory-2012.wav"></audio>
    <script src="project2.js"></script>
</body>
</html>
```

Css code:

```
* {
  box-sizing: border-box;
}

body {
  font-family: "Mukta", Tahoma, sans-serif;
  background-color: #CAF4FF;
}

.control-buttons {
  position: fixed;
  left: 0;
  top: 0;
  width: 100%;
  height: 100%;
  z-index: 2;
  background-color: rgba(0, 0, 0, 0.8);
}

.control-buttons span {
  position: absolute;
  left: 50%;
  top: 50%;
  transform: translate(-50%, -50%);
  background-color: #f44336;
  color: #FFF;
  padding: 15px 25px;
  font-size: 30px;
  text-align: center;
  border-radius: 6px;
  cursor: pointer;
}
.control-buttons span:hover {
  opacity: 0.8;
}

.memory-game-blocks {
  display: grid;
  grid-template-columns: repeat(4, 1fr);
  gap: 10px;
  width: 800px;
  margin: 20px auto;
```

```
}

.memory-game-blocks .game-Block {
  height: 120px;
  transition: transform 0.5s;
  transform-style: preserve-3d;
  cursor: pointer;
  position: relative;

}

.memory-game-blocks .game-Block .front {
 background-image: url("data:image/svg+xml,<svg id='patternId' width='100%'
height='100%' xmlns='http://www.w3.org/2000/svg'><defs><pattern id='a'
patternUnits='userSpaceOnUse' width='35.584' height='30.585'
patternTransform='scale(2) rotate(0)'><rect x='0' y='0' width='100%'
height='100%' fill='hsla(0,0%,100%,1)'/><path d='M36.908 9.243c-5.014 0-7.266
3.575-7.266 7.117 0 3.376 2.45 5.726 5.959 5.726 1.307 0 2.45-.463 3.244-
1.307.744-.811 1.125-1.903 1.042-3.095-.066-.811-.546-1.655-1.274-2.185-.596-
.447-1.639-.894-3.162-.546-.48.1-.778.58-.662 1.06.1.48.58.777 1.06.661.695-.149
1.274-.066 1.705.249.364.265.546.645.562.893.05.679-.165 1.308-.579 1.755-
.446.48-1.125.744-1.936.744-2.55 0-4.188-1.538-4.188-3.938 0-2.466 1.44-5.347
5.495-5.347 2.897 0 6.008 1.888 6.388 6.058.166 1.804.067 5.147-2.598
7.034a.868.868 0 00-.142.122c-1.311.783-2.87 1.301-4.972 1.301-4.088 0-6.123-
1.952-8.275-4.021-2.317-2.218-4.7-4.518-9.517-4.518-4.094 0-6.439 1.676-8.479
3.545.227-1.102.289-2.307.17-3.596-.496-5.263-4.567-7.662-8.159-7.662-5.015 0-
7.265 3.574-7.265 7.116 0 3.377 2.45 5.727 5.958 5.727 1.307 0 2.449-.463 3.243-
1.308.745-.81 1.126-1.903 1.043-3.095-.066-.81-.546-1.654-1.274-2.184-.596-.447-
1.639-.894-3.161-.546-.48.1-.778.58-.662 1.06.099.48.579.777 1.059.66.695-.148
1.275-.065 1.705.25.364.264.546.645.563.893.05.679-.166 1.307-.58 1.754-.447.48-
1.125.745-1.936.745-2.549 0-4.188-1.539-4.188-3.939 0-2.466 1.44-5.345 5.495-
5.345 2.897 0 6.008 1.87 6.389 6.057.163 1.781.064 5.06-2.504 6.96-1.36.864-2.978
1.447-5.209 1.447-4.088 0-6.124-1.952-8.275-4.021-2.317-2.218-4.7-4.518-9.516-
4.518v1.787c4.088 0 6.123 1.953 8.275 4.022 2.317 2.218 4.7 4.518 9.516 4.518 4.8
0 7.2-2.3 9.517-4.518 2.151-2.069 4.187-4.022 8.275-4.022s6.124 1.953 8.275
4.022c2.318 2.218 4.701 4.518 9.517 4.518 4.8 0 7.2-2.3 9.516-4.518 2.152-2.069
4.188-4.022 8.276-4.022s6.123 1.953 8.275 4.022c2.317 2.218 4.7 4.518 9.517
4.518v-1.788c-4.088 0-6.124-1.952-8.275-4.021-2.318-2.218-4.701-4.518-9.517-
4.518-4.103 0-6.45 1.683-8.492 3.556.237-1.118.304-2.343.184-3.656-.497-5.263-
4.568-7.663-8.16-7.663z'  stroke-width='1' stroke='none' fill='hsla(214, 59%,
59%, 1)'/><path d='M23.42 41.086a.896.896 0 01-.729-.38.883.883 0 01.215-
1.242c2.665-1.887 2.764-5.23 2.599-7.034-.38-4.187-3.492-6.058-6.389-6.058-4.055
0-5.495 2.88-5.495 5.346 0 2.4 1.639 3.94 4.188 3.94.81 0 1.49-.265 1.936-
.745.414-.447.63-1.076.58-1.755-.017-.248-.2-.629-.547-.893-.43-.315-1.026-.398-
1.704-.249a.868.868 0 01-1.06-.662.868.868 0 01.662-1.059c1.523-.348 2.566.1
3.161.546.729.53 1.209 1.374 1.275 2.185.083 1.191-.298 2.284-1.043 3.095-
.794.844-1.936 1.307-3.244 1.307-3.508 0-5.958-2.35-5.958-5.726 0-3.542 2.25-
7.117 7.266-7.117 3.591 0 7.663 2.4 8.16 7.663.347 3.79-.828 6.868-3.344
8.656a.824.824 0 01-.53.182zm0-30.585a.896.896 0 01-.729-.38.883.883 0 01.215-
1.242c2.665-1.887 2.764-5.23 2.599-7.034-.381-4.187-3.493-6.058-6.389-6.058-4.055
0-5.495 2.88-5.495 5.346 0 2.4 1.639 3.94 4.188 3.94.81 0 1.49-.266 1.936-
.746.414-.446.629-1.075.58-1.754-.017-.248-.2-.629-.547-.894-.43-.314-1.026-.397-
1.705-.248A.868.868 0 0117.014.77a.868.868 0 01.662-1.06c1.523-.347 2.566.1
3.161.547.729.53 1.209 1.374 1.275 2.185.083 1.191-.298 2.284-1.043 3.095-
.794.844-1.936 1.307-3.244 1.307-3.508 0-5.958-2.35-5.958-5.726 0-3.542 2.25-
7.117 7.266-7.117 3.591 0 7.663 2.4 8.16 7.663.347 3.79-.828 6.868-3.344
8.656a.824.824 0 01-.53.182zm29.956 1.572c-4.8 0-7.2-2.3-9.517-4.518-2.151-2.069-
4.187-4.022-8.275-4.022S29.46 5.486 27.31 7.555c-2.317 2.218-4.7 4.518-9.517
4.518-4.8 0-7.2-2.3-9.516-4.518C6.124 5.486 4.088 3.533 0 3.533s-6.124 1.953-
8.275 4.022c-2.317 2.218-4.7 4.518-9.517 4.518-4.8 0-7.2-2.3-9.516-4.518-2.152-
2.069-4.188-4.022-8.276-4.022V1.746c4.8 0 7.2 2.3 9.517 4.518 2.152 2.069 4.187
4.022 8.275 4.022s6.124-1.953 8.276-4.022C-7.2 4.046-4.816 1.746 0 1.746c4.8 0
7.2 2.3 9.517 4.518 2.151 2.069 4.187 4.022 8.275 4.022s6.124-1.953 8.275-
4.022c2.318-2.218 4.7-4.518 9.517-4.518 4.8 0 7.2 2.3 9.517 4.518 2.151 2.069
4.187 4.022 8.275 4.022s6.124-1.953 8.275-4.022c2.317-2.218 4.7-4.518 9.517-
4.518v1.787c-4.088 0-6.124 1.953-8.275 4.022-2.317 2.234-4.717 4.518-9.517
4.518z'  stroke-width='1' stroke='none' fill='hsla(346, 82%, 52%,
1)'/></pattern></defs><rect width='800%' height='800%' transform='translate(0,0)'
fill='url(%23a)'/></svg>");
  background-size: cover;
  line-height: 120px;

}
.memory-game-blocks .game-Block .front:hover{
  box-shadow: 0px 0px 2px 2px lemonchiffon;
}


.memory-game-blocks .game-Block .back {
  background-color: lemonchiffon;
  transform: rotateY(180deg);
  color: #FFF;
}

.memory-game-blocks .game-Block .back img {
  width: 80px;
  height: 80px;
  margin-top: 20px;
}
```

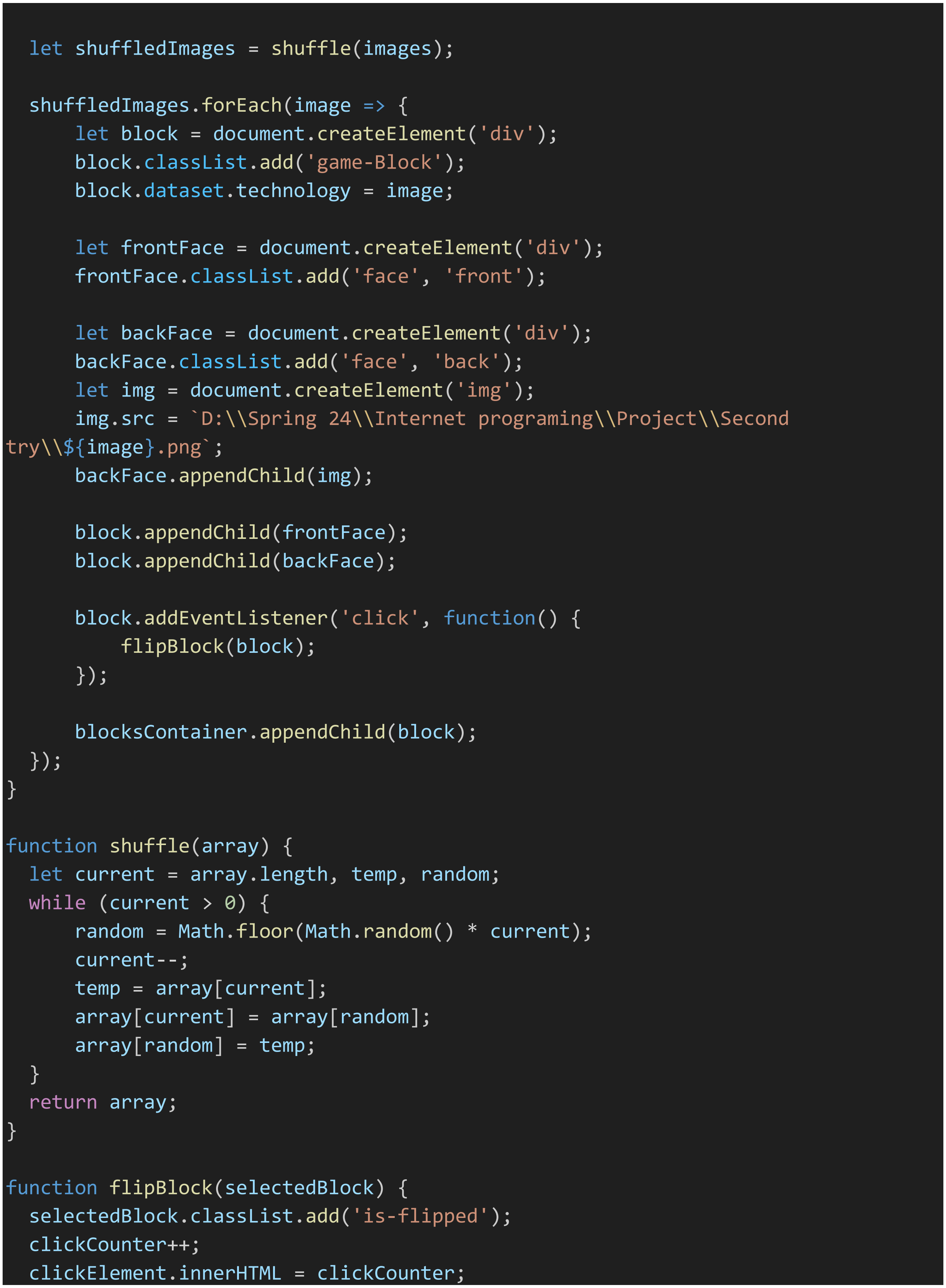


```
.memory-game-blocks .game-Block .face {
  position: absolute;
  width: 100%;
  height: 100%;
  text-align: center;
  backface-visibility: hidden;
  border: 3px solid #2196f3;
  border-radius: 5px;
}

.memory-game-blocks .game-Block.is-flipped {
  transform: rotateY(180deg);
}

.memory-game-blocks .game-Block.has-match {
  transform: rotateY(180deg);
}

.no-clicking {
  pointer-events: none;
}

.info-container {
  width: 970px;
  margin: 24px auto 0;
  background-color: #FFF9D0;
  padding: 28px;
  font-size: 23px;
  overflow: hidden;
  border: 2px solid #2196f3;
  border-radius: 6px;
  color: #f44336;
}

.info-container .name,
.info-container .tries,
.info-container .clicks,
.info-container .score,
.info-container .timer {
  float: left;
  width: 20%;
  text-align: center;
}

.reset-button {
```

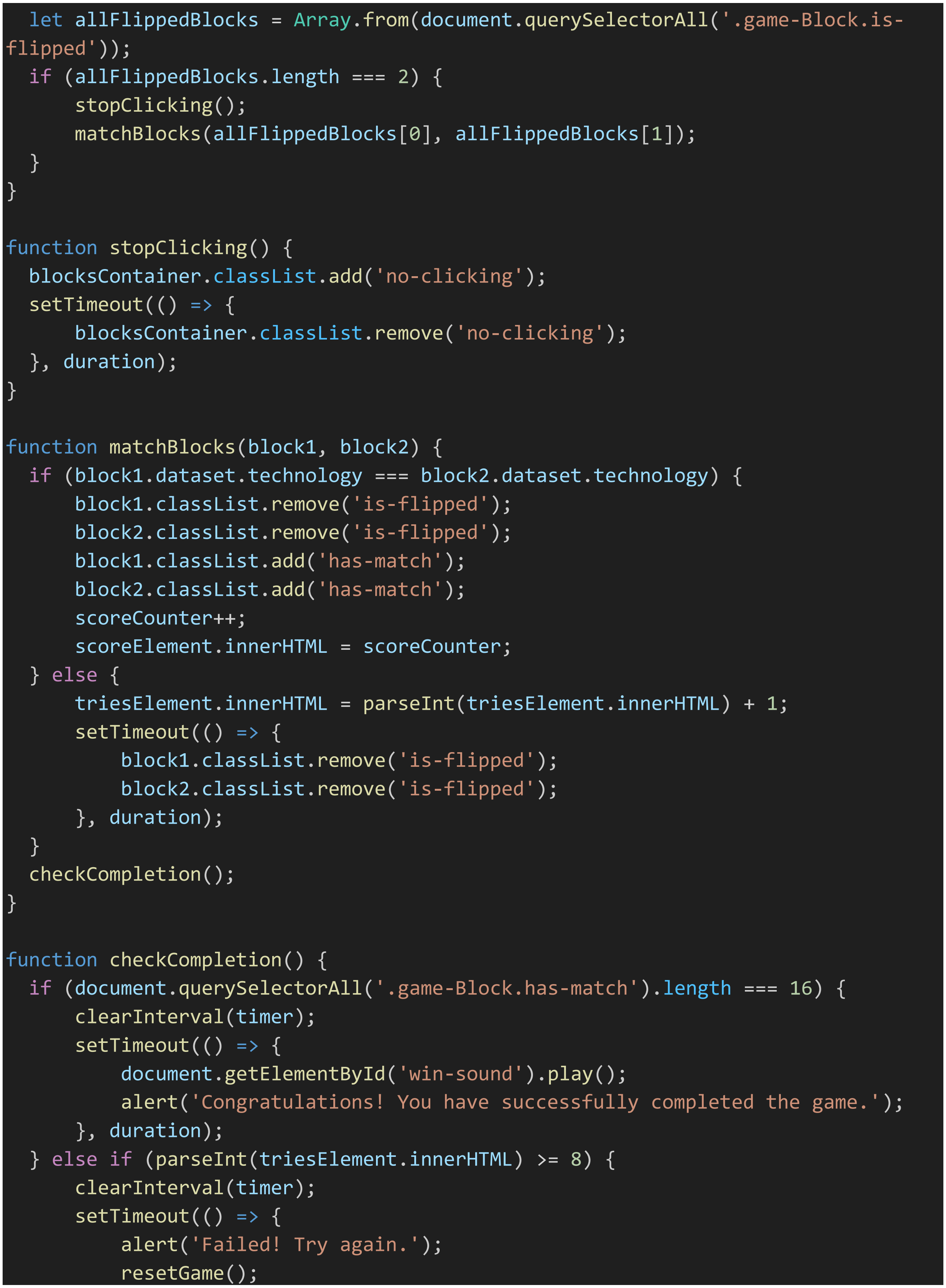
```css
  display: block;
  margin: 20px auto;
  padding: 10px 20px;
  font-size: 28px;
  background-color: #FFF9D0;
  color: #f44336;
  border: none;
  border-radius: 6px;
  cursor: pointer;
}

.reset-button:hover {
  opacity: 0.8;
}
```

Js code:

```
document.querySelector(".control-buttons span").onclick = function() {
  let yourname = prompt("What's your Name?");
  if (yourname == null || yourname == "") {
      document.querySelector(".name span").innerHTML = 'Unknown Player';
  } else {
      document.querySelector(".name span").innerHTML = yourname;
  }
  document.querySelector(".control-buttons").remove();
  startTimer();
  initializeGame();
};

let duration = 3000;
let blocksContainer = document.querySelector(".memory-game-blocks");
let triesElement = document.querySelector('.tries span');
let clickElement = document.querySelector('.clicks span');
let scoreElement = document.querySelector('.score span');
let timerElement = document.querySelector('.timer span');
let clickCounter = 0;
let scoreCounter = 0;
let timer;
let seconds = 0;

document.querySelector(".reset-button").onclick = resetGame;

function initializeGame() {
  blocksContainer.innerHTML = '';
  triesElement.innerHTML = '0';
  clickElement.innerHTML = '0';
  scoreElement.innerHTML = '0';
  clickCounter = 0;
  scoreCounter = 0;

  let images = [
      "cherries", "cherries",
      "pineapple", "pineapple",
      "tomato", "tomato",
      "watermelon", "watermelon",
      "strawberry", "strawberry",
      "orange", "orange",
      "grapes", "grapes",
      "lemon", "lemon"
  ];
```



```
  let shuffledImages = shuffle(images);

  shuffledImages.forEach(image => {
      let block = document.createElement('div');
      block.classList.add('game-Block');
      block.dataset.technology = image;

      let frontFace = document.createElement('div');
      frontFace.classList.add('face', 'front');

      let backFace = document.createElement('div');
      backFace.classList.add('face', 'back');
      let img = document.createElement('img');
      img.src = `D:\\Spring 24\\Internet programing\\Project\\Second
try\\${image}.png`;
      backFace.appendChild(img);

      block.appendChild(frontFace);
      block.appendChild(backFace);

      block.addEventListener('click', function() {
          flipBlock(block);
      });

      blocksContainer.appendChild(block);
  });
}

function shuffle(array) {
  let current = array.length, temp, random;
  while (current > 0) {
      random = Math.floor(Math.random() * current);
      current--;
      temp = array[current];
      array[current] = array[random];
      array[random] = temp;
  }
  return array;
}

function flipBlock(selectedBlock) {
  selectedBlock.classList.add('is-flipped');
  clickCounter++;
  clickElement.innerHTML = clickCounter;
```



```
  let allFlippedBlocks = Array.from(document.querySelectorAll('.game-Block.is-flipped'));
  if (allFlippedBlocks.length === 2) {
      stopClicking();
      matchBlocks(allFlippedBlocks[0], allFlippedBlocks[1]);
  }
}

function stopClicking() {
  blocksContainer.classList.add('no-clicking');
  setTimeout(() => {
      blocksContainer.classList.remove('no-clicking');
  }, duration);
}

function matchBlocks(block1, block2) {
  if (block1.dataset.technology === block2.dataset.technology) {
      block1.classList.remove('is-flipped');
      block2.classList.remove('is-flipped');
      block1.classList.add('has-match');
      block2.classList.add('has-match');
      scoreCounter++;
      scoreElement.innerHTML = scoreCounter;
  } else {
      triesElement.innerHTML = parseInt(triesElement.innerHTML) + 1;
      setTimeout(() => {
          block1.classList.remove('is-flipped');
          block2.classList.remove('is-flipped');
      }, duration);
  }
  checkCompletion();
}

function checkCompletion() {
  if (document.querySelectorAll('.game-Block.has-match').length === 16) {
      clearInterval(timer);
      setTimeout(() => {
          document.getElementById('win-sound').play();
          alert('Congratulations! You have successfully completed the game.');
      }, duration);
  } else if (parseInt(triesElement.innerHTML) >= 8) {
      clearInterval(timer);
      setTimeout(() => {
          alert('Failed! Try again.');
          resetGame();
      }, duration);
  }
}

function resetGame() {
  clearInterval(timer);
  seconds = 0;
  timerElement.innerHTML = seconds;
  startTimer();
  initializeGame();
}

function startTimer() {
  timer = setInterval(() => {
      seconds++;
      timerElement.innerHTML = seconds;
  }, 1000);
}

window.onload = initializeGame;
```

Screenshots of the output:

Press the start game button to start the game.

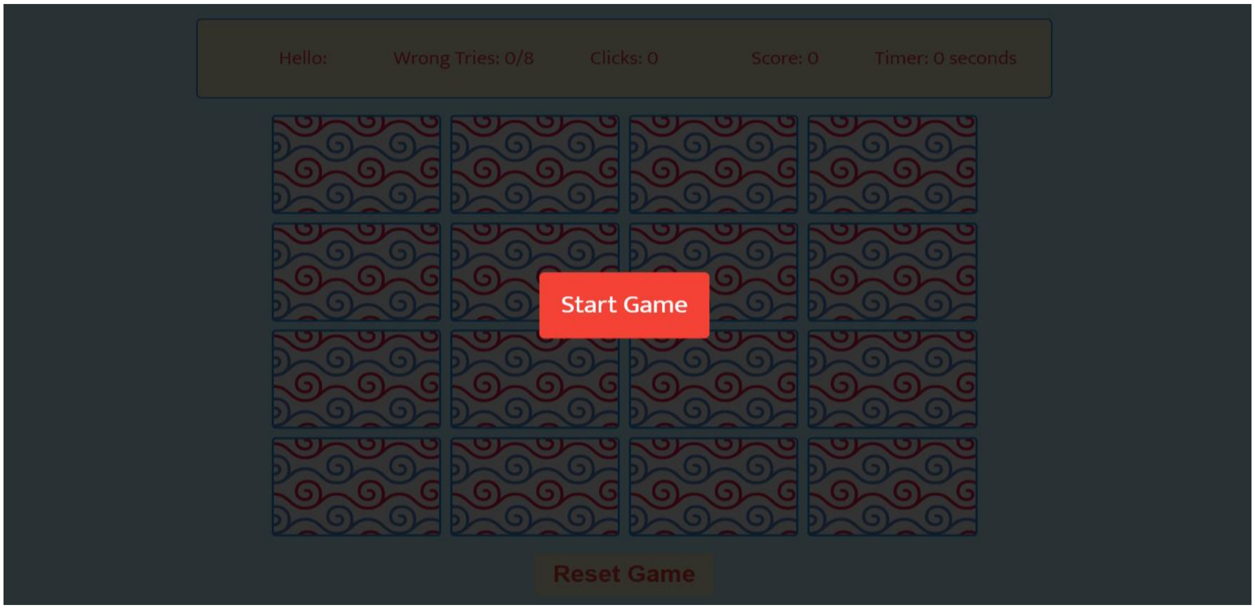
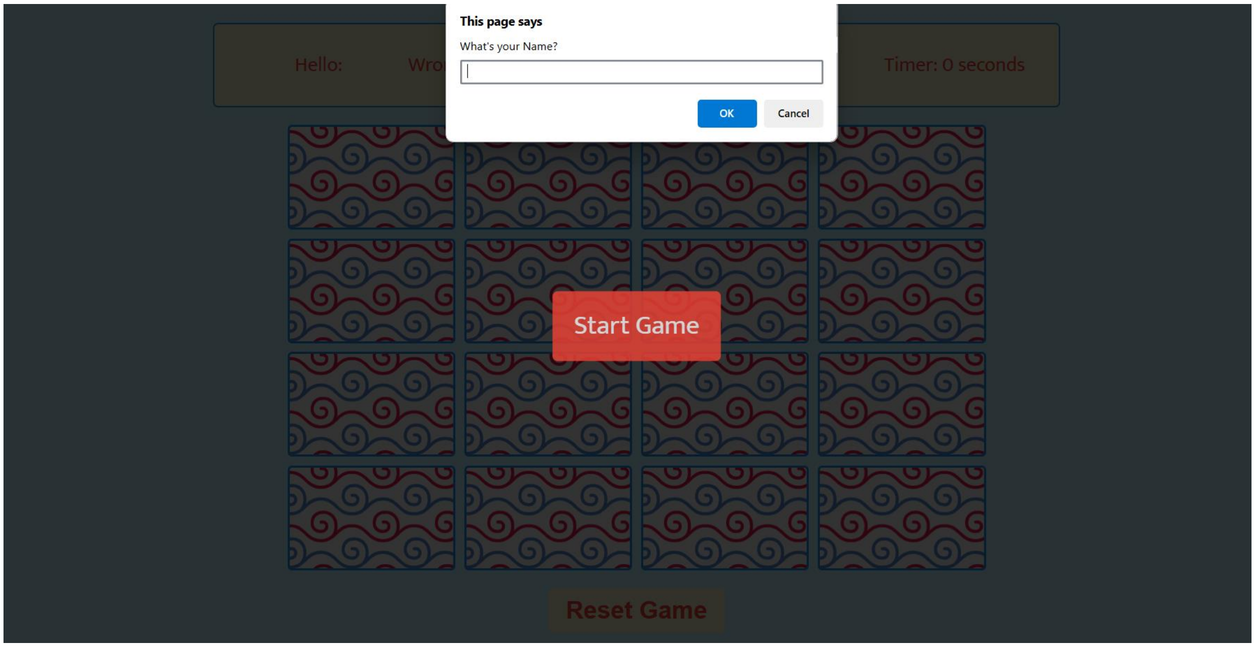


Figure 1 Start screen of the game.

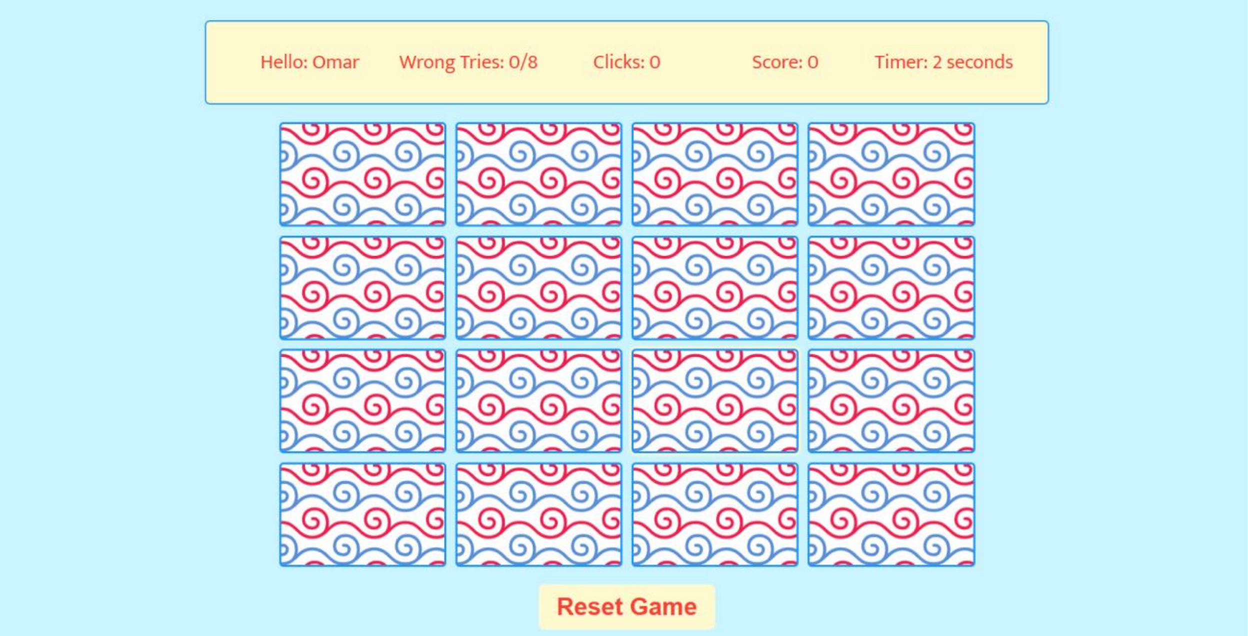
Write the name you want to enter the game by it.


This page says

What's your Name?

Figure 2 Enter your name.

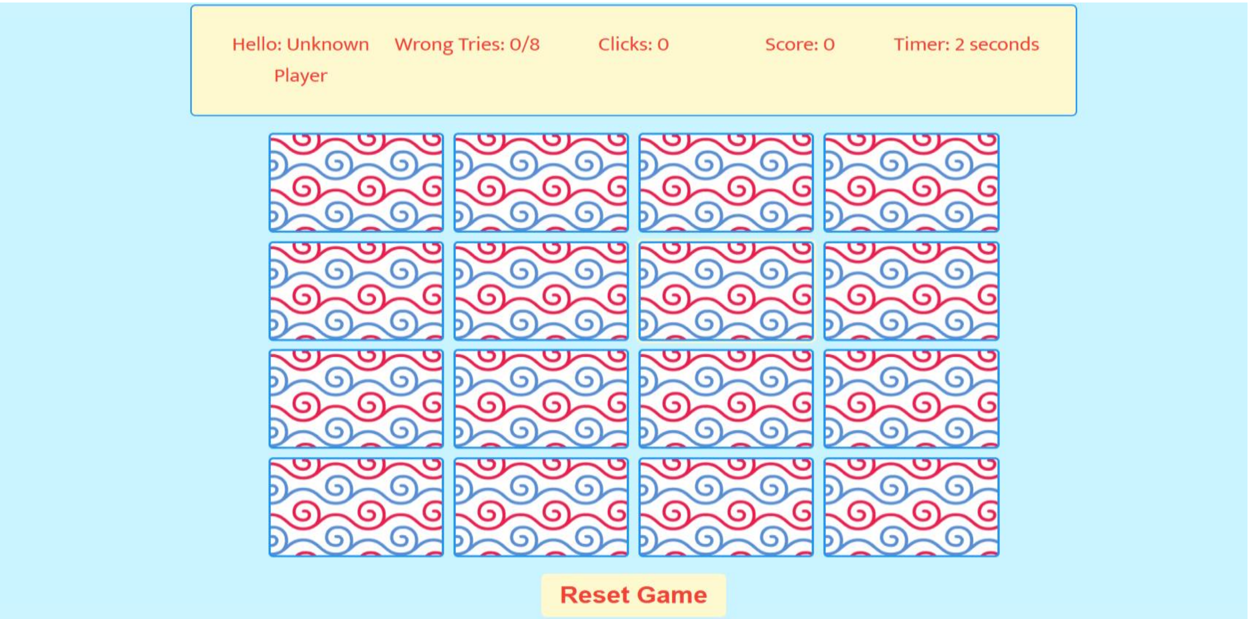
Now you Enter the game after you insert your name you can start to play now.



Hello: Omar Wrong Tries: 0/8 Clicks: 0 Score: 0 Timer: 2 seconds

Figure 3 Game screen after insert name of the player.

If you enter the game without, write your name and press ok or cancel it will be written as unknown player.



Hello: Unknown Player Wrong Tries: 0/8 Clicks: 0 Score: 0 Timer: 2 seconds

Figure 4 Game screen if you press ok or cancel without write a name.

Timer: start to count time in seconds from the beginning of the game till the end of the game.

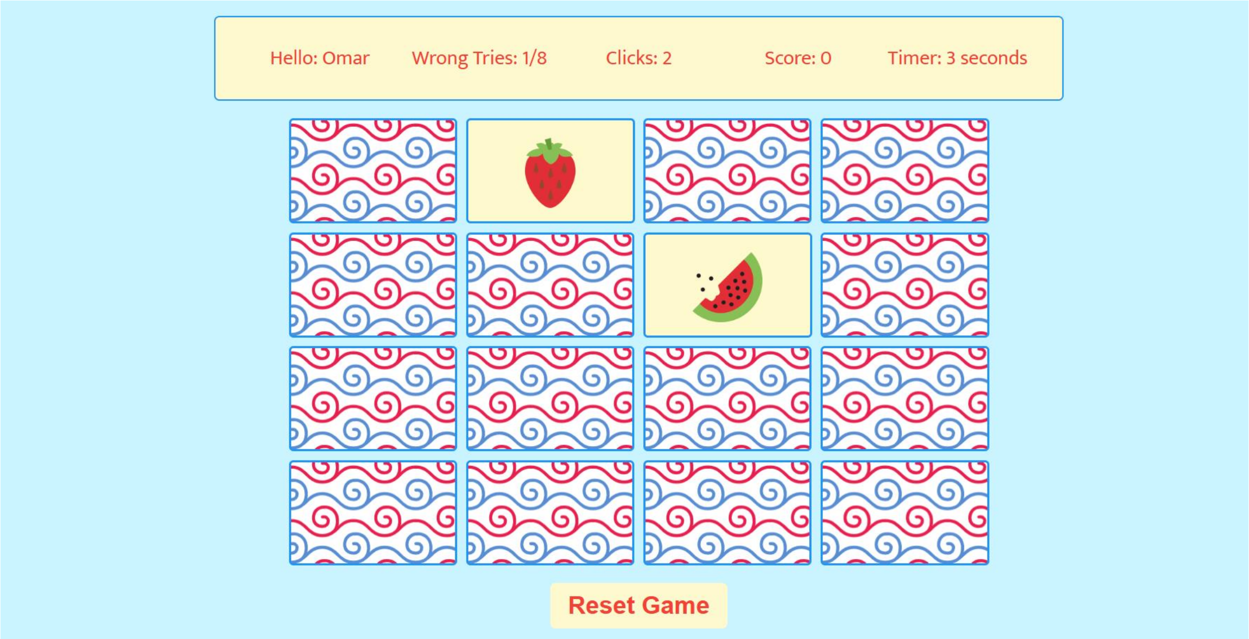
clicks: it is a counter counts the number of clicks you press from the beginning till the end.

Wrong tries: it the number of trials that you fail to match 2 images each 2 clicks = 1 Wrong try also if it reaches 8 wrong tries the player fail and start the game again.

Score: it counts number of times you successfully can match 2 images.

Reset Game: it is a button if you press at any time the game reset and start it again from the beginning.

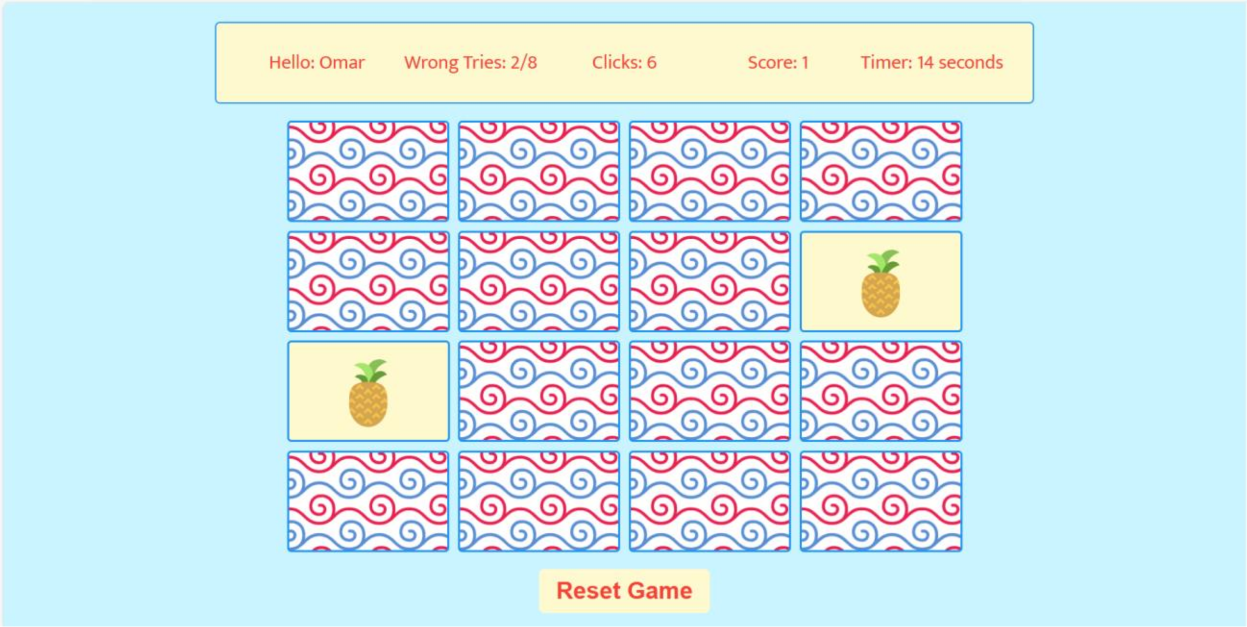
If the two images are not matched, the two cards are flipped again after 3 seconds I enhance this point as it is mentioned in the description of the project to be equal 5 seconds, but I put it 3 seconds to be more practical.



Hello: Omar Wrong Tries: 1/8 Clicks: 2 Score: 0 Timer: 3 seconds

Figure 5 start to play the game.

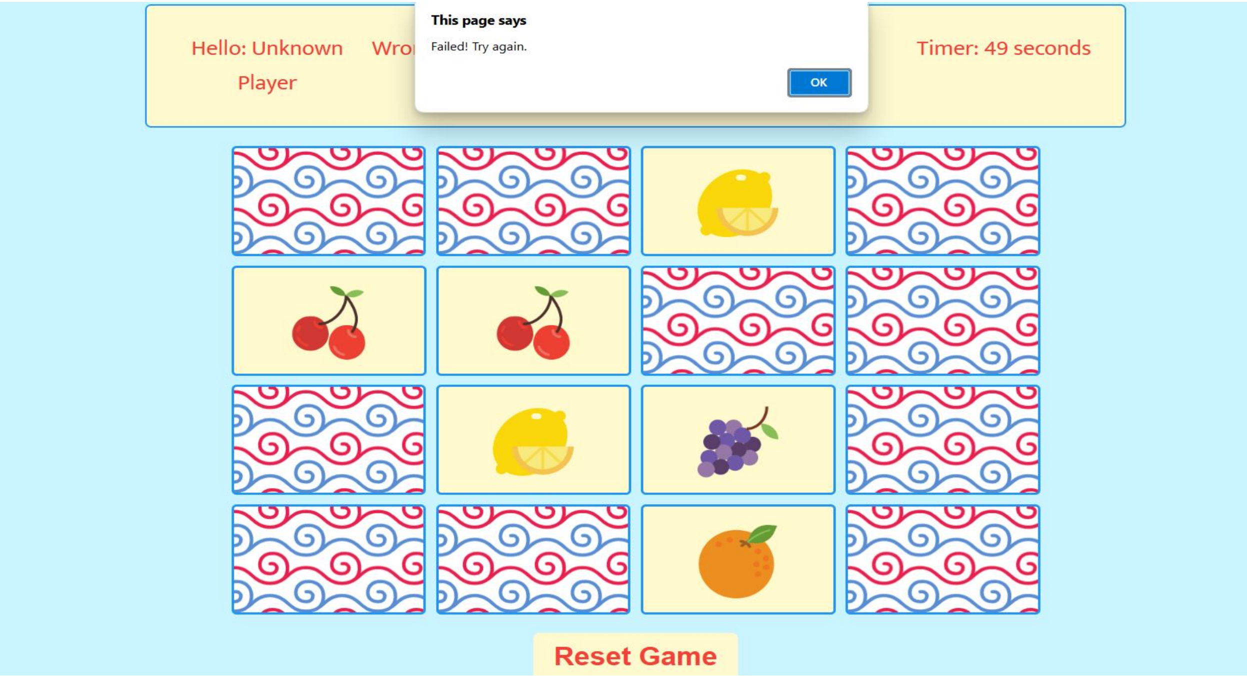
The score increases by one after 2 images matched successfully also. If the two images are matched, the two cards are excluded from the board after 3 seconds, I enhance this point as it is mentioned in the description of the project to be equal 5 seconds, but I put it 3 seconds to be more practical.



Hello: Omar Wrong Tries: 2/8 Clicks: 6 Score: 1 Timer: 14 seconds

Figure 6 the first 2 images matched

If the number of Wrong tries becomes greater than or equal to 8 the player fails and receive message that he fails, then reset the game from the beginning after press ok.



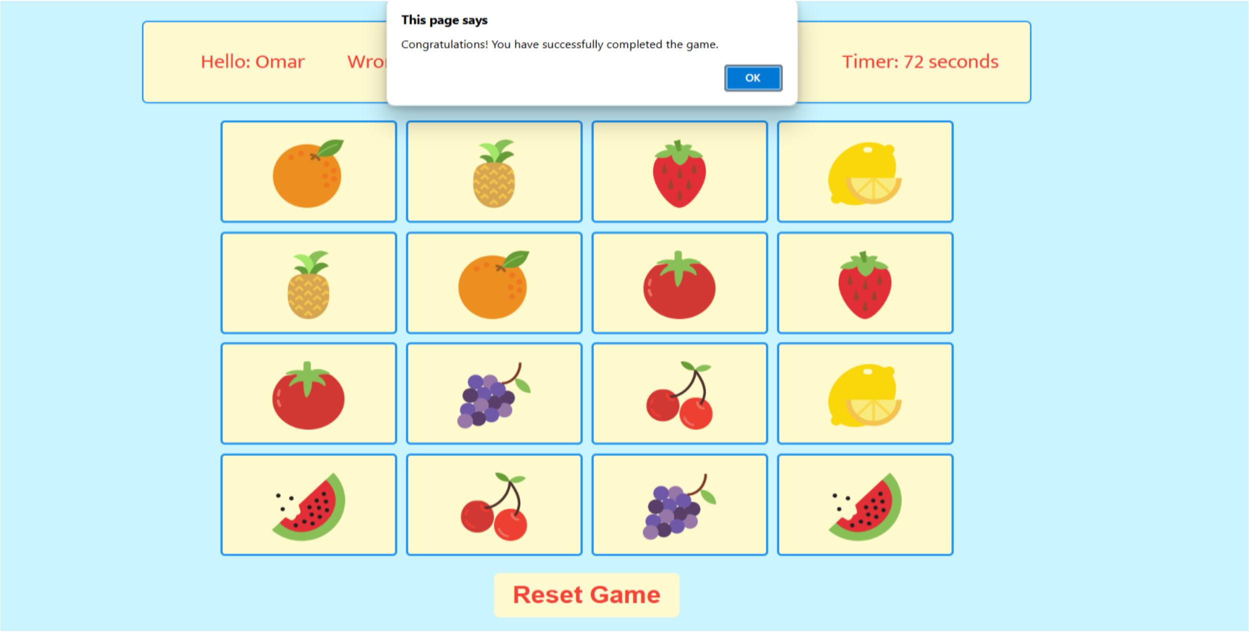
Hello: Unknown Player Wro Timer: 49 seconds

This page says

Failed! Try again.

Figure 7 the player reaches the maximum number of wrong tries (=8), so he failed.

When the player wins the game and matches all the images successfully appear to him greeting message congratulate him that he success and complete the game and after he press ok on this message there is a sound effect also greeting the player on his success in the game.


This page says

Congratulations! You have successfully completed the game.

Hello: Omar Wro Timer: 72 seconds

Figure 8 player win the game and have congratulation message.

This screen after the player wins and complete the game mention his score='8' and time he takes to complete the game='72 seconds' and number of wrong tries he has till reach the end of game='6' in this case if he gets two more extra wrong tries, he will be fail because equals 8 player fails.

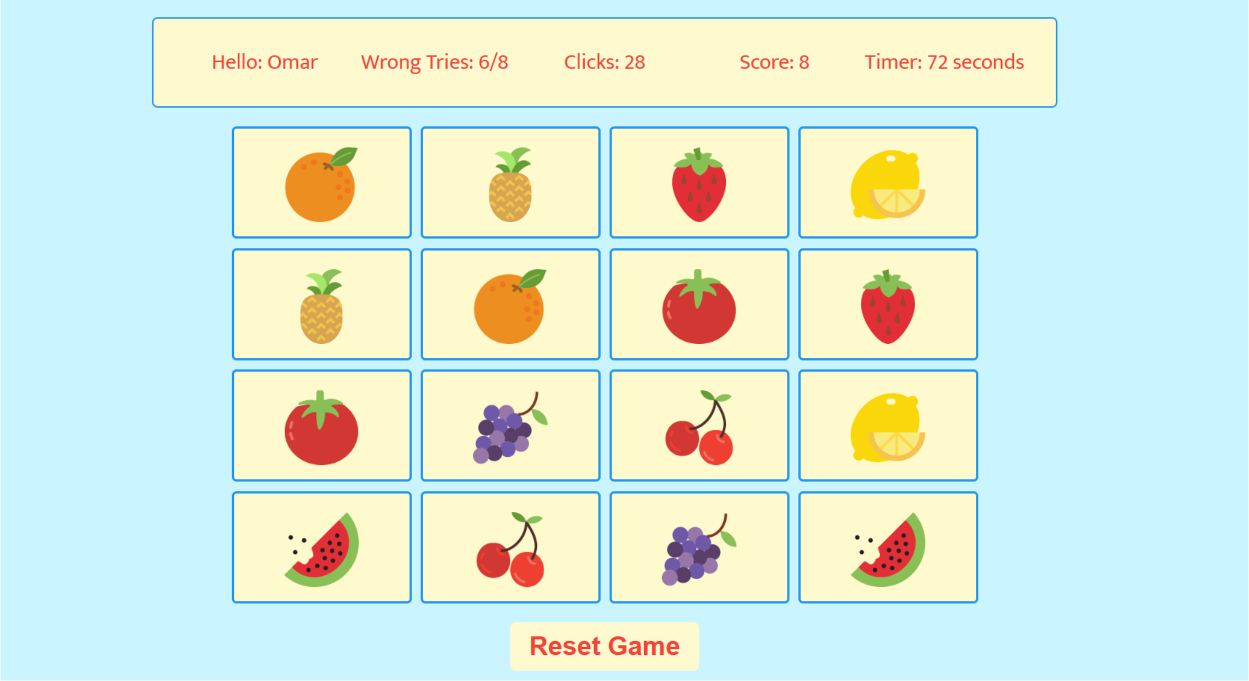
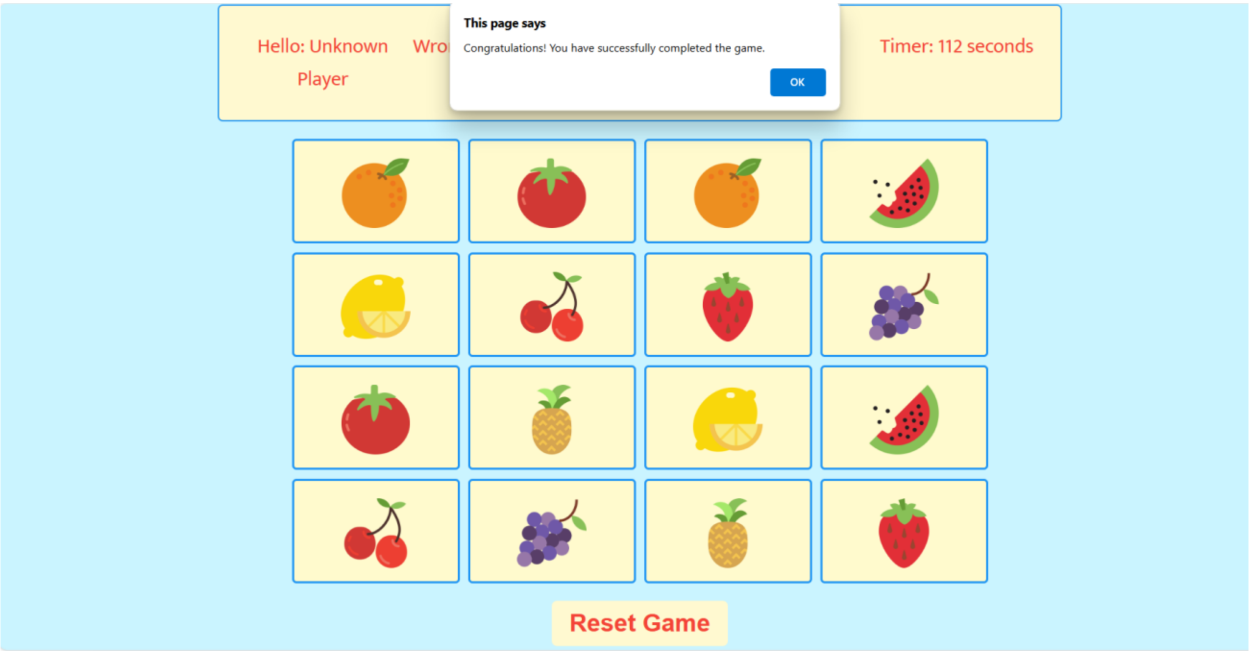

Hello: Omar Wrong Tries: 6/8 Clicks: 28 Score: 8 Timer: 72 seconds

Figure 9 screen of the game after player wins and the game end.

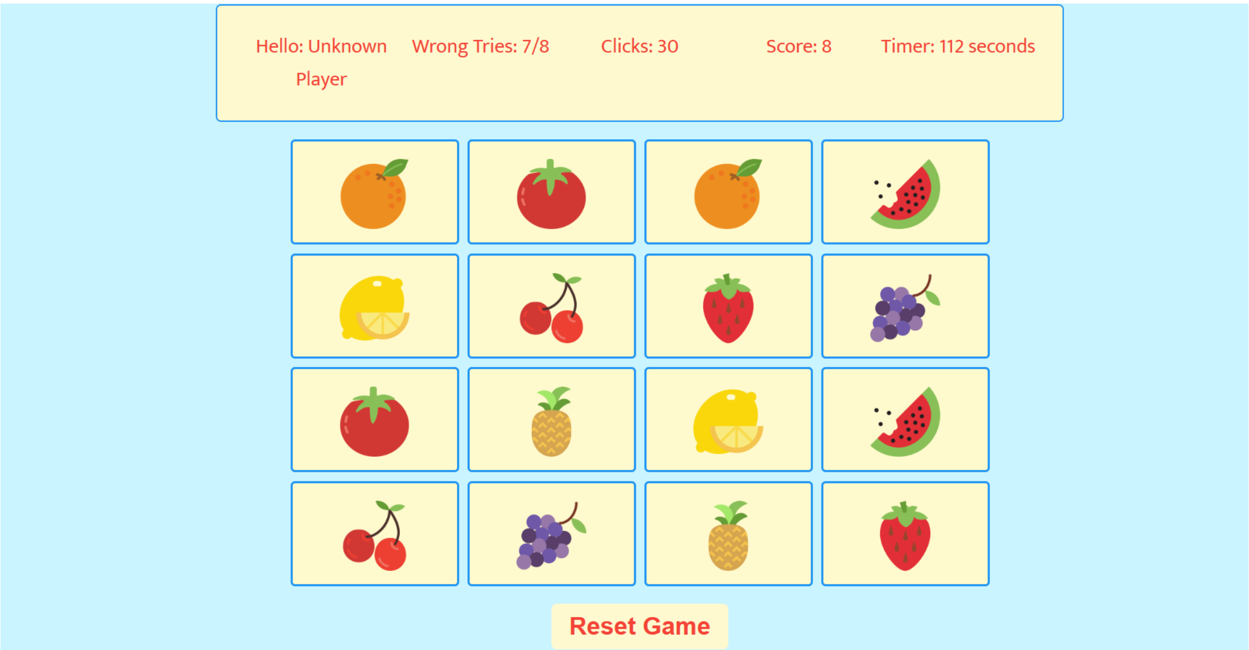
And here it is another player who also win and complete the game successfully with long time and a greater number of wrong tries.



Hello: Unknown Player Wro Timer: 112 seconds

This page says

Congratulations! You have successfully completed the game.



Hello: Unknown Player Wrong Tries: 7/8 Clicks: 30 Score: 8 Timer: 112 seconds