

MỤC TIÊU:

Kết thúc bài thực hành này bạn có khả năng

- ✓ Làm việc được với các thuộc tính css
- ✓ Sử dụng được mouse event, key event

NỘI DUNG:

PHẦN I:

Bài 1: lab61.html - Collapsible accordion component

Tạo trang web có giao diện và xử lý như sau:



• Gợi ý cấu trúc tập tin html

```
<style>
  .active {
    display: block !important;
  }
  .myText {
    display: none;
  }
  .title {
    font-size: 1.5em;
    background-color: #ddd;
  }
</style>
```

X

```
<body>
  <div class="container">
    <div class="title">Title #1</div>
    <div class="myText">Just some text #1</div>
    <div class="title">Title #2</div>
    <div class="myText">Just some text #2</div>
    <div class="title">Title #3</div>
    <div class="myText">Just some text #3</div>
  </div>
  <script type="text/javascript" src="chapter6.js">
  </script>
</body>
```

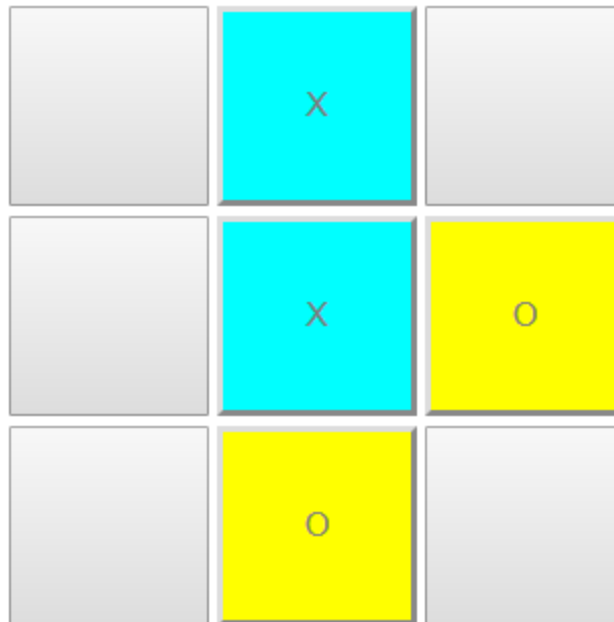
- **Gợi ý javascript**

```
const menus = document.querySelectorAll(".title");
const openText = document.querySelectorAll(".myText");
menus.forEach((el) => {
  el.addEventListener("click", (e) => {
    console.log(el.nextElementSibling);
    remover();
    el.nextElementSibling.classList.toggle("active");
  })
})
function remover() {
  openText.forEach((ele) => {
    ele.classList.remove("active");
  })
}
```

Bài 2: lab62.html

Tạo trang web cho phép 2 người chơi trò tictactoe có giao diện như sau.

TICTACTOE GAME



Mô tả hoạt động

- Hiển thị 9 nút trống
- Người chơi nhấp vào nút trống
- Lượt lẻ: tạo nhãn X và nền xanh (aqua)
- Lượt chẵn: tạo nhãn O nền vàng
- Vô hiệu hóa nút được click để không click được nữa
- Nếu tổng click 8 lần thì báo kết thúc và load lại trang web

Hướng dẫn thực hiện

- Sử dụng trang web Lab61.html đính kèm chú ý cấu trúc mã sau

```
<table align="center">
  <tr>
    <td><button></button></td>
    <td><button></button></td>
    <td><button></button></td>
  </tr>
  ...
</table>
```

- Bổ sung mã javascript theo hướng dẫn sau

- Khai báo biến toàn cục và hàm
`var flag = true; // để chọn nhãn (true: X, false: O)`
`function mark(index){...} // để điều khiển sự kiện click nút thứ index`
- Điều khiển sự kiện cho các nút
`<button onclick="mark(0)"></button>`
 ...
`<button onclick="mark(8)"></button>`
- Viết mã cho hàm mark()
`var button = document.getElementsByTagName("button")[index];`
`button.innerText = flag?"X":"O";`
`button.style.backgroundColor = flag?"aqua":"yellow";`
`button.setAttribute("disabled", "true");`
`flag = !flag; // đảo cờ để lần sau hiển thị nhãn và màu nền khác`
- Chạy thử lần 1
- Kiểm soát kết thúc game
 - o Khai báo biến toàn cục để ghi nhận số lần click nút
`var count = 0; // để đếm số lần click nút trên bàn cờ`
 - o Bổ sung mã script vào cuối hàm mark() để tăng số lần click và kiểm tra điều kiện kết thúc
`count++; // đếm số lần click`
`if(count == 8){`
 `alert("Game over!");`
 `location.reload(); }`
- Chạy thử lần 2

PHẦN II:

Bài 3: lab63.html – image rollover và slideshow

Thực hiện lại bài ví dụ demo image rollover và slideshow hình ảnh trong slide bài giảng

Hình ảnh gợi ý:



Hình1: bình thường



Hình2: khi đưa chuột lên

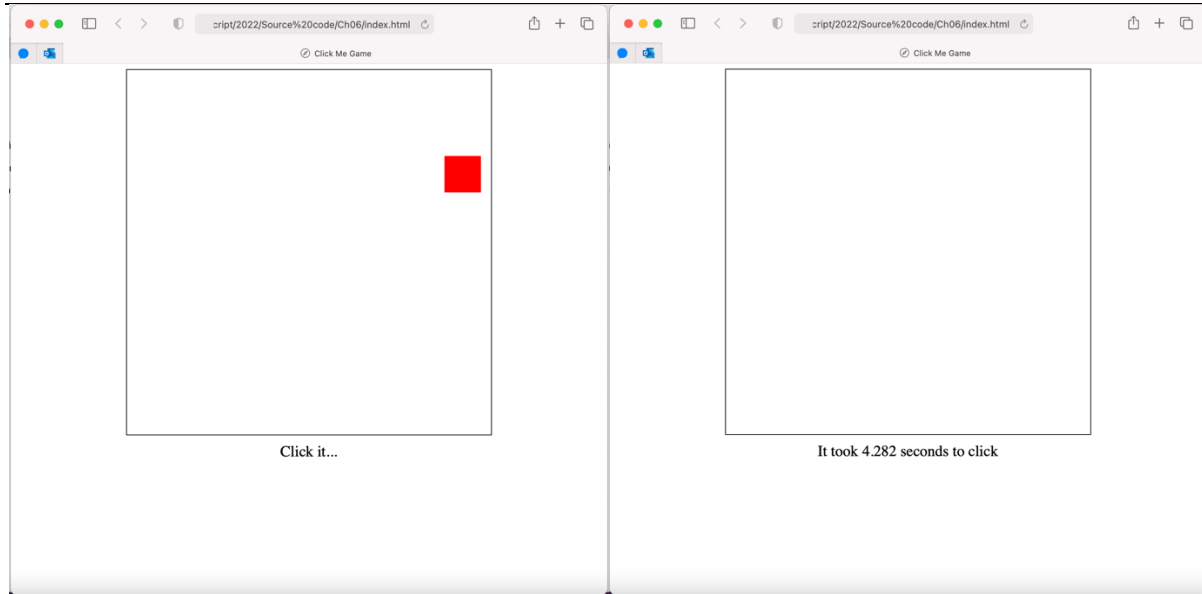


First Prev Next Last

- Chủ đề hình ảnh (tự chọn)
- Đối với slideshow
 - o Tối thiểu 5 hình ảnh trong slideshow
 - o Dùng javascript xử lý đầy đủ các control sau: first, last, prev, next

Bài 4: lab64.html – box clicker speed test game

Xây dựng trang web được mô tả như hình sau



Mô tả hoạt động

- Khi trang web vừa load , nhấp vào ô màu đỏ càng nhanh càng tốt khi nó xuất hiện.
- Ô màu đỏ đặt ngẫu nhiên trong div (vùng box đen)
- Xử lý sự kiện: theo dõi thời gian bắt đầu và nhấp chuột để tính toán thời lượng (duration) của sự kiện click.

Hướng dẫn viết mã

- Cấu trúc mã HTML như sau

```
<!DOCTYPE html>
<html>
<head>
  <title>Click Me Game</title>
  <style>
    .output {
      width: 500px;
      height: 500px;
      border: 1px solid black;
      margin: auto;
      text-align: center;
    }
    .box {
      width: 50px;
      height: 50px;
      position: relative;
      top: 50px;
      left: 20%;
      background-color: red;
    }
    .message {
      text-align: center;
      padding: 10px;
      font-size: 1.3em;
    }
  </style>
</head>
<body>
  <div class="output"></div>
  <div class="message"></div>
  <script type="text/javascript" src="chapter6.js"></script>
</body>
</html>
```

- Javascript
 - o Lấy/Thiết lập 2 element

```
const output = document.querySelector('.output');
const message = document.querySelector('.message');
```

- Thiết lập message bắt đầu, thời gian game bắt đầu và tạo ô vuông (đỏ)

```
message.textContent = "Press to Start";
const box = document.createElement('div');
const game = {
  timer: 0,
  start: null
};
box.classList.add('box');
output.append(box);
```

- Tạo vị trí ngẫu nhiên

```
function ranNum(max) {
  return Math.floor(Math.random() * max);
}
```

- Load ô vuông ngẫu nhiên, lấy thời gian bắt đầu

```
function addBox() {
  message.textContent = 'Click it...';
  game.start = new Date().getTime();
  box.style.display = 'block';
  box.style.left = ranNum(450) + 'px';
  box.style.top = ranNum(450) + 'px';
}
```

- Sự kiện click chuột: tính toán thời lượng và hiển thị thông báo

```
box.addEventListener('click', () => {
  box.textContent = '';
  box.style.display = 'none';
  game.timer = setTimeout(addBox, ranNum(3000));
  if (!game.start) {
    message.textContent = 'Loading....';
  } else {
    const cur = new Date().getTime();
    const dur = (cur - game.start) / 1000;
    message.textContent = `It took ${dur} seconds to click`;
  }
});
```


***** Yêu cầu nộp bài:**

SV nén file (*hoặc share thư mục google drive*) bao gồm các yêu cầu đã thực hiện trên, nộp LMS đúng thời gian quy định của giảng viên. **KHÔNG NỘP BÀI COI NHƯ KHÔNG CÓ ĐIỂM.**

--- Hết ---