

Bài tập 1

Code html:

```
<form id="form">
  

  <input type="button" class="prev" onclick="xuly(false)" value="prev"></input>
  <input type="button" class="next" onclick="xuly(true)" value="next"></input>
  <br>
  <label class="current_page">Ảnh 1 / 20</label>
  <form id="form">
</form>
```

1. Tạo hiển thị hình ảnh bằng thẻ img
2. Tạo 2 button "prev", "next", đặt event của button là `onclick = xuly(...)`
3. Tạo Label thể hiện mã số ảnh hiện tại.

Code Javascript:

```
//hàm tạo vòng lặp tự động bằng thời gian
var theCountdownDate = new Date().getTime(); // thời gian hiện tại
var game_turn = 0;
var theDistance; // chênh lệch sử dụng trong hàm setInterval

var theInterValsFunction = setInterval( function(){
  var theNowTimeDate = new Date().getTime();
  theDistance = theNowTimeDate - theCountdownDate;
  if (theDistance - 10000*game_turn >= 10000){
    game_turn += 1;
    xuly(true);
  }
})
```

Các chuỗi lệnh để tạo 1 vòng lặp chạy hàm theo thời gian:

1. Dùng hàm `Date().getTime()`; để lấy thời gian hiện tại từ hệ thống.
2. `setInterval` dùng để thiết lập một hàm nào đó sẽ được thực thi sau một khoảng thời gian xác định và hành động này được lặp lại mãi mãi.
3. Điều kiện `if (theDistance - 10000*game_turn >= 10000)`:
 - a. Do hàm `setInterval` được đếm theo millisecond, nên `10000` tương ứng với 10s.
 - b. Cứ sau 10s chương trình sẽ chạy hàm `xuly()` để qua ảnh tiếp theo.

```
function xuly(dieuhuong){
    if (dieuhuong == true && current < max_number){
        current++;
        // bộ đếm trở về 0
        game_turn=0;
        theCountdownDate= new Date().getTime();
    }
    else if (dieuhuong == true && current == max_number){
        current = 1;
        // bộ đếm trở về 0
        game_turn=0;
        theCountdownDate= new Date().getTime();
    }
    else if (dieuhuong == false && current > 1) {
        current--;
        // bộ đếm trở về 0
        game_turn=0;
        theCountdownDate= new Date().getTime();
    }
    document.querySelector(".current_page").innerHTML = "Ảnh " + current.toString() + " / " + max_number.toSt
    //thay hình theo biến current
    document.querySelector("#main_img").src = "images/"+ (current%5+1).toString() + ".jpg";
}
```

Phân tích Hàm xuly():

- Hàm xuly() xử dụng để tăng mã số ảnh hoặc giảm mã số ảnh:
 - Nếu tham số của hàm xuly() là **true**: mã số tăng 1, chuyển qua ảnh tiếp theo.
 - Nếu tham số của hàm xuly() là **false**: mã số giảm 1, trở về ảnh trước đó.
- Điều kiện khi mã số ảnh tối đa sẽ trở về 1: **current == max_number**
- Cuối cùng, dùng hàm **toString()** để chuyển **current** từ dạng số sang chuỗi để hiển thị trên Label (được tạo trên html)

Bài tập 2

Code html:

```
<form onsubmit="kiem_tra(); return false;" method="POST">

  <label for="fname">Mã Sinh Viên:</label>
  <input class="form-control" onkeyup="changeColor(this)" type="text" id="ms_sv"
  placeholder="Nhập mã sinh viên">

  <label for="fname">Họ Tên:</label>
  <input class="form-control" onkeyup="changeColor(this)" type="text" id="ten_sv"
  name="ten_sp" placeholder="Nhập tên" >

  <div class="form-group">
    <label for="exampleInputEmail1">Email: </label>
    <input type="email" class="form-control" id="email_sv" onkeyup="changeColor(this)"
    aria-describedby="emailHelp" placeholder="Nhập email">
  </div>
```

1. Tạo các dòng nhập dữ liệu bằng thẻ `input`.
2. Email sử dụng `type="email"` để hỗ trợ kiểm tra khi người dùng nhập email hợp lệ.

```
<label for="fname">Giới Tính:</label>
<div id="gioi_tinh" onchange="changeColor(this)">
  <div class="alert" style="border: 1px solid #E0E0E0;">
    <input type="radio" name="GT" aria-label="Radio button for following text input">&nbsp;Nam&nbsp;
    <input type="radio" name="GT" aria-label="Radio button for following text input">&nbsp;Nữ
  </div>
</div>

<label for="fname">Sở Thích:</label>
<div id="so_thich" onchange="changeColor(this)">
  <div class="alert" style="border: 1px solid #E0E0E0;">
    <input type="checkbox" name="ST1" aria-label="Radio button for following text input">&nbsp;Đọc sách&nbsp;
    <input type="checkbox" name="ST2" aria-label="Radio button for following text input">&nbsp;Du lịch&nbsp;
    <input type="checkbox" name="ST3" aria-label="Radio button for following text input">&nbsp;Ăn thực&nbsp;
    <input type="checkbox" name="ST4" aria-label="Radio button for following text input">&nbsp;Âm nhạc&nbsp;
    <input type="checkbox" name="ST5" aria-label="Radio button for following text input">&nbsp;Khác
  </div>
</div>

<label for="cellphone">Quốc Tịch:</label>
<select class="form-control" name="national" id="choose" onchange="changeColor(this)">
  <option value="" disabled selected>Chọn Quốc Tịch</option>
  <option value="vie">Việt Nam</option>
  <option value="lao">Lào</option>
  <option value="cam">Campuchia</option>
  <option value="usa">Mỹ</option>
</select>
```

3. Giới tính sử dụng thẻ radio với `name="GT"`
4. Sở thích sử dụng thẻ checkbox với `name="ST"`
5. Quốc tịch sử dụng thẻ `select`:
 - a. "Chọn quốc tịch" sẽ đặt ở trạng thái **Disabled**, để không thể chọn từ người dùng.
 - b. mỗi lựa chọn được đặt value tương ứng để nhận biết trong javascript.

```
<label for="price">Ghi Chú:</label>
<textarea class="form-control" id="ghi_chu" onkeyup="changeColor(this)"></textarea><br>
<input type="submit" class="btn btn-danger" value="Đăng Ký" >
```

6. Ghi chú sử dụng thẻ `textarea`.
7. Nút submit ở cuối form để chạy hàm `onsubmit=kiemtra()`;

Code Javascript:

```
var kt;
var ms_sv = document.querySelector('#ms_sv');
if (ms_sv.value.length == 0){
    ms_sv.style.backgroundColor = "yellow";
    ms_sv.placeholder = 'Vui lòng không bỏ trống Mã số sinh viên!'
    alert('Vui lòng không bỏ trống Mã số sinh viên!')
    kt = 0;
}
```

Đặt điều kiện nhập ô mã số sinh viên:

1. Dùng lệnh `Document.querySelector` để tìm kiếm Element `ms_sv`
2. Kiểm tra độ dài chuỗi bằng lệnh `length`, để xác định ô được nhập từ người dùng hay ô đang bỏ trống.
3. Nếu ô bỏ trống sẽ đặt giá trị `kt` là 0. Cuối function chúng ta sẽ thiết lập nếu `kt = 0` submit sẽ trả về `False`.

```
if (kt == 0){
    alert("Đăng ký thất bại!!")
}
else{
    alert("Đăng ký thành công!!")
}
```

Đặt điều kiện nhập tên sinh viên và email cũng thiết lập tương tự ô mã số sinh viên.

```
//Hàm kiểm tra radio
var radio = document.querySelectorAll('[name="GT"]');
var ischecked = false;
radio.forEach(function(a){
    if (a.checked){
        ischecked = true;
    }
})
if (ischecked == false){
    document.querySelector('#gioi_tinh').style.backgroundColor = "yellow";
    alert('Vui lòng chọn giới tính!')
    kt = 0
};
```

Đặt điều kiện chọn radio giới tính:

1. Dùng lệnh `document.querySelectorAll` để xác định tập hợp của các Element.
2. Dùng hàm `forEach()` để duyệt từng phần tử trong tập hợp.
3. Nếu tồn tại Element đã được chọn (`isChecked`) thì tập hợp radio hợp lệ, ngược lại, sẽ trả về `kt = 0`.

Đặt điều kiện chọn checkbox sở thích cũng thiết lập tương chọn radio giới tính.

```
//Hàm kiểm tra choose
//Tìm kiếm element dựa trên name của input
var choose = document.querySelector('#choose');
if (choose.value == ""){
    choose.style.backgroundColor = "yellow";
    alert('Vui lòng chọn Quốc tịch!')
    kt = 0
};
```

Đặt điều kiện thẻ select Quốc tịch:

1. Kiểm tra trực tiếp giá trị `value` của thẻ, nếu `value` đang để trống sẽ trả về `kt = 0`

```
//Hàm kiểm tra ghi chú
//Tìm kiếm element dựa trên id
var ghi_chu = document.querySelector('#ghi_chu');
if (ghi_chu.value.length >= 200){
    ghi_chu.style.backgroundColor = "yellow";
    alert('Ghi chú phải ít hơn 200 ký tự!')
    kt = 0
};
```

Đặt điều kiện kiểm tra cho ô nhập ghi chú:

1. Kiểm tra độ dài chuỗi bằng lệnh `length`, nếu độ dài lớn hơn `200` sẽ trả về `kt = 0`.

```
// Thêm vào nhập, check, màu sẽ dùng event onchange
function changeColor(a){
    a.style.backgroundColor = "";
}
```

Hàm `changeColor()` để thay đổi màu nền ô nhập của form trong html, khi ô nhập bị báo lỗi hoặc bỏ trống.

Bài tập 3

Code html:

```
<select id="muc_gia" class="col-2" onchange="xuly1()">
  <option value="" disabled selected>Mức giá</option>
  <option value="0">--Tất cả --</option>
  <option value="1">Dưới 100</option>
  <option value="2">100 - 200</option>
  <option value="3">200 - 400</option>
  <option value="4">400 trở lên</option>
</select>
```

Sử dụng thẻ `select` để tạo lựa chọn mức giá.

<input type="checkbox"/>	Hàng hóa	Đơn giá	Số lượng	Thành tiền
<input type="checkbox"/>	iPhone	700	<input type="text"/>	
<input type="checkbox"/>	Sam Sung	400	<input type="text"/>	
<input type="checkbox"/>	Nokia	100	<input type="text"/>	
<input type="checkbox"/>	Sony Xperia	450	<input type="text"/>	
<input type="checkbox"/>	Motorola	180	<input type="text"/>	
<input type="checkbox"/>	Oppo	600	<input type="text"/>	
<input type="checkbox"/>	bPhone	90	<input type="text"/>	
TỔNG				

Vẽ table trên html, dùng thẻ `<input type="checkbox">` cho mỗi cột đầu của row.

Code Javascript:

```
//xác định mức giá được lựa chọn
var muc_gia = document.querySelector('#muc_gia').value
//chọn ra list sản phẩm theo class don_gia
var sp = document.querySelectorAll('.don_gia')
```

Tạo hàm `xuly1()` để kiểm tra hiển thị sản phẩm theo mức giá.

1. Dùng `document.querySelector` và `document.querySelectorAll` để xác định Element mức giá được chọn và danh sách Element sản phẩm.

```
//ẩn sản phẩm theo điều kiện
if (muc_gia == 1){
    sp.forEach(function(item){
        if(item.innerHTML >= 100){
            var san_pham = item.parentElement
            san_pham.style.display = "none";
            xoa(san_pham)
        }
    })
}
```

Ẩn sản phẩm theo mức giá được chọn lựa:

1. Mức giá 1: Ẩn các Element sản phẩm có giá trị ≥ 100 (USD) , dùng lệnh `innerHTML` để xác định đơn giá của sản phẩm. Các mức giá khác cũng thực hiện tương tự.
2. Sử dụng lệnh `san_pham.style.display = "none";` để ẩn Element không phù hợp.

```
function chon(box){
    var box2 = box.parentElement;
    var san_pham = box2.parentElement;
    if (box.checked) {
        san_pham.querySelector('.so_luong').disabled = false;
    } else {
        xoa(san_pham)
        san_pham.querySelector('.so_luong').disabled = true;
    }
}
```

Các Element sản phẩm ban đầu được thiết lập ở trạng thái `Disabled` để tránh người dùng nhập vào khi chưa tích vào ô checkbox, cho nên hàm `chon(box)` để kiểm soát trạng thái của các dòng sản phẩm này, chỉ khi được tích vào check box, người dùng mới có thể thêm số lượng sản phẩm.

```
function xuly2(sp, soluong){
    // dùng parentElement để lấy dữ liệu row
    san_pham = sp.parentElement
    don_gia = san_pham.querySelector('.don_gia').innerHTML
    thanh_tien = san_pham.querySelector('.thanh_tien')

    if (!isNaN(soluong)){
        thanh_tien.value = don_gia * soluong
        thanh_tien.innerHTML = thanh_tien.value
        console.log(thanh_tien)
        tinhhtong()
    }
}
```

Khi số lượng được nhập, hàm `xuly2()` sẽ tính thành tiền của mỗi sản phẩm.

```
function tinhTong(){
    var tong = 0;
    var tt = document.querySelectorAll('.thanh_tien')
    tt.forEach(function(thanh_tien){
        if (!isNaN(thanh_tien.value)){
            tong += thanh_tien.value
        }
    })
    document.querySelector('#tong').innerHTML = tong
}
```

Khi có 1 hoặc nhiều hơn 1 sản phẩm được tính thành tiền, hàm `tinhTong()` sẽ cộng tất cả giá trị ô thành tiền.

```
function xoa(san_pham){
    san_pham.querySelector('.so_luong').value = 0
    san_pham.querySelector('.thanh_tien').value = 0
    san_pham.querySelector('.thanh_tien').innerHTML = 0
    tinhTong()
}
```

Khi một ô được tích nhưng lại bỏ chọn, hàm `xoa()` sẽ làm mới lại các dòng sản phẩm thành giá trị 0.

```
function chon_tat_ca(a){
    var boxs = document.querySelectorAll('.box')
    if (a.checked) {
        boxs.forEach(function(box){
            box.checked = true;
            chon(box)
        })
    } else {
        boxs.forEach(function(box){
            box.checked = false;
            var box2 = box.parentElement;
            var san_pham = box2.parentElement;
            xoa(san_pham)
            san_pham.querySelector('.so_luong').disabled = true;
        })
    }
}
```

1. Hàm `chon_tat_ca()` được chạy khi người dùng tích vào ô check all, cả dòng sản phẩm sẽ được hủy bỏ trạng thái `Disabled` để người dùng có thể nhập giá trị và tính thành tiền.
2. Sử dụng `parentElement` xác định sản phẩm (row của table) dựa trên giá trị ô (cell của table).