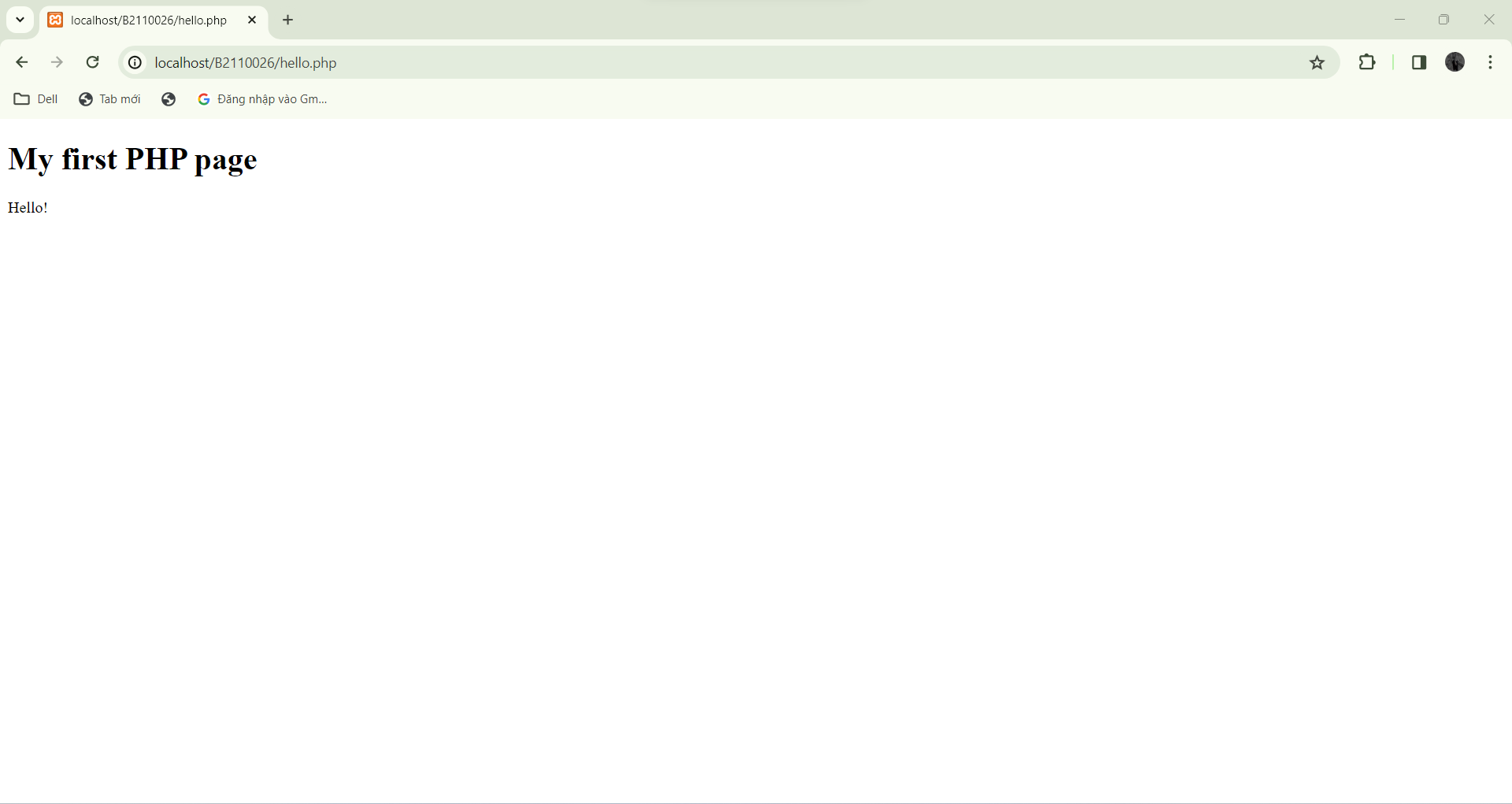
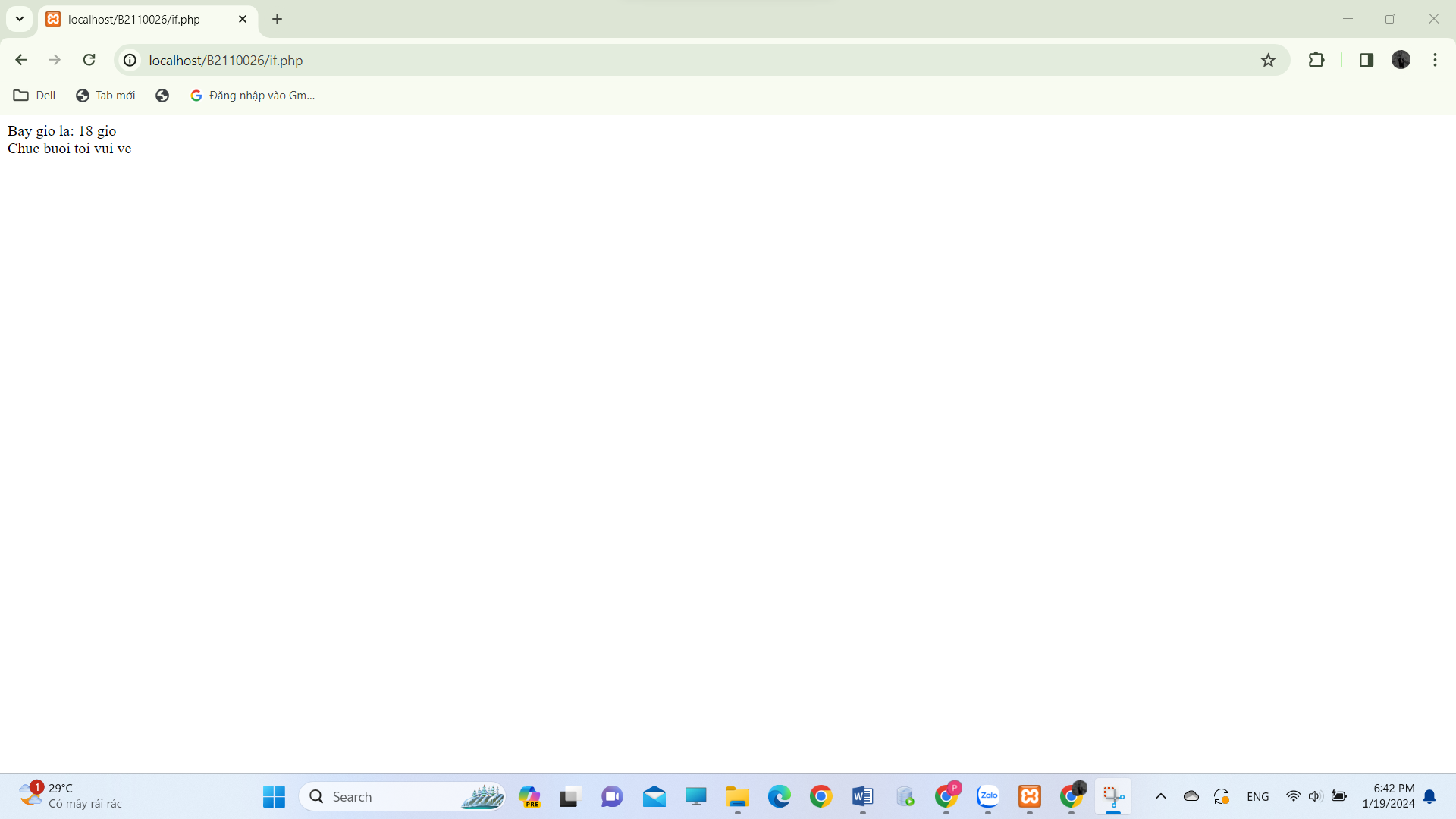
**YÊU CẦU BÀI THỰC HÀNH**

**1. Bạn hãy chạy tất cả các lệnh hướng dẫn ở trên và chụp lại màn hình kết quả.**

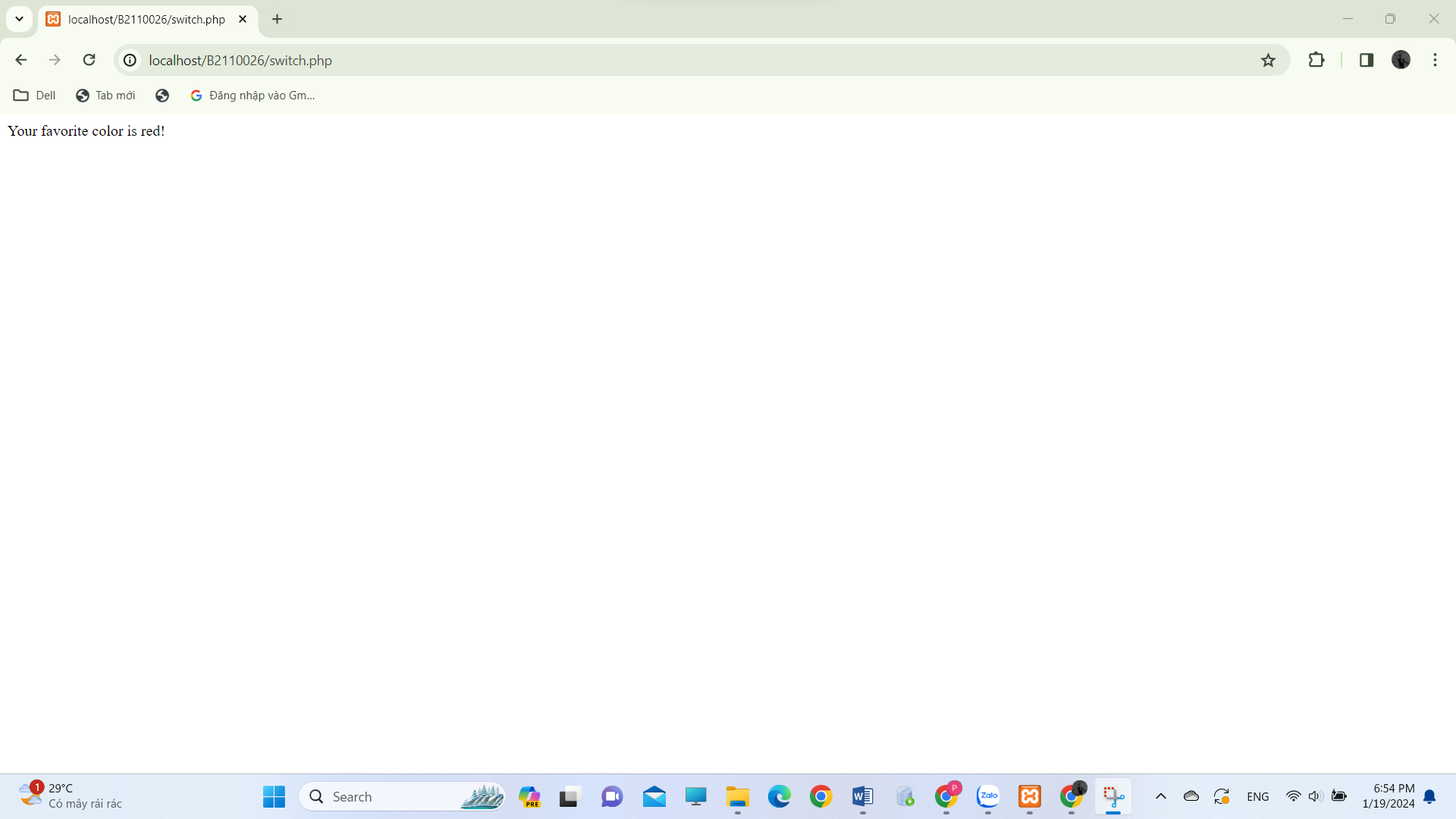
*hello.php*



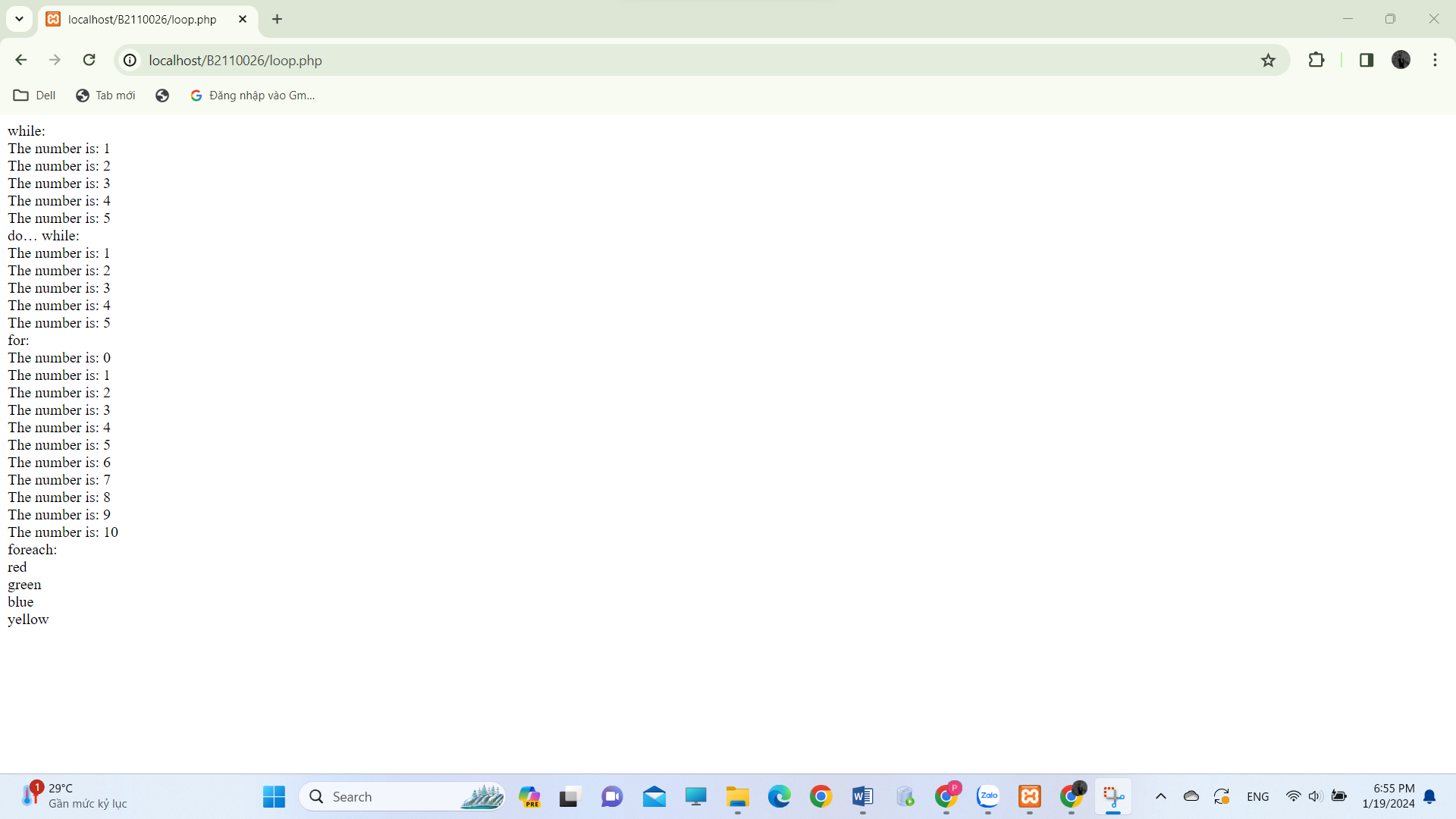
*if.php*

**

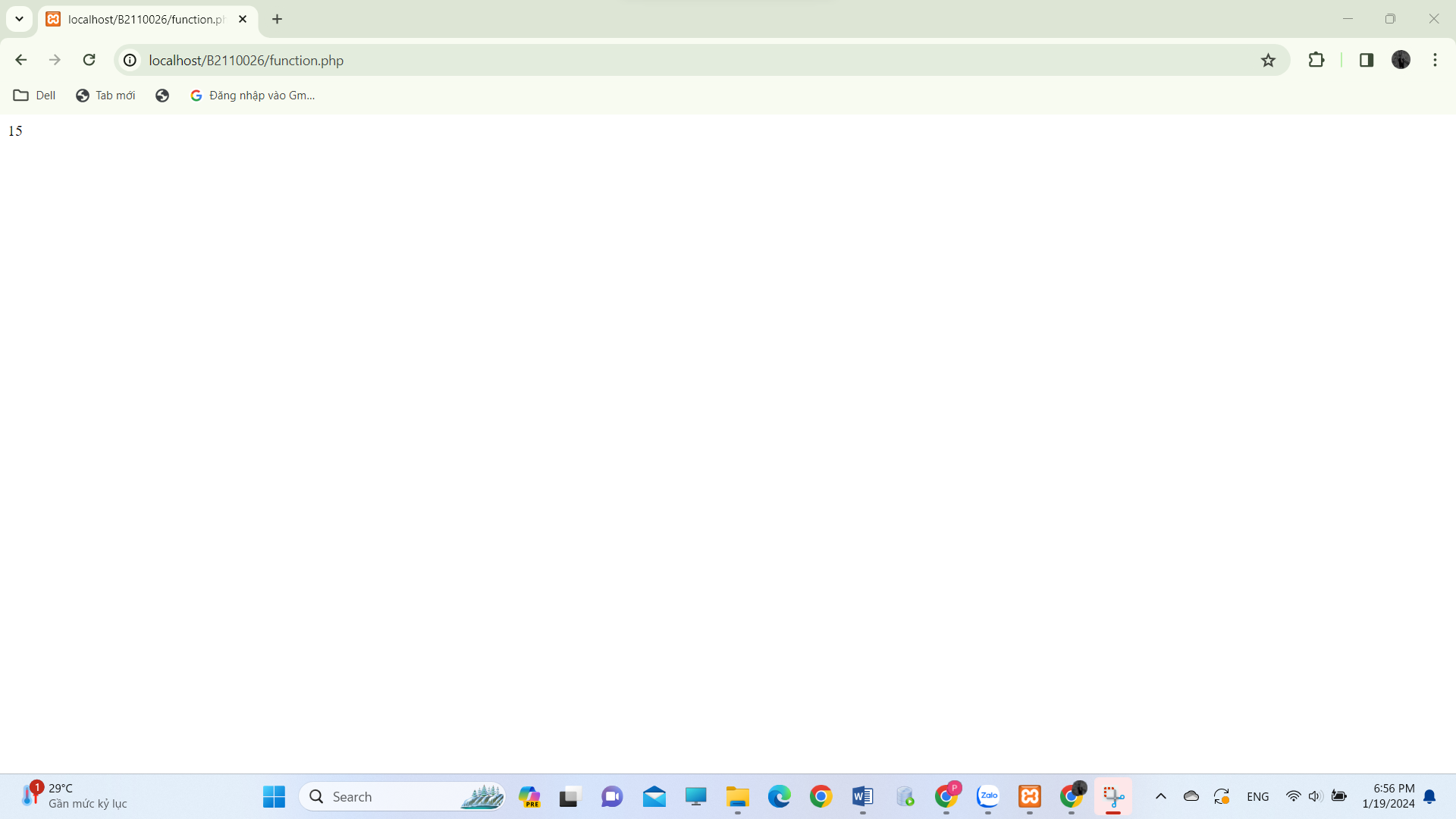
*switch.php*

**

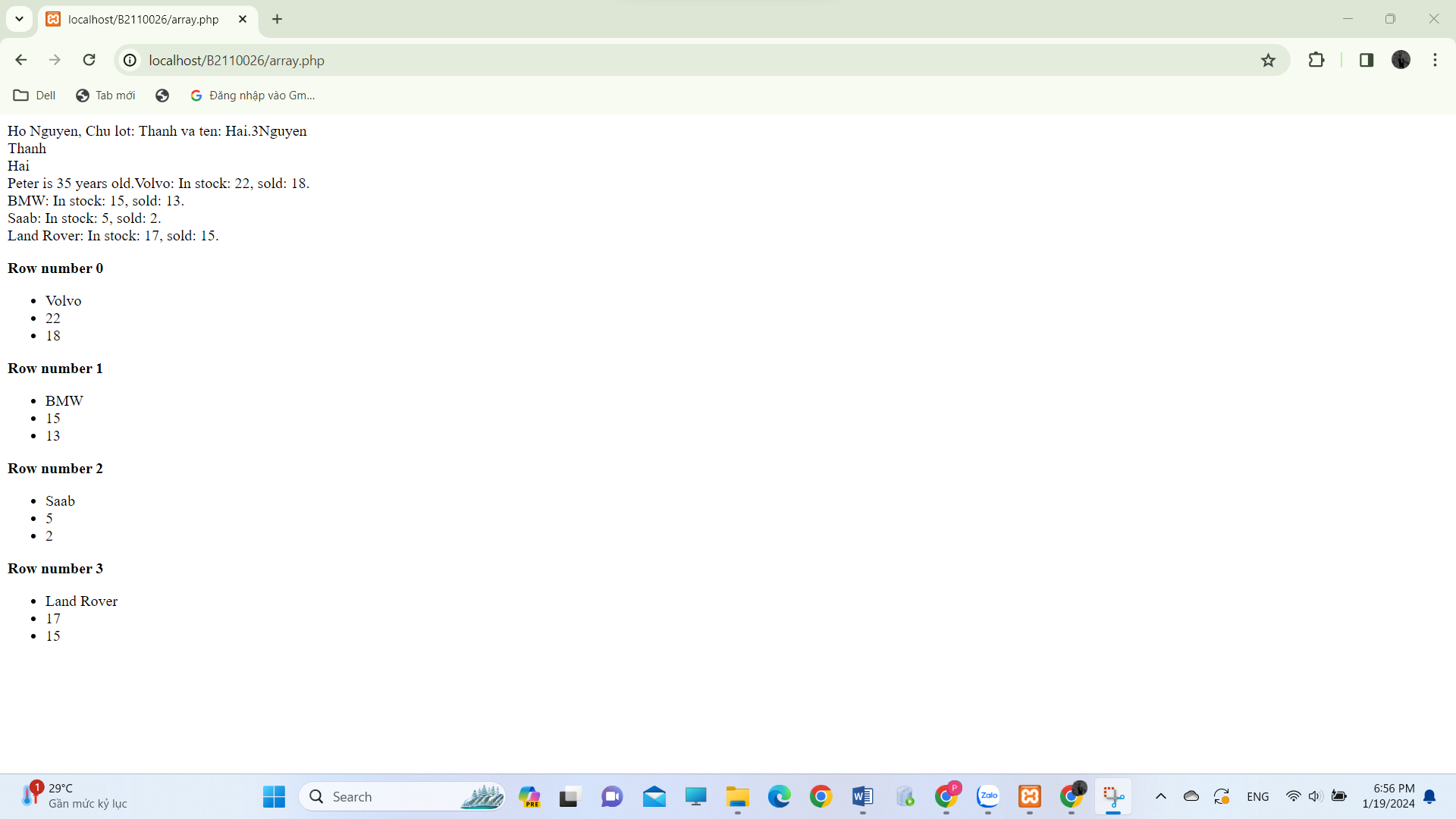
*loop.php*

**

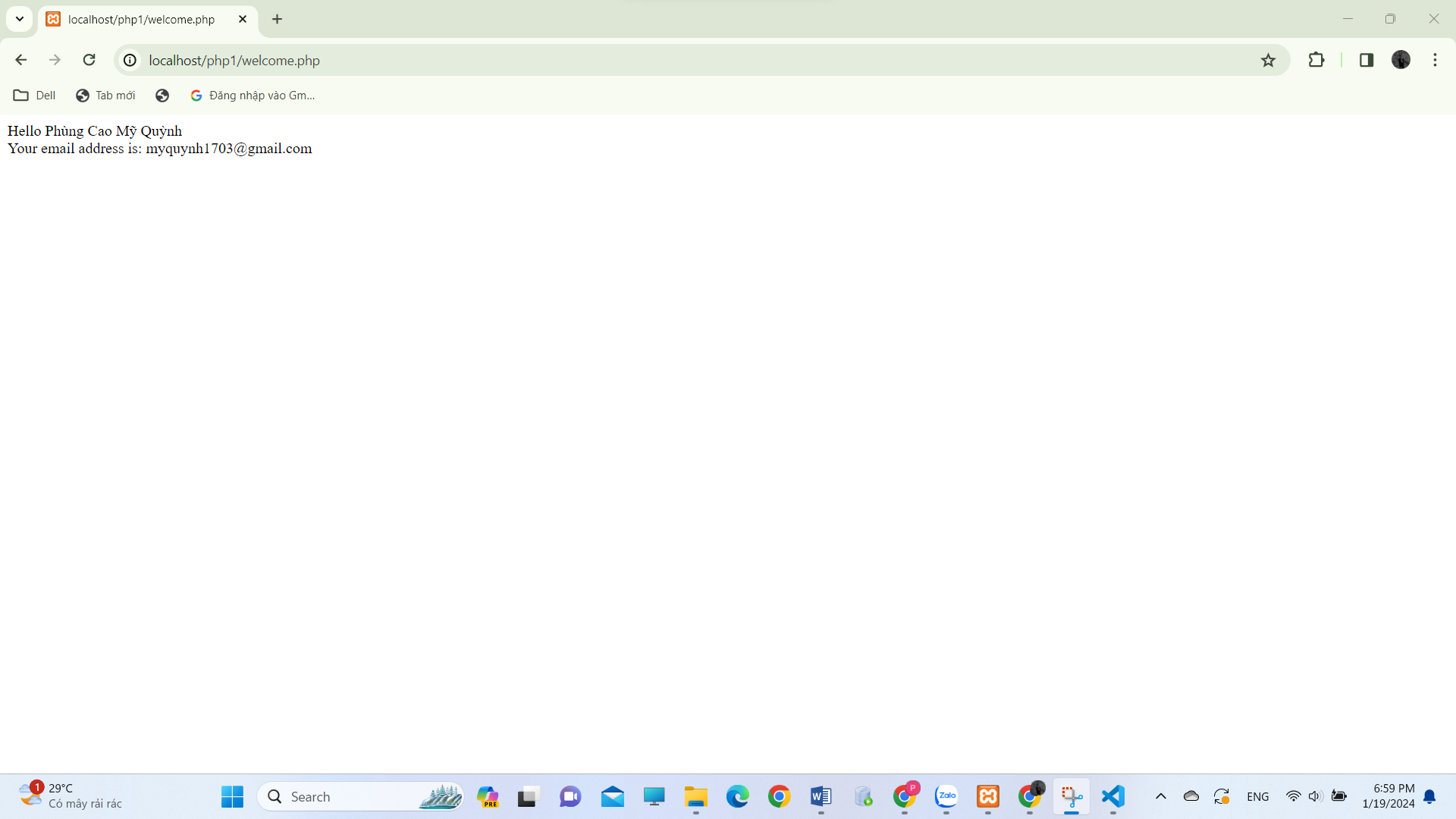
*function.php*

**

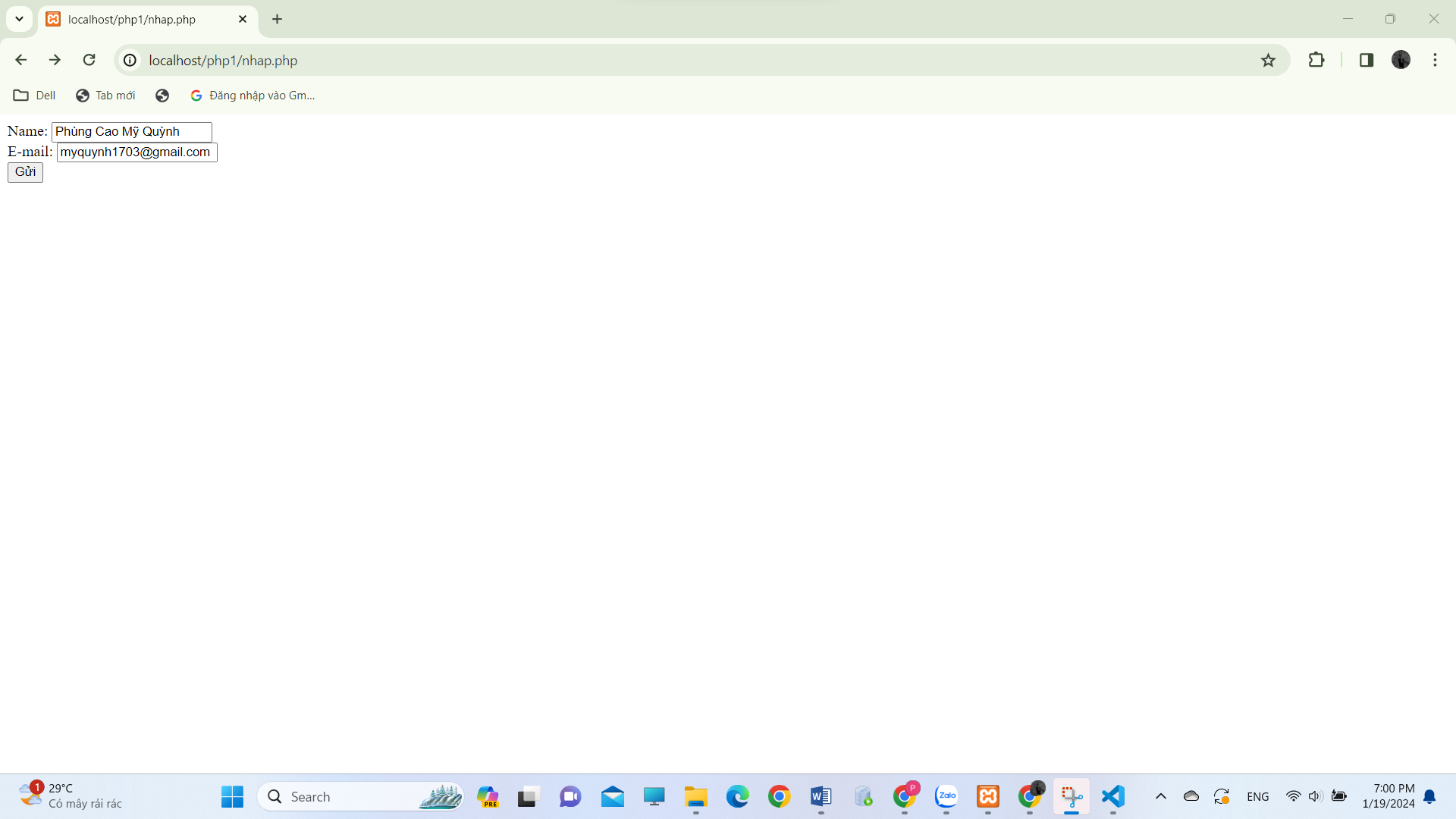
*array.php*

**

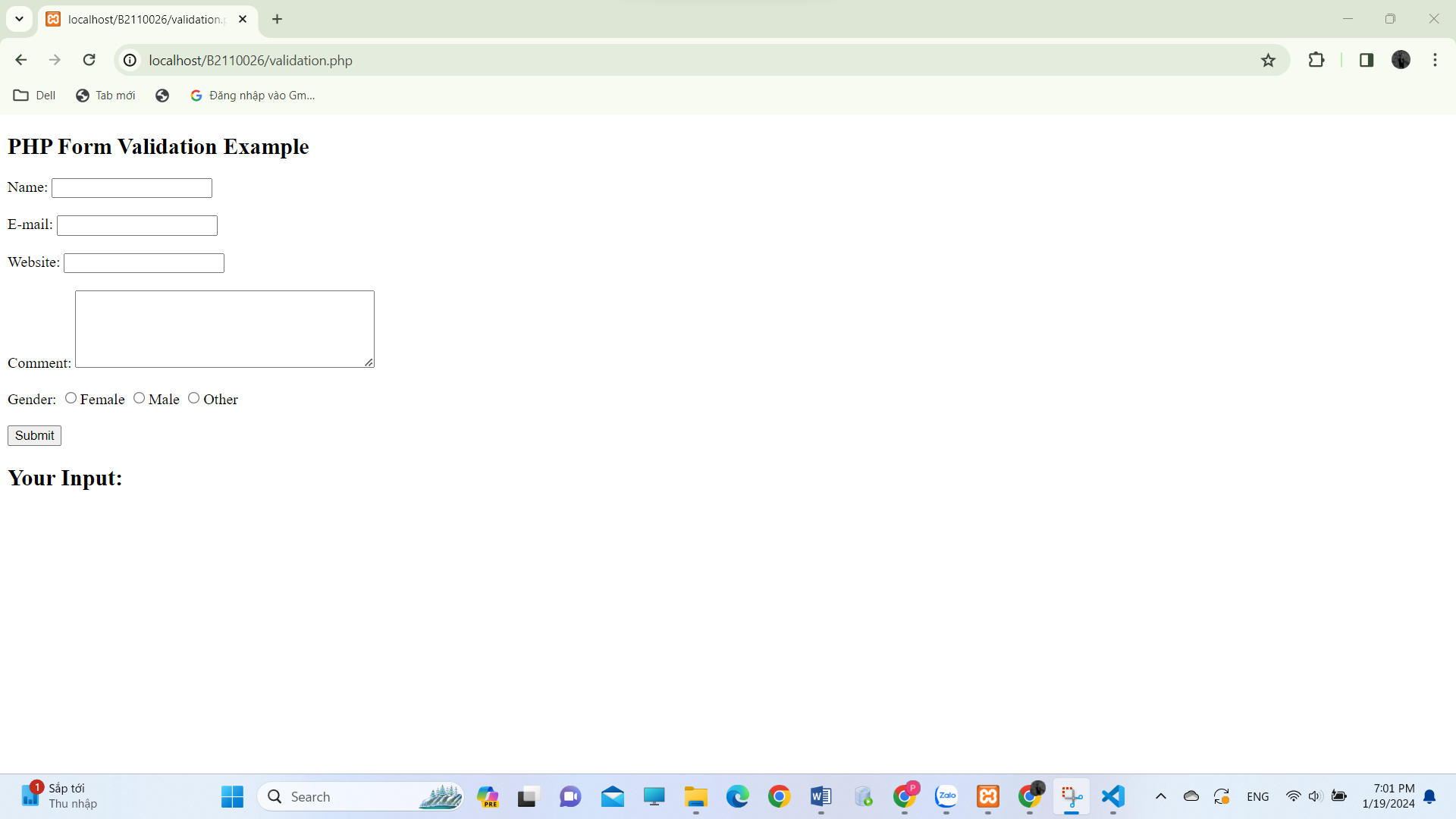
*welcome.php*

**

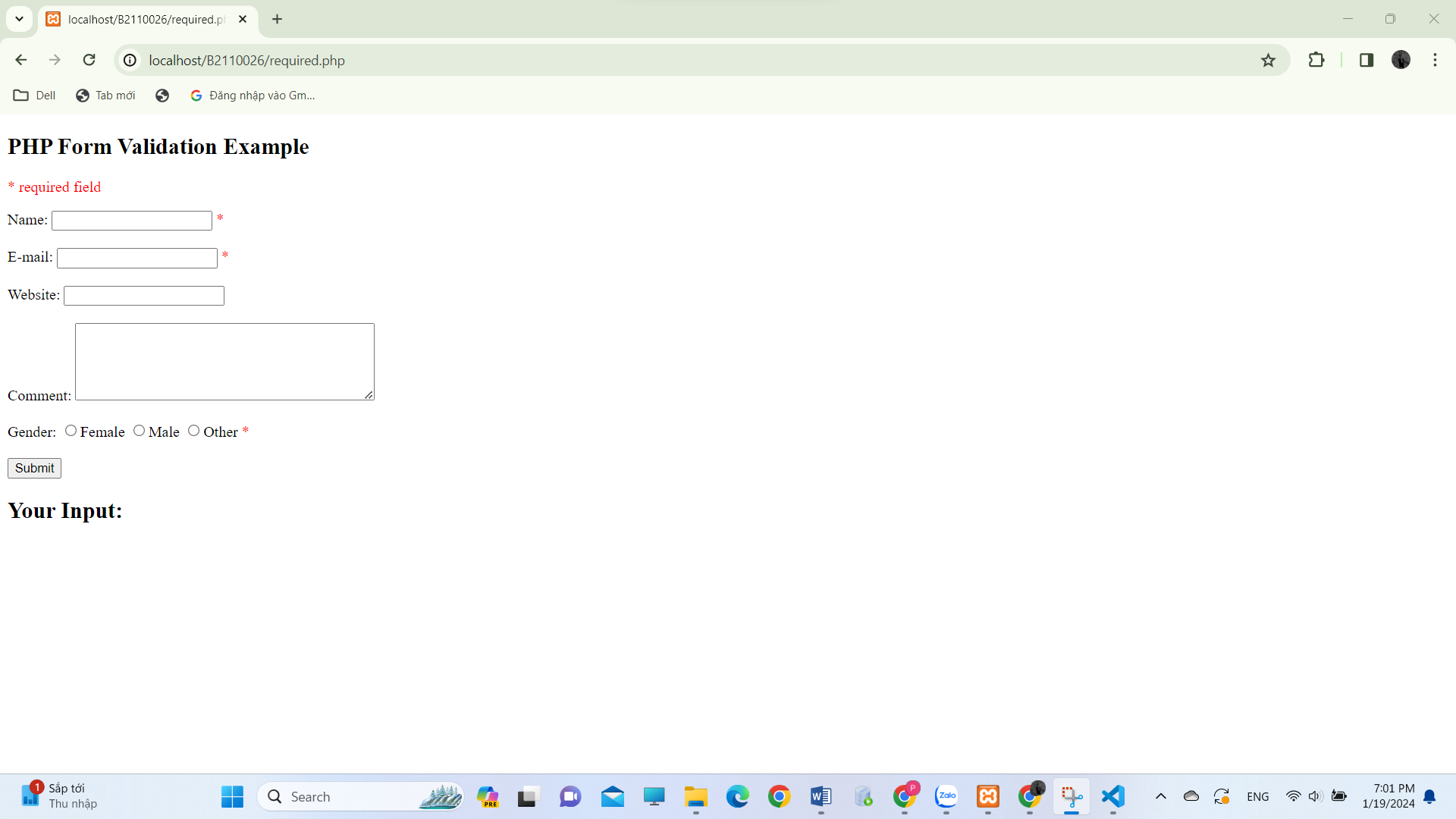
*nhap.php*

**

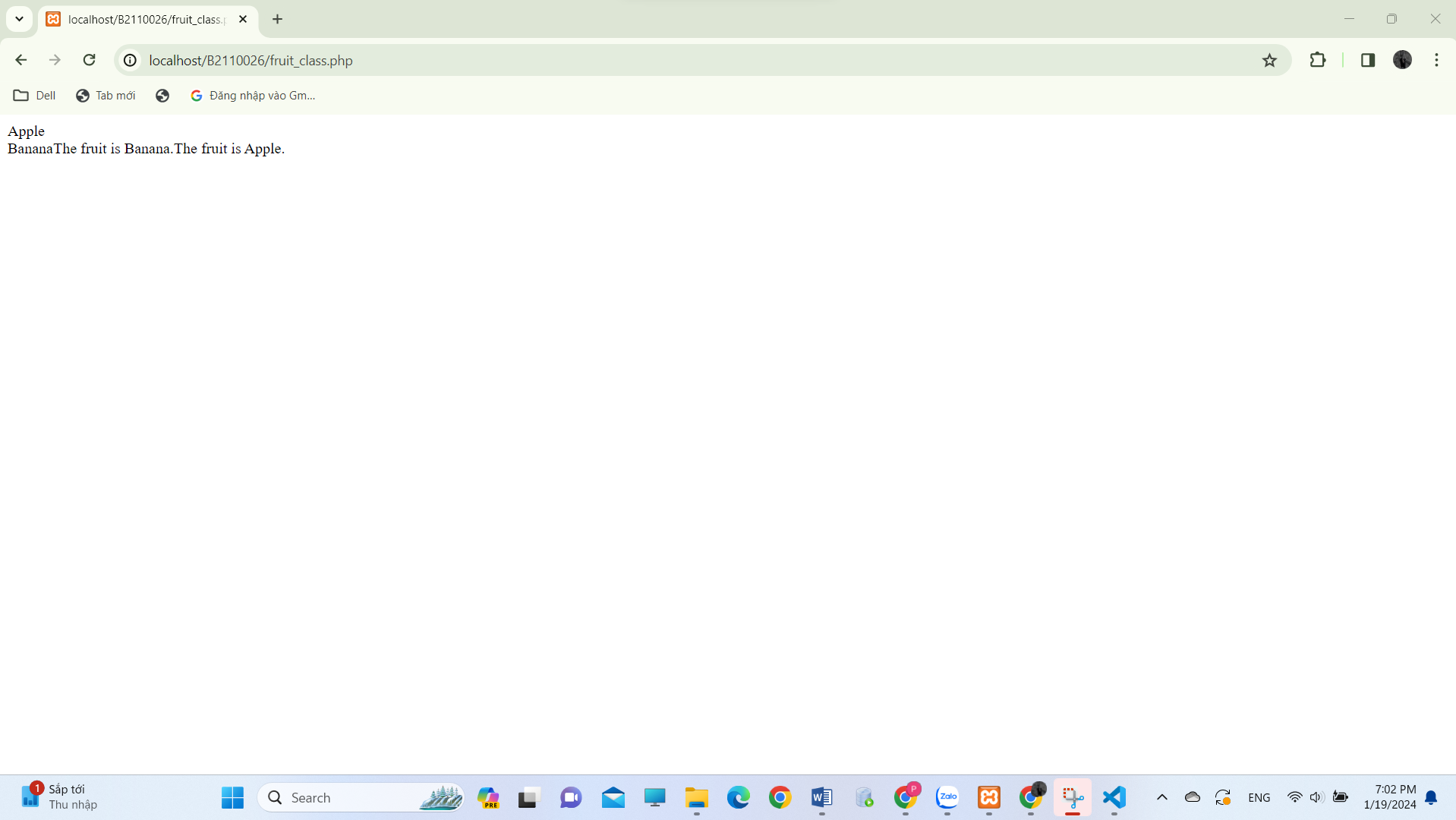
*validation.php*

**

*required.php*

**

*fruit\_class.php*



**2. Hãy cho biết cách khai báo biến, và gán trị cho biến trong PHP. Liệt kê các chú ý khi đặt tên biến. Cho các ví dụ về khai báo biến, và gán giá trị kiểu chuỗi, kiểu số, ngày tháng.**

- Khai báo biến:

Cú pháp: $ten\_bien

- Gán giá trị cho biến

Trong PHP, khi khai báo biến không cần ghi kiểu mà nó tự xác định kiểu dữ liệu cho biến khi biến được gán giá trị

- Các chú ý khi đặt tên biến

* Bắt đầu bằng ký tự **$**, theo sau là một ký tự hoặc dấu **\_**, tiếp đó là ký tự, ký tự số hoặc dấu **\_**
* Nên khởi tạo giá trị ban đầu cho biến
* Không trùng với tên hàm
* Không nên bắt đầu bằng ký số

\*Lưu ý: Tên biến có phân biệt chữ HOA – chữ thường

- Ví dụ

* Khai báo và gán giá trị kiểu chuỗi: **$ho\_ten = "John Doe";**
* Khai báo và gán giá trị kiểu số: **$tuoi = 20;**
* Khai báo và gán giá trị ngày tháng:

**$ngay\_sinh = "2003-03-17"; // Chuỗi ngày tháng**

**$ngay\_sinh = strtotime($ngay\_sinh); // Chuyển đổi thành timestamp**

**3. Liệt kê ký hiệu để biết bắt đầu và kết thúc khối lệnh PHP.**

Trong PHP, các khối mã lệnh được đặt bên trong cặp dấu **‘ { } ’**. Dấu mở **‘ { ‘** đánh dấu bắt đầu của một khối lệnh và dấu đóng **‘ } ‘** đánh dấu kết thúc của khối lệnh.

**4. Liệt kê các ký hiệu có thể được dùng để tạo comment (ghi chú) trong khối lệnh PHP.**

- Comment dạng một dòng:

* Sử dụng dấu gạch chéo **` // `** bắt đầu một comment và comment này chỉ có hiệu lực trên một dòng.

- Comment dạng nhiều dòng:

* Sử dụng dấu **‘ \*/ ‘** để bắt đầu và dấu **‘ \*/ ‘** để kết thúc một comment. Mọi nội dung ở giữa này sẽ được gọi là comment.

**5. Cho biết có bao nhiêu loại khối lệnh để thực hiện vòng lặp. Cho biết sự khác biệt của foreach với các loại khối lệnh vòng lặp khác. Cho biết cách dùng của lệnh “continue” và lệnh “break” trong các vòng lặp.**

*- Trong PHP, có ba khối lệnh được đặt trong vòng lặp:*

* `**for` loop**: Sử dụng để lặp qua một dãy số cố định
* **`while` – loop**: Sử dụng để lặp với một điều kiện kiểm tra
* **`do - while` loop**: Giống như `while` loop, nhưng kiểm tra điều kiện sau mỗi vòng lặp

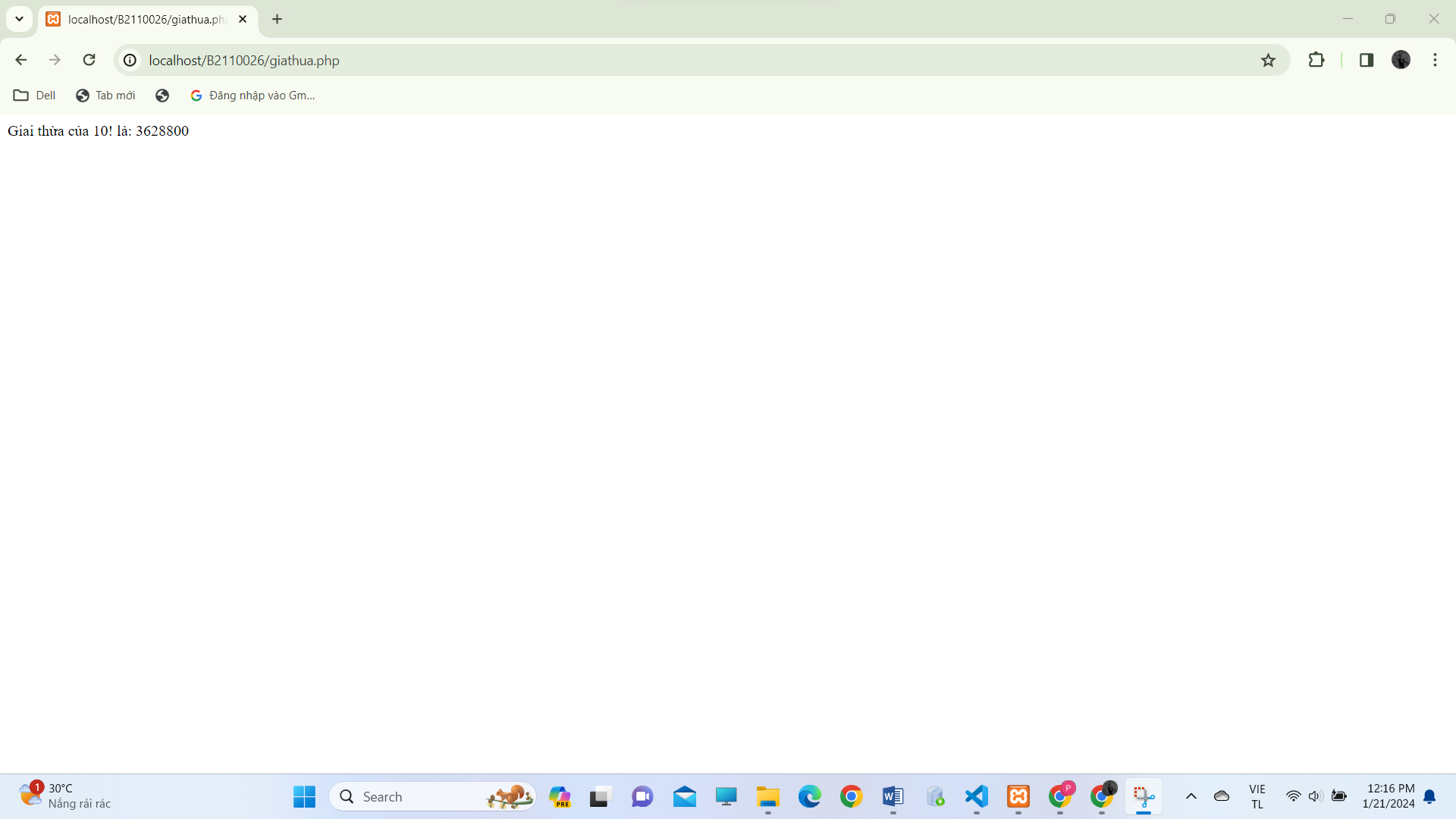
*- Sự khác biệt của `foreach` với các loại khối lệnh vòng lặp khác:*

**`foreach`** được sử dụng để lặp qua một mảng hay đối tượng. Nó tự động duyệt qua từng phần tử của mảng hoặc các thành phần của đối tượng mà không cần sử dụng chỉ số

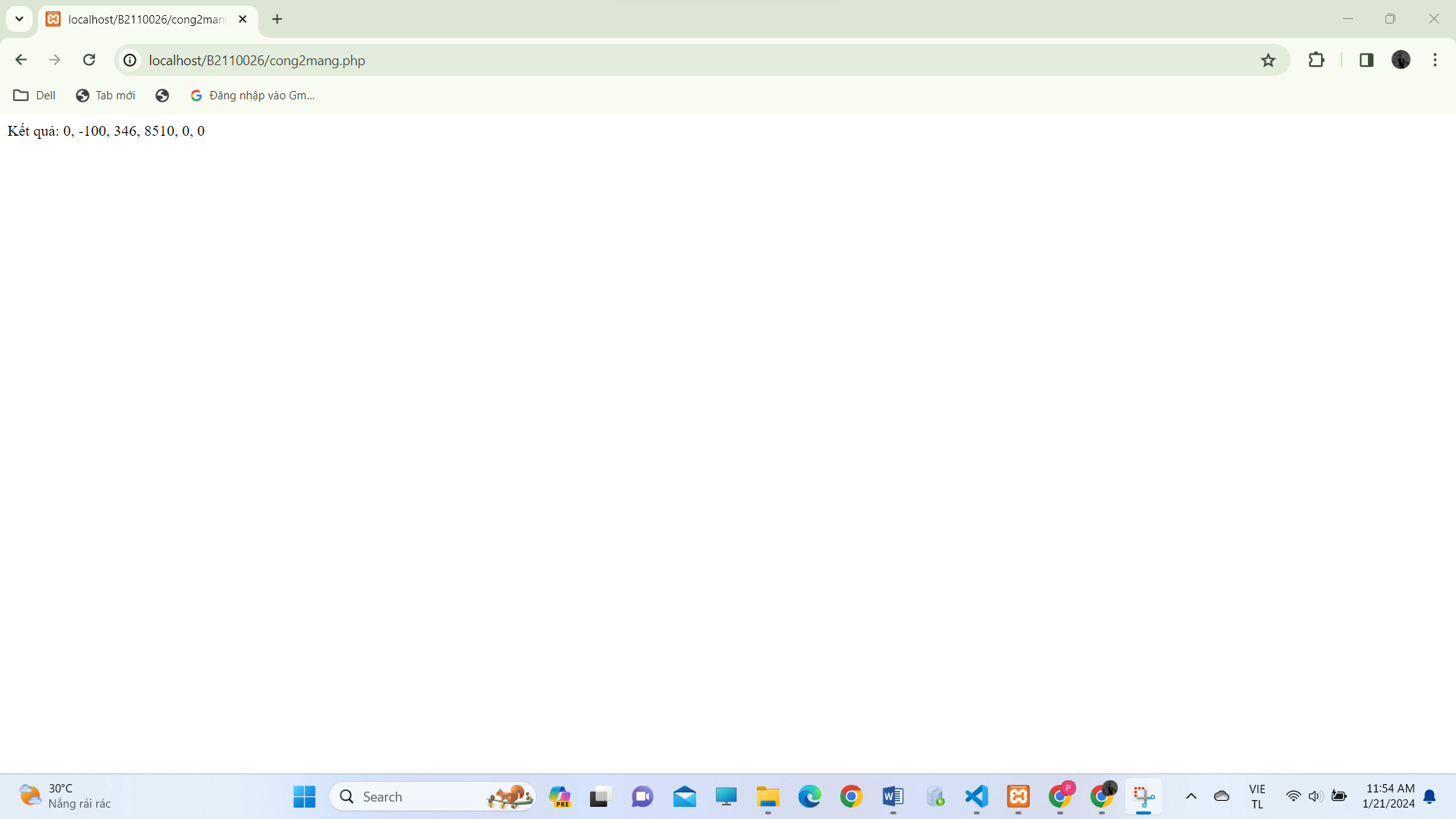
*- Cách sử dụng `continue` và `break` trong các vòng lặp:*

* **`continue`**: Dùng để bỏ phần còn lại của vòng lặp và chuyển đến vòng lặp tiếp theo
* **`break`**: Dùng để thoát khỏi vòng lặp ngay lập tức khi nó được gọi.

**6. Viết 1 tập tin php tên giaithua.php, định nghĩa hàm giai thừa và chạy thử với 10!**

****

**7.Viết 1 tập tin cong2mang.php viết hàm xử lý để cộng được 2 mảng có cùng độ dài, nếu 2 mảng không cùng độ dài thì thông báo lỗi. Chạy thử hàm với 2 mảng sau: a={344,224,223,7737,9922,-828}, b={-344,-324,123,773,-9922,828} và cho biết kết quả hiển thị.**

****

**8. Chạy ví dụ tập tin validation.php, dựa vào link 2 , bạn hãy giải thích những bước diễn ra quá trình này.**

**Bước 1:** Đầu tiên, các biến được khai báo và gán giá trị rỗng.

**Bước 2**: Kiểm tra xem biểu mẫu đã được gửi đi hay chưa bằng cách sử dụng $\_SERVER["REQUEST\_METHOD"].Nếu phương thức là “POST”, tức là biểu mẫu đã được gửi đi và tiến hành xử lý dữ liệu.

**Bước 3**: Dữ liệu từ biểu mẫu được lấy bằng cách sử dụng $\_POST và truyền qua hàm test\_input() để làm sạch

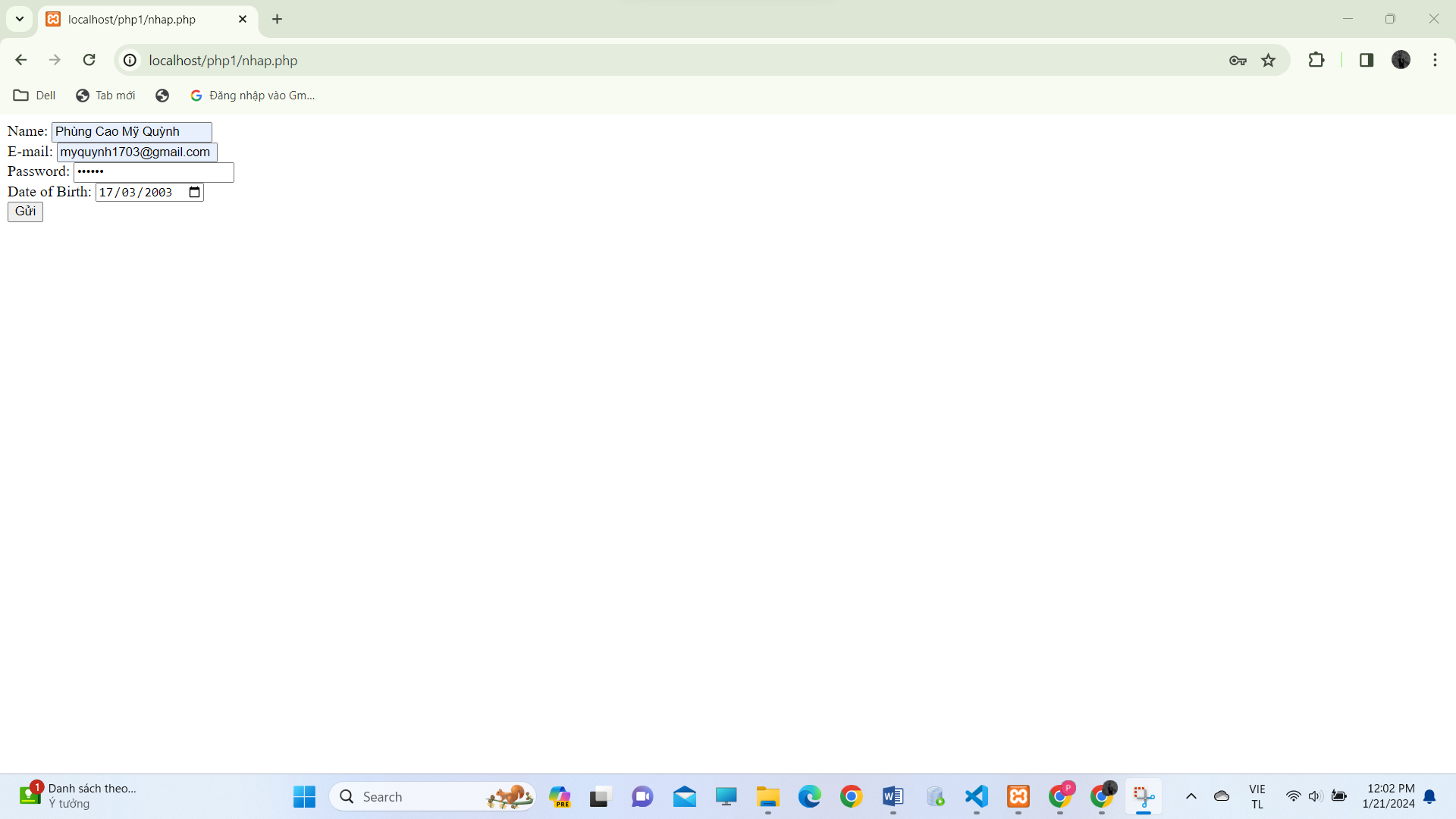
**Bước 4**: Sau khi xử lý dữ liệu, các giá trị được gán cho biến tương ứng

**Bước 5**: Một biểu mẫu HTML được hiển thị cho người dùng nhập dữ liệu. Khi người dùng nhấn nút “Submit”, biểu mẫu sẽ được gửi đi và quá trình xử lý sẽ được thực hiện lại.

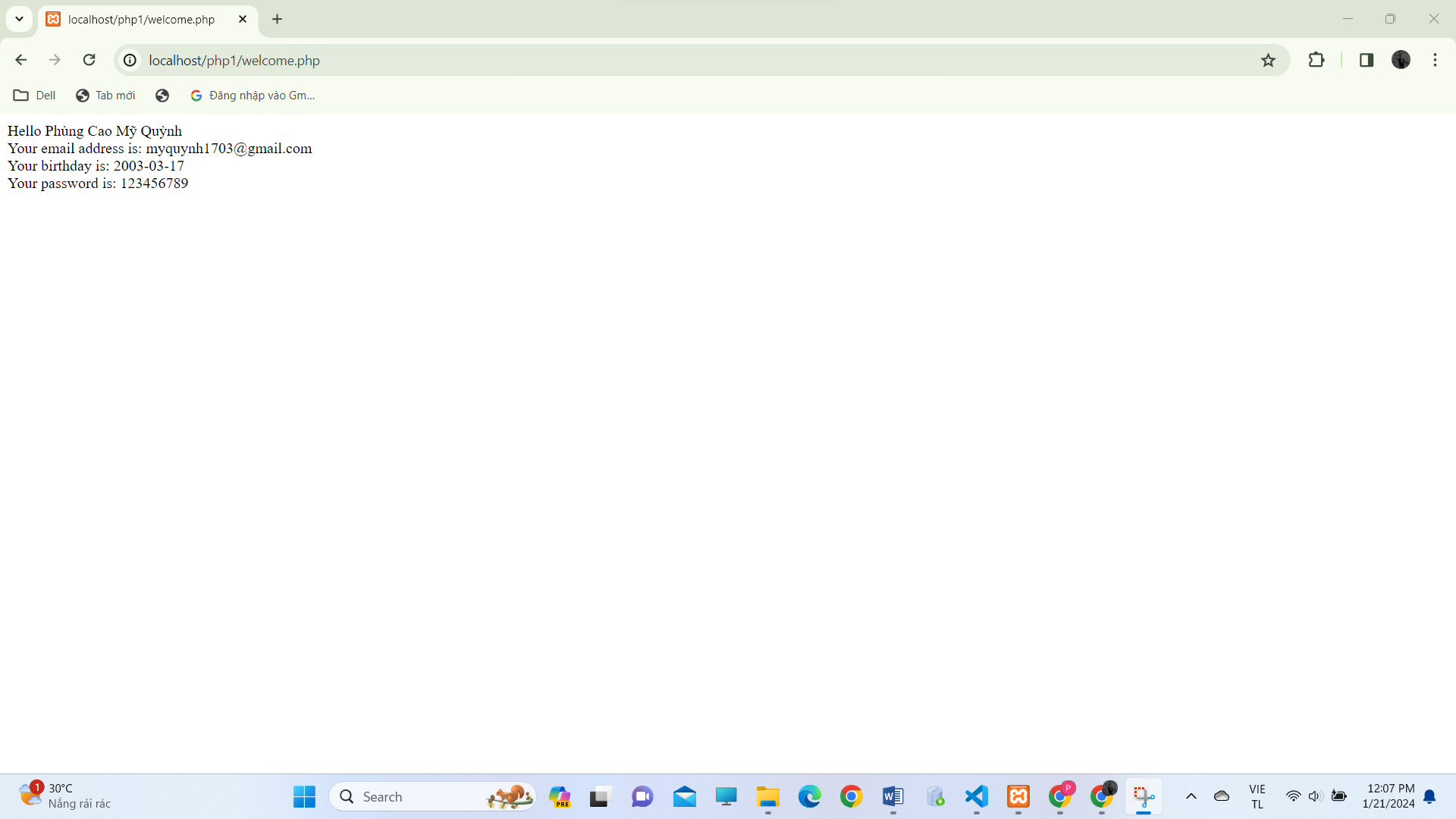
**Bước 6**: Cuối cùng, các giá trị được nhập từ biểu mẫu và đã được xử lý được hiển thi ra màn hình

**9. Hãy sửa lại tập tin nhap.php, thêm 1 ô để nhập mật khẩu (các ký tự nhập vào sẽ biến thành dấu \*), và 1 ô để nhập ngày tháng năm sinh của bạn. Sau khi nhập xong hiển thị tên, email và mật khẩu, ngày tháng năm sinh đã nhập. Chụp màn hình sau khi bạn đã nhập dữ liệu của chính bạn, chụp màn hình hiển thị dữ liệu nhập.**

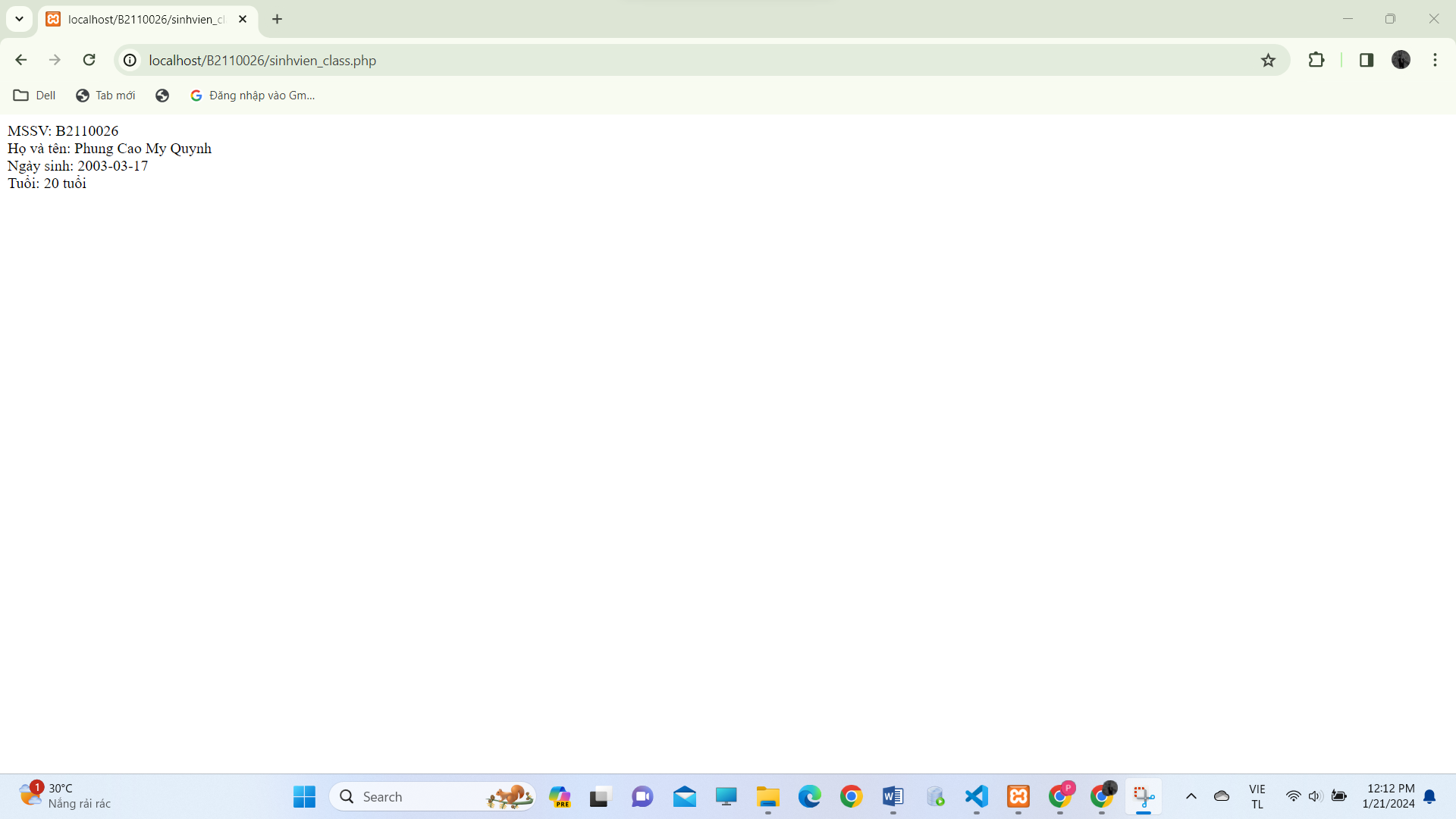
*nhap.php*



*welcome.php*



**10. Viết tập tin sinhvien\_class.php để khai báo 1 lớp sinhvien với các thuộc tính: mssv, hoten, ngaysinh với kiểu giá trị phù hợp. Viết các phương thức xây dựng, hủy, gán trị, trả về giá trị của các thuộc tính, viết hàm tính tuổi. Khai báo 1 biến sv có kiểu class sinhvien, gán các giá trị cần thiết với thông tin của bạn và thử tính tuổi dựa vào hàm vừa mới xây dựng trong class.**

****