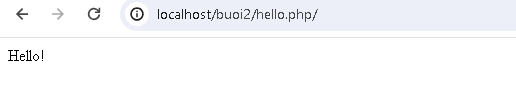
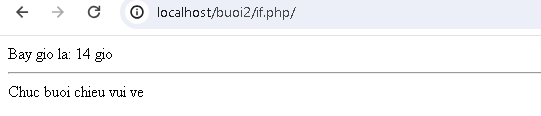
**1. Bạn hãy chạy tất cả các lệnh hướng dẫn ở trên và chụp lại màn hình kết quả.**

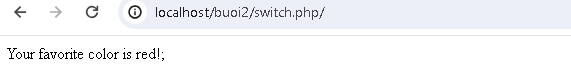
***-Hello.php***



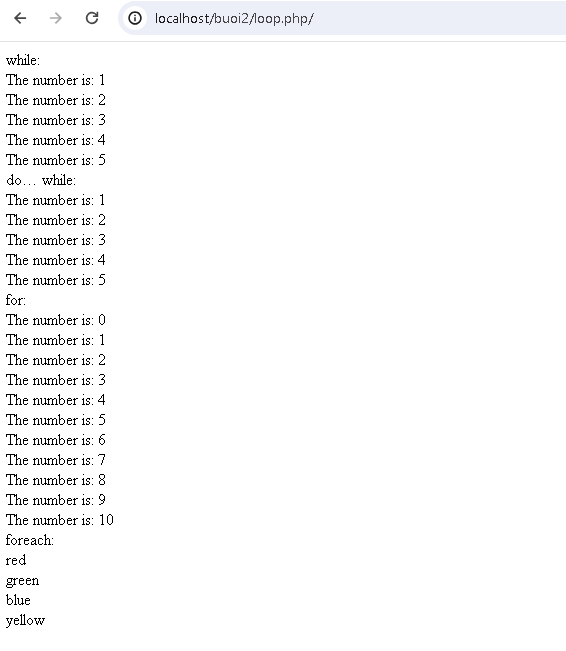
***-if.php***



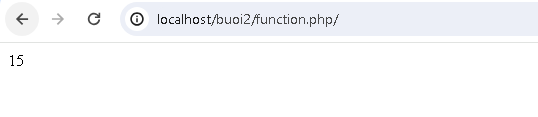
***-switch.php***



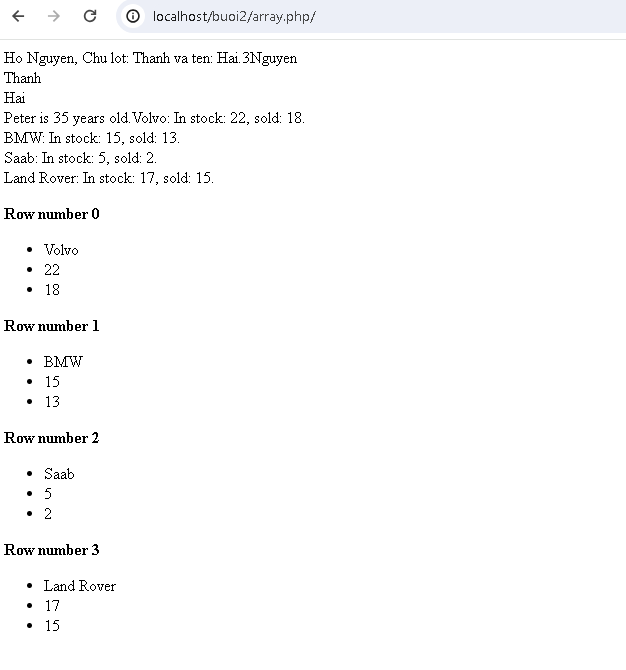
***-loop.php***



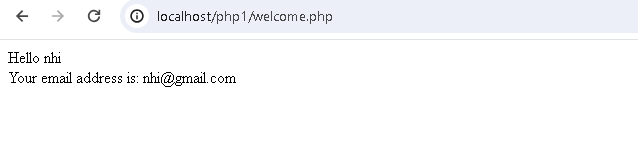
***-function.php***



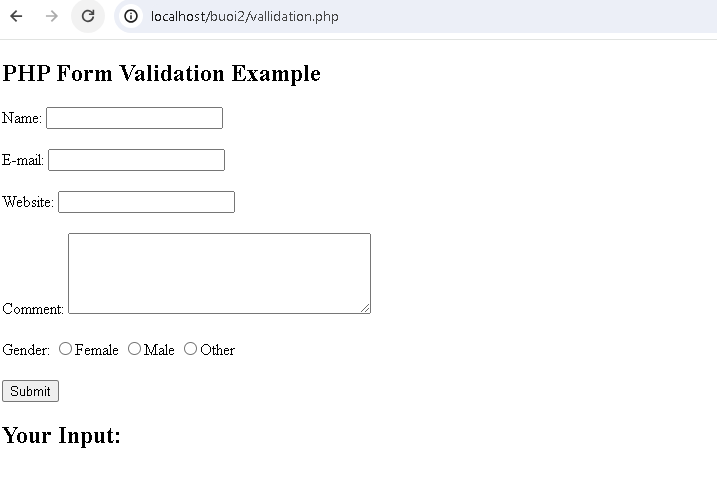
***-array***



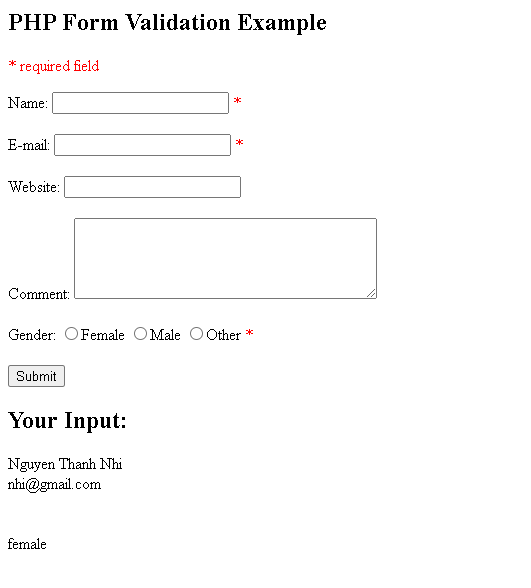
***-php1***



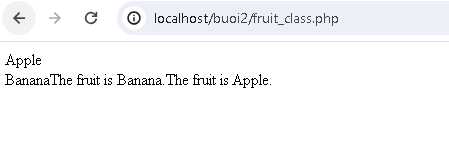
***-validation.php***



***-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 biên bằng cách sử dụng dấu $ và gán giá trị cho biến bằng dấu =

- Các chú ý khi đặt tên biến:

* Tên biến phải bắt đầu bằng 1 chữ cái hoặc dấu \_
* Tên biến phân biệt chữ hoa, thường
* Tránh sử dụng các ký tự đặc biệt khi đặt tên biến

- Ví dụ về khai báo biến

* Kiểu chuỗi: $name = “Nhi”;
* Kiểu số: $age = 21;
* Kiểu ngày tháng: $cur\_date = date(y-m-d);

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

-”{“: để bắt đầu 1 khối lệnh

-”}”: để kết thúc 1 khối lệnh

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

-// được dùng để ghi chú trên 1 dòng

-/\* và \*/: được dùng để ghi chú tất cả các lệnh nằm giữa hai dấu.

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

-for: sử dụng khi biết chi tiết về số lần lặp.

-while: sử dụng khi muốn lặp cho đến khi điều kiện không còn đúng thì vòng lặp sẽ ngừng

-do-while: tương tự như while, nhưng kiểm tra điều kiện ở cuối vòng lặp, đảm bảo ít nhất một lần thực hiện.

-foreach: sử dụng để lặp qua các phần tử trong mảng.

-> foreach khác với các khối lệnh vòng lặp khác là do nó được thiết kế đặc biệt để lặp qua mảng, giúp ngắn gọn hóa mã và tránh các lỗi liên quan đến chỉ số và độ dài mảng.

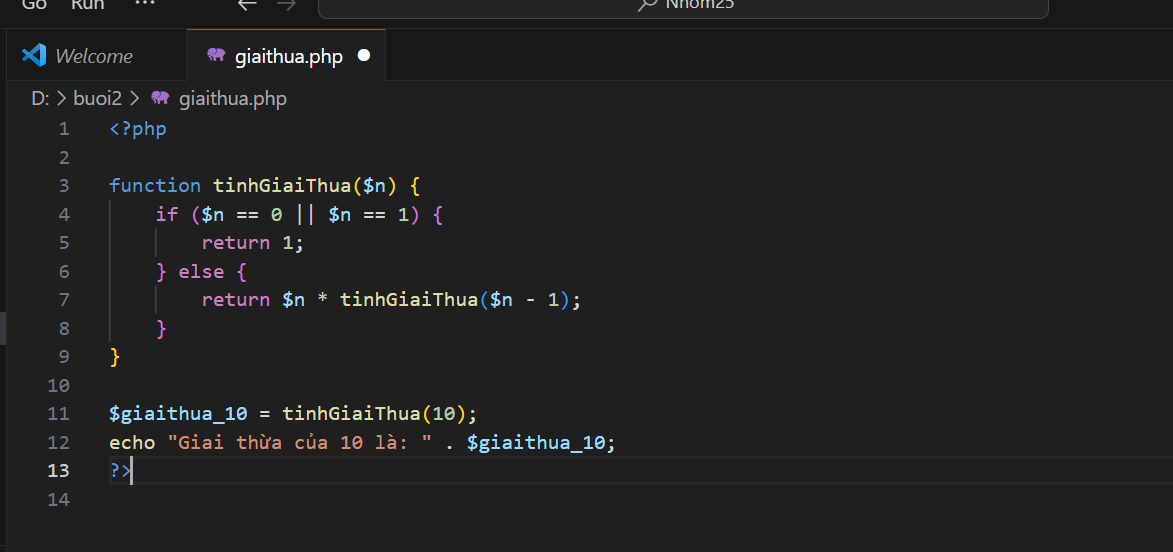
-Lệnh continue và break được dùng để kiểm soát luồng trong vòng lặp, giúp kiểm soát flow và tăng tính linh hoạt cho vòng lặp.

+continue: bỏ qua phần còn lại của vòng lặp và chuyển sang lần lặp tiếp theo.

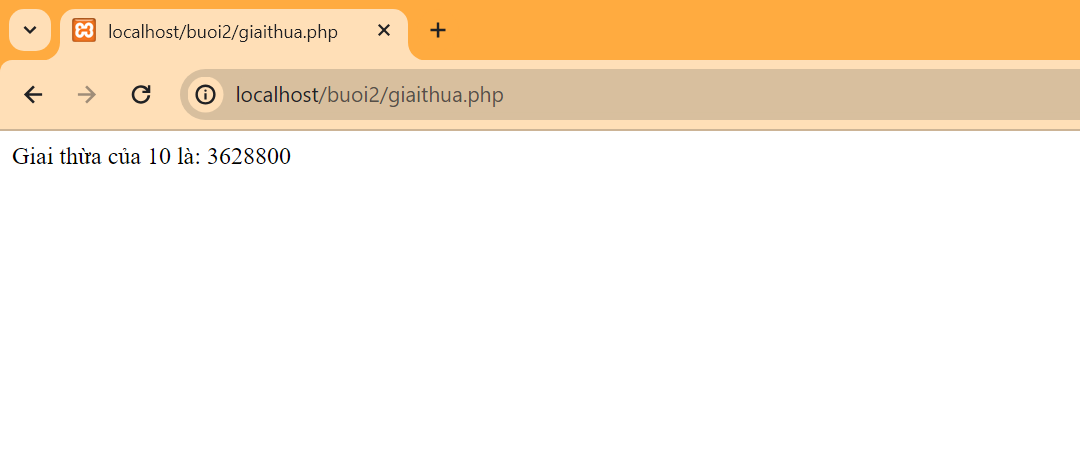
+break: kết thúc lệnh lặp ngay lặp tức và di chuyển đến các lệnh sau lệnh lặp.

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

-giaithua.php

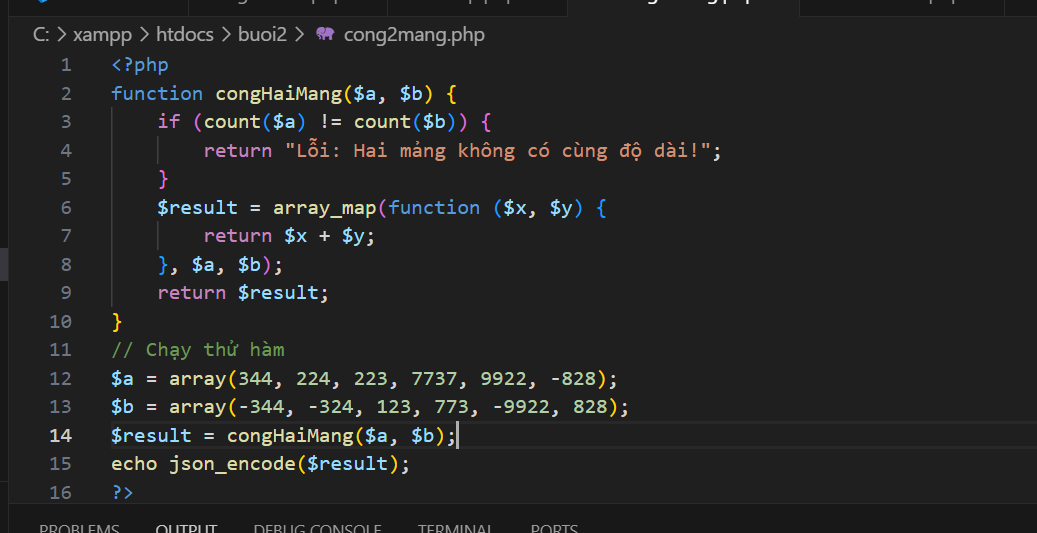


-Kết quả:

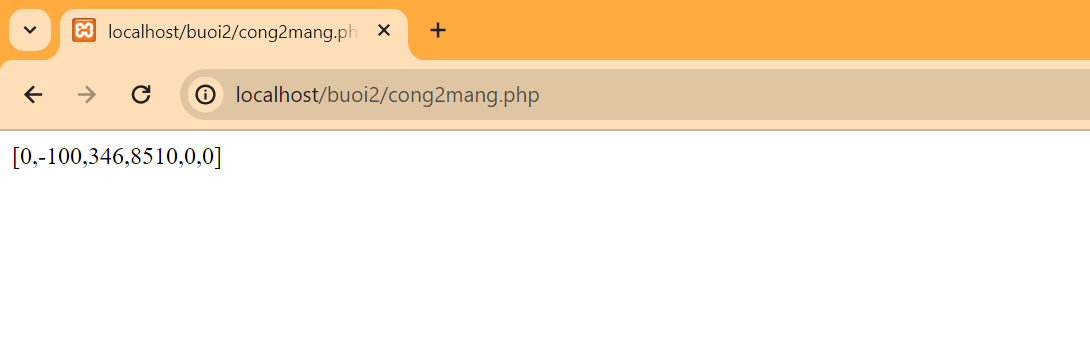


**7. Viết 1 tập tin cong2mang.php viết hàm xử lý để cộng được 2 mảng có cùng độ dài, nếu 2 mảng không cùng độ dài thì thông báo lỗi. Chạy thử hàm với 2 mảng sau: a={344,224,223,7737,9922,-828}, b={-344,-324,123,773,-9922,828} và cho biết kết quả hiển thị.**

-cong2mang.php:



-Kết quả:



**8. Chạy ví dụ tập tin validation.php, dựa vào link [[1]](#footnote-0), bạn hãy giải thích những bước diễn ra quá trình này.**

B1: Khai báo và thiết lập giá trị mặc định

->$name,$email,$gender,$comment,$website

B2: Kiểm tra form đã được submit chưa bằng cách sử dụng $\_SERVER[“REQUEST\_METHOD”] ==”POST”

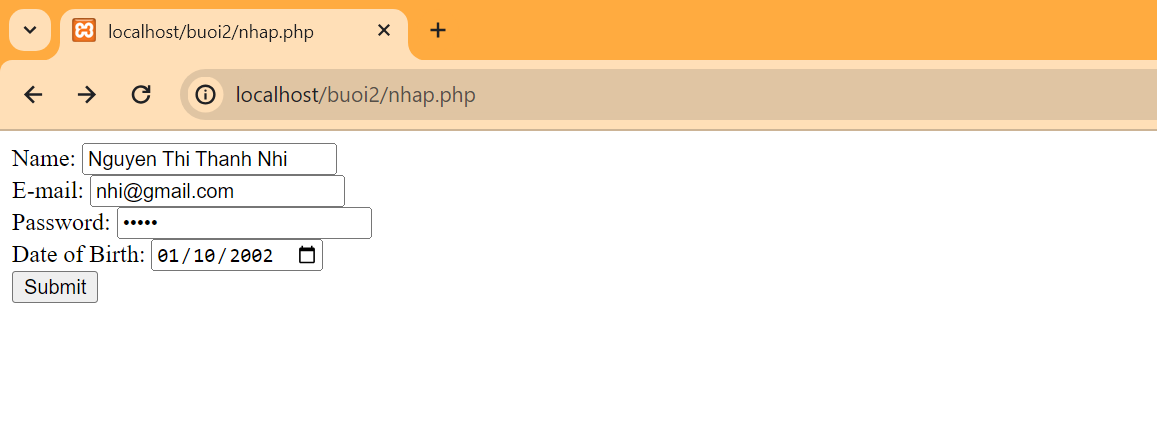
B3: Sử dụng hàm test\_input để lấy dữ liệu từ form và thực hiện xử lý dữ liệu như loại bỏ khoảng trắng không cần thiết và mã hóa HTML.

B4: Hiển thị form với các trường như tên, email, website, comment, gender và thu thập dữ liệu từ người dùng, dữ liệu sẽ được $\_SERVER[“PHP\_SEFT”] xử lí khi người dùng nhấn submit.

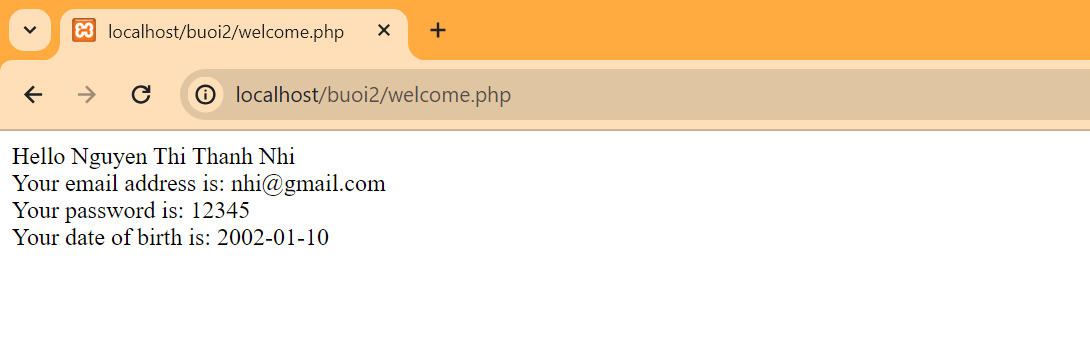
B5: Hiển thị kết quả là các dữ liệu đã nhập, dữ liệu được hiển thị thông qua lệnh echo.

**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

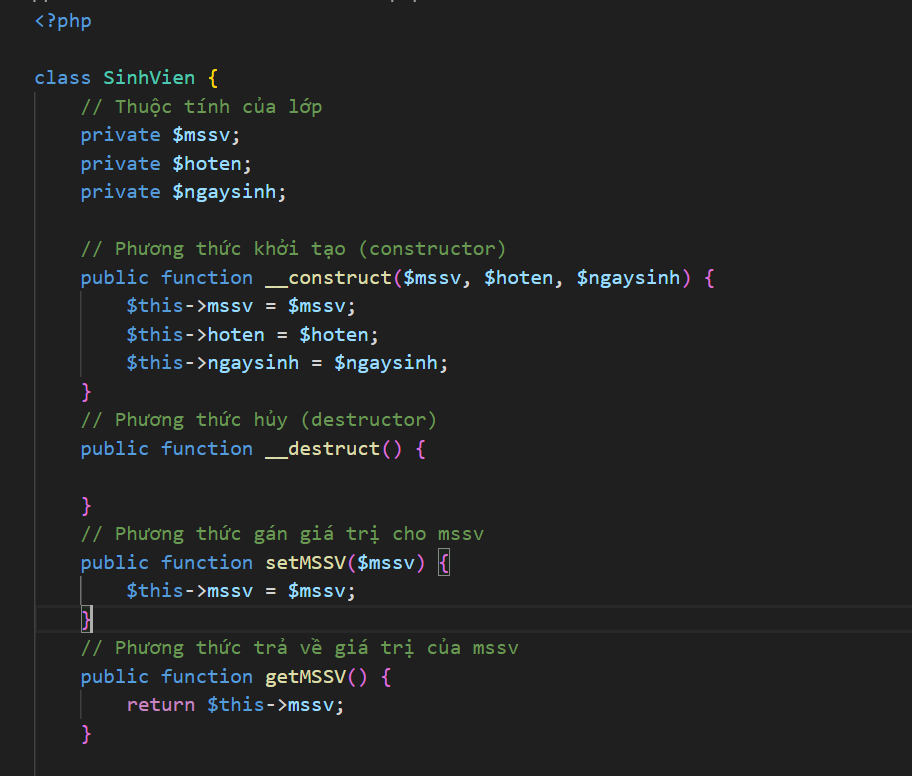


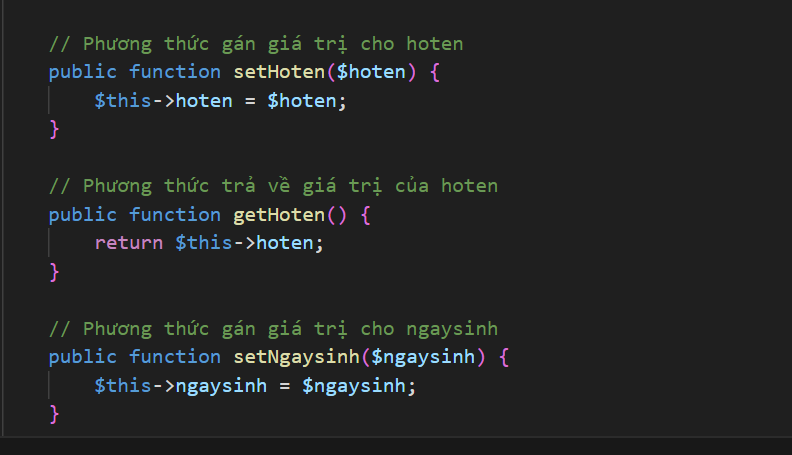
-welcome.php



**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.**

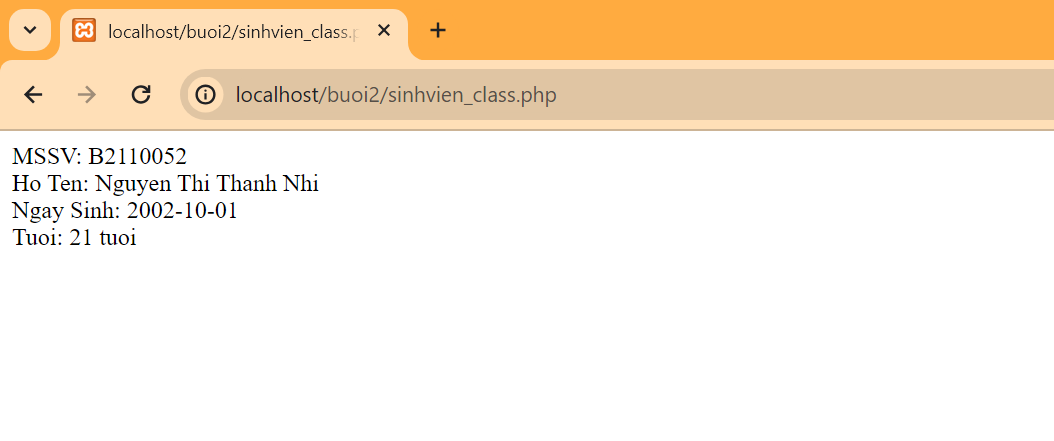
-sinhvien\_class.php







-Kết quả:



1. [↑](#footnote-ref-0)