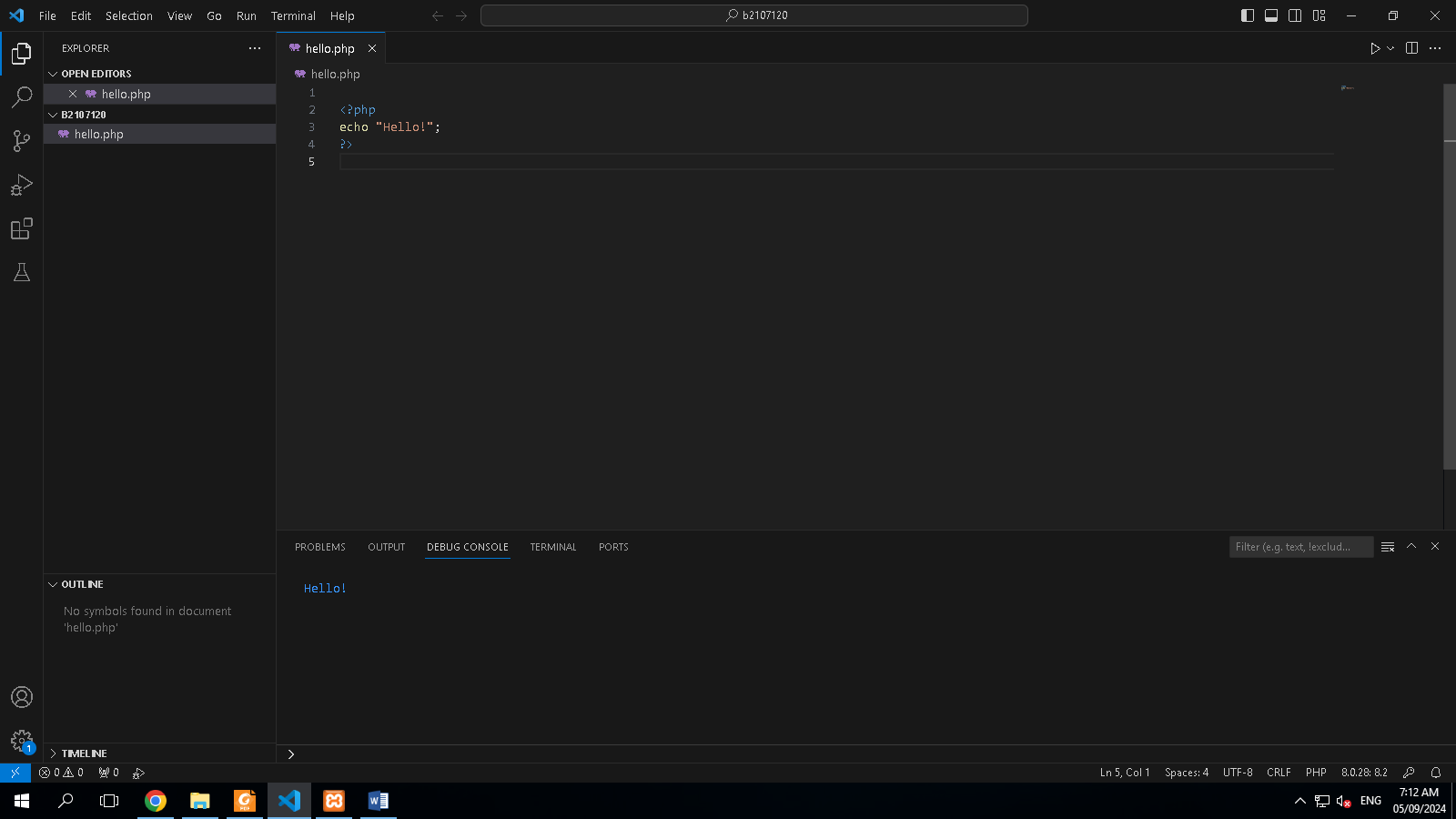
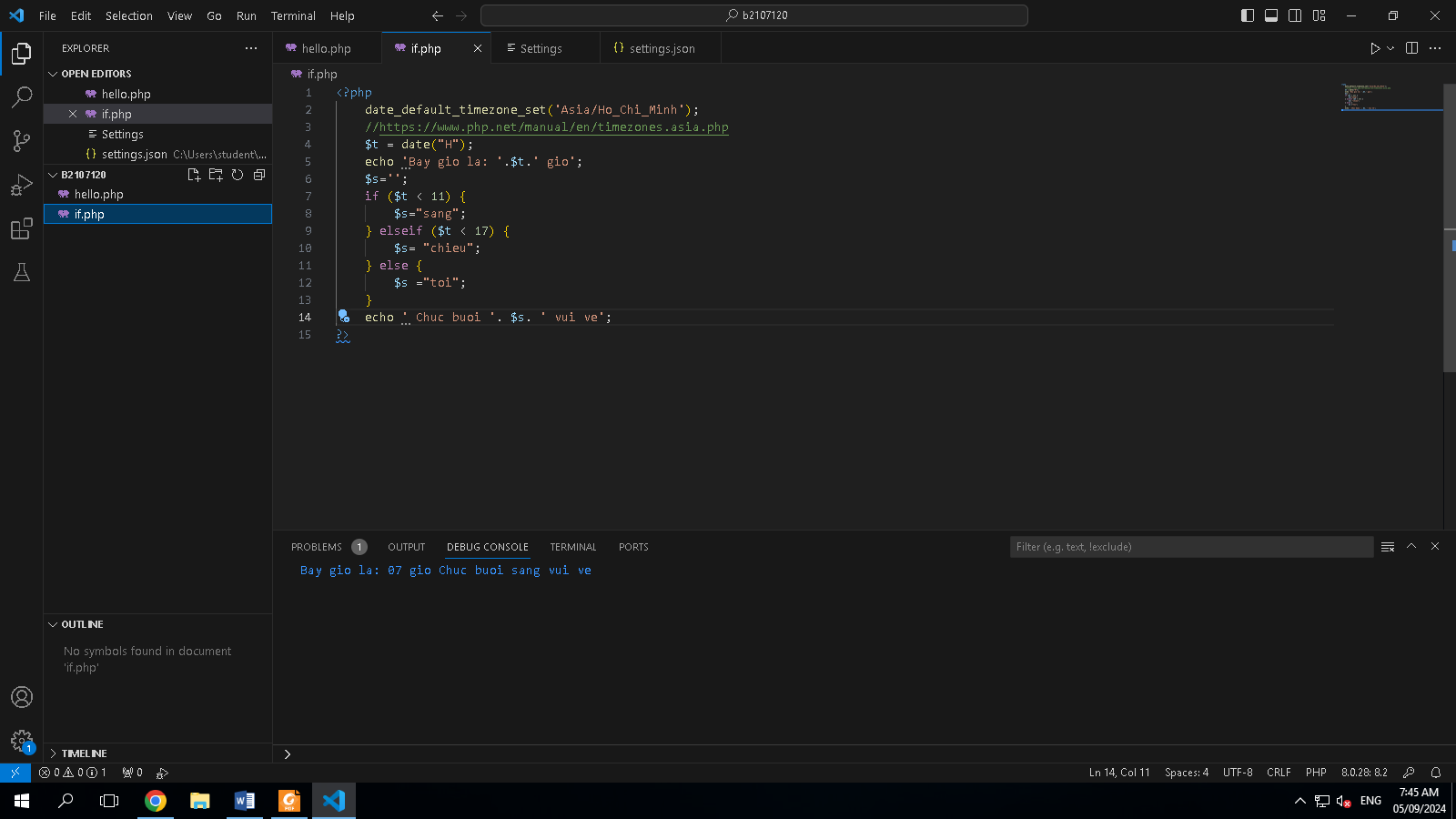
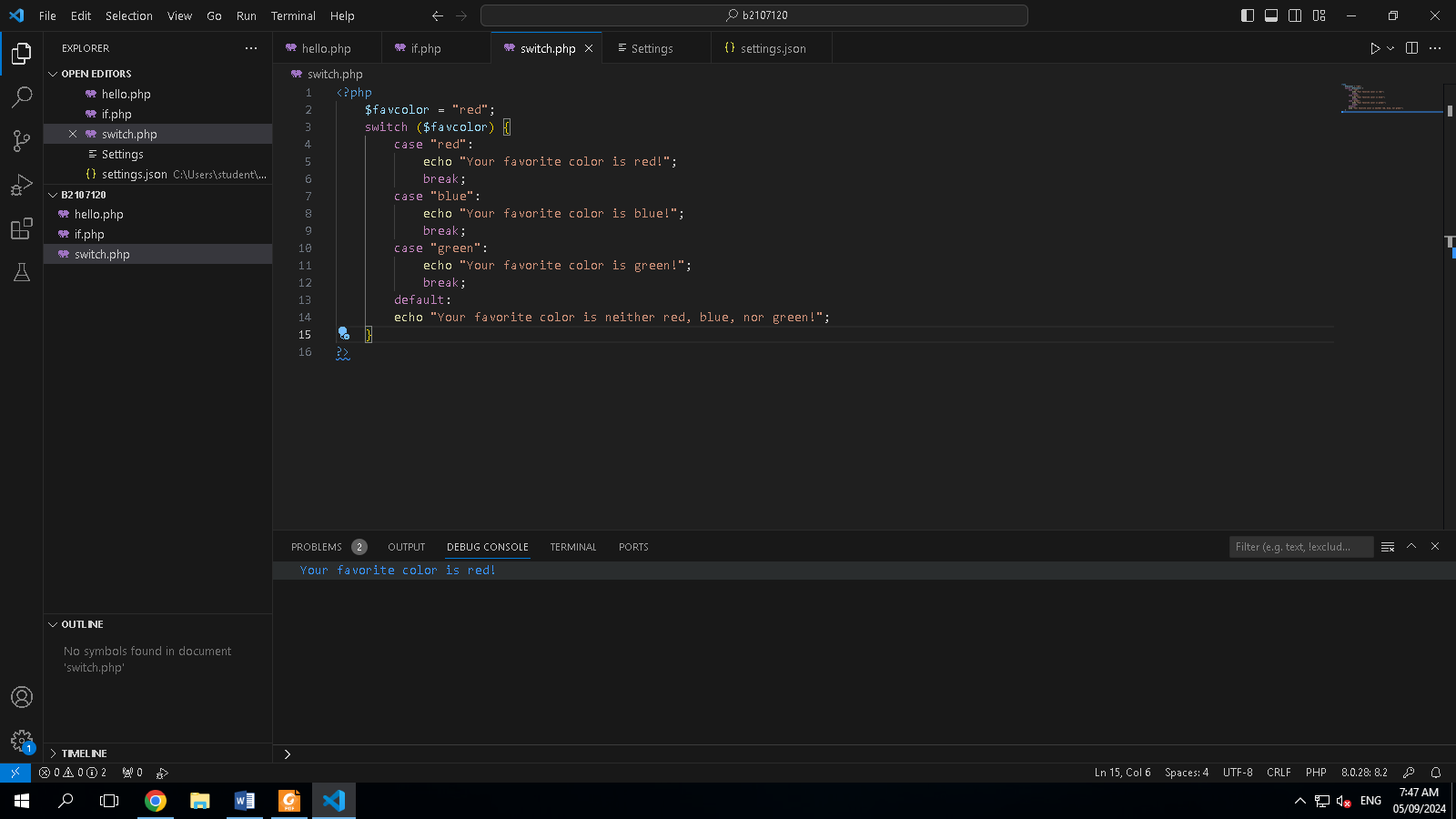
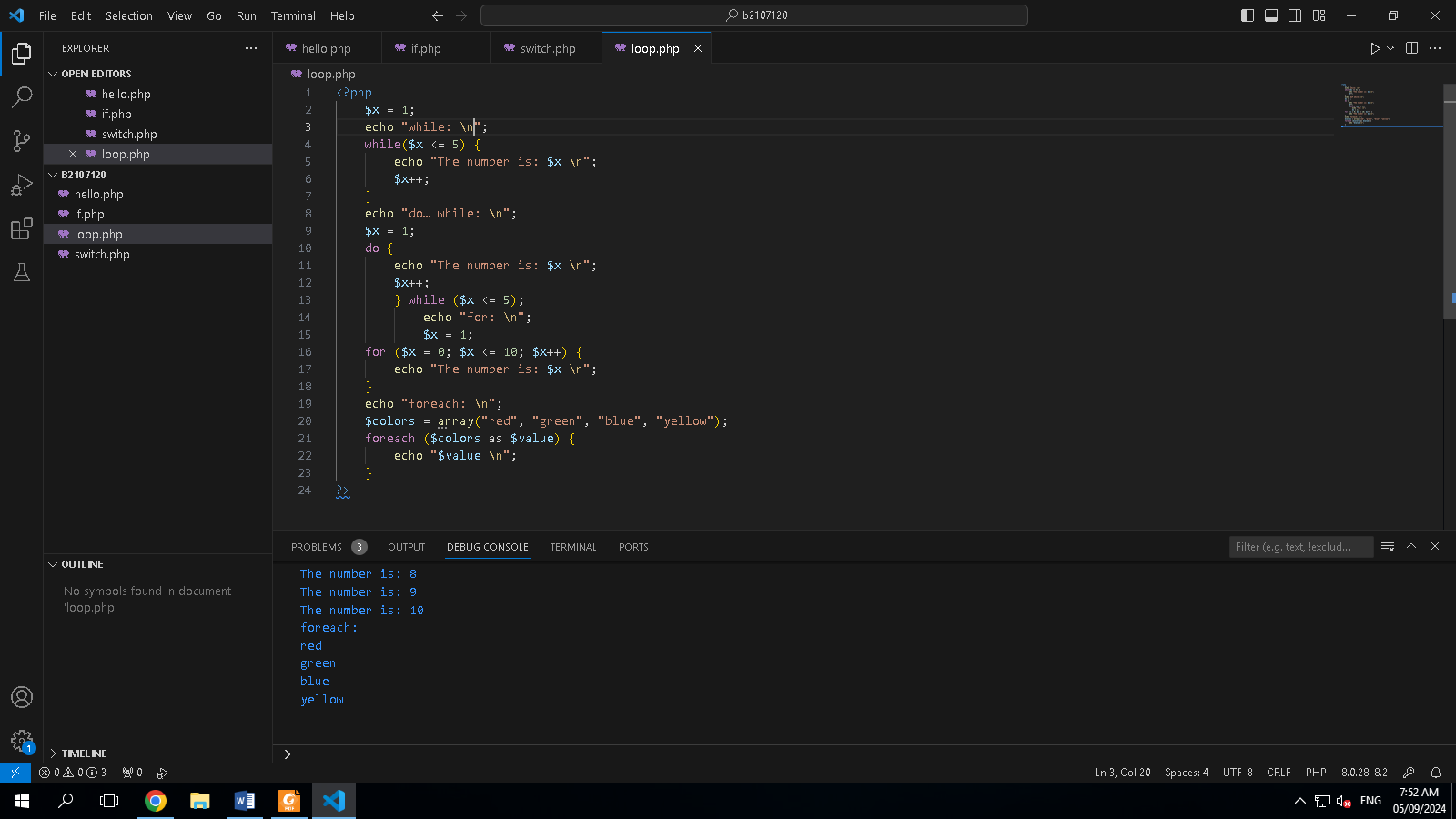
# **THỰC HÀNH BUỔI 2**

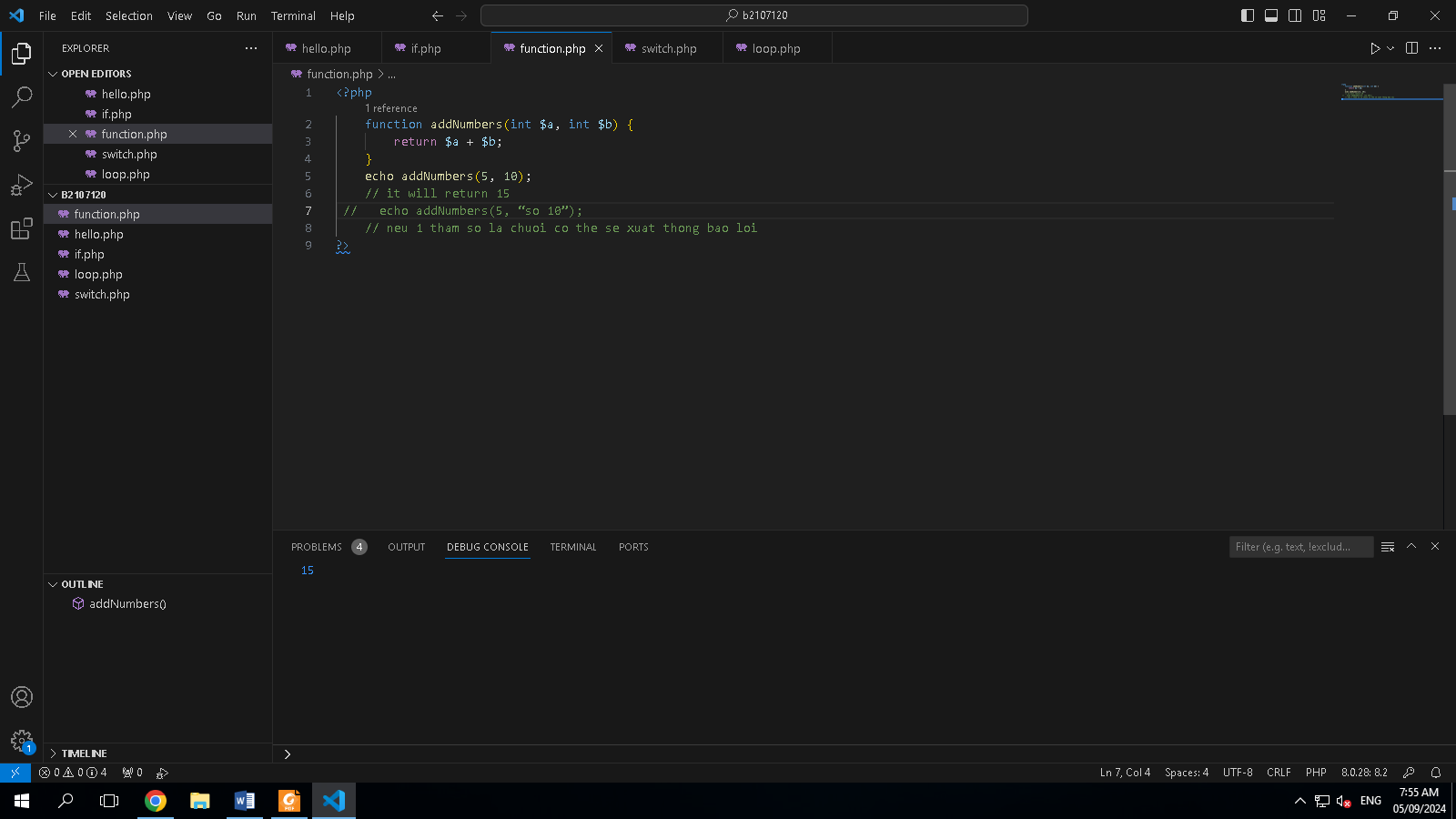
**1. Bạn hãy chạy tất cả các lệnh hướng dẫn ở trên và chụp lại màn hình kết quả.**

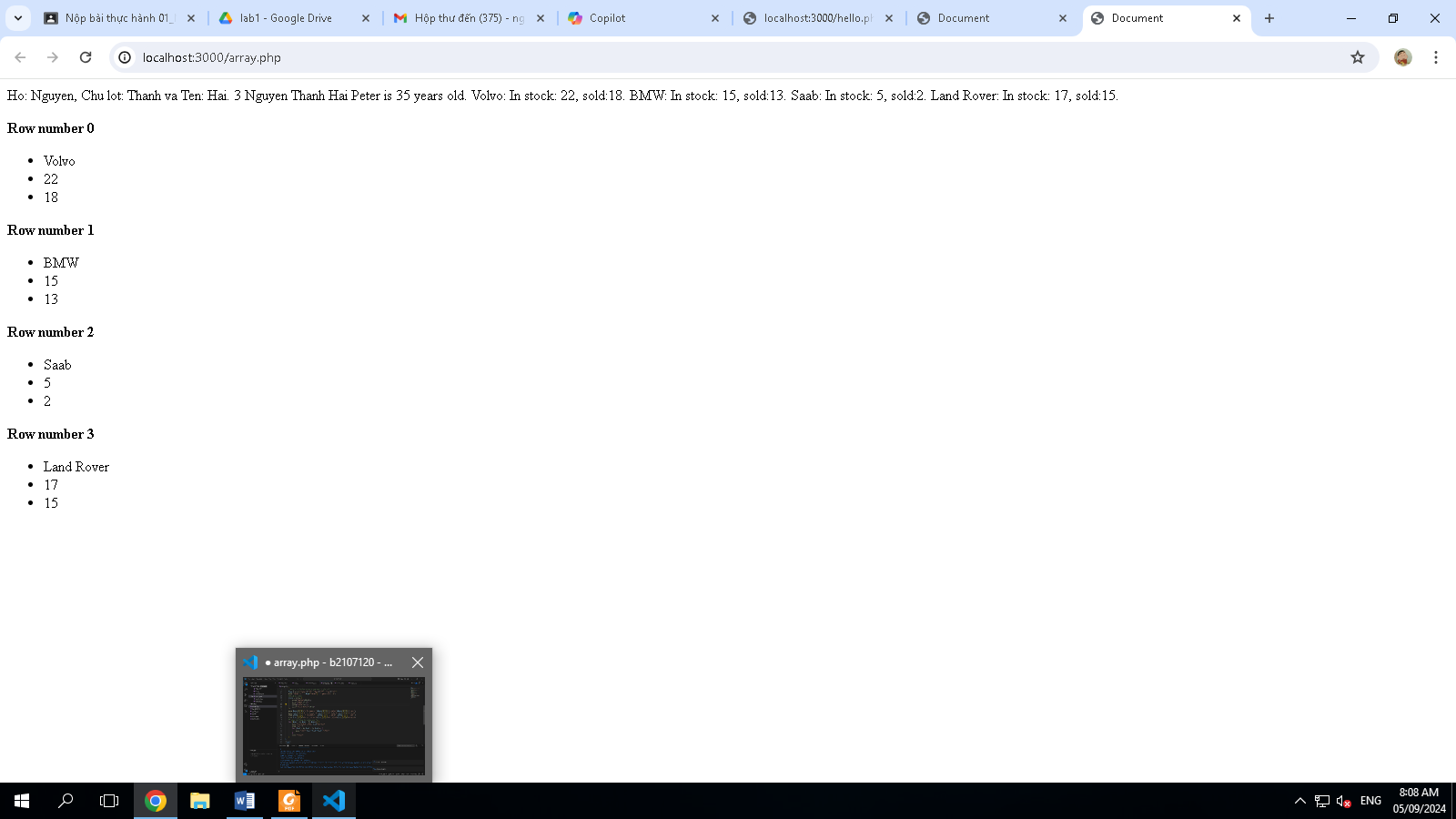
****

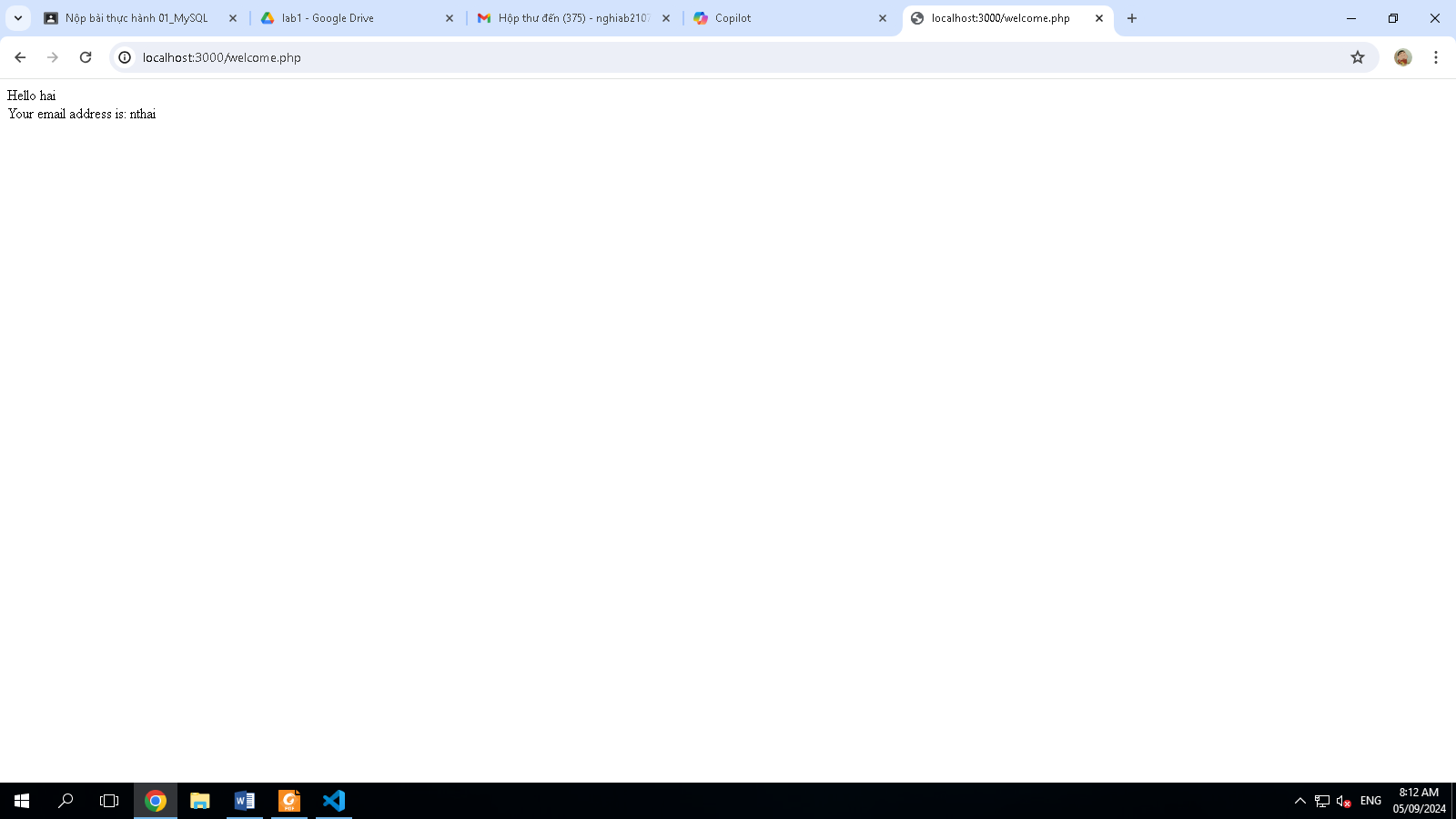
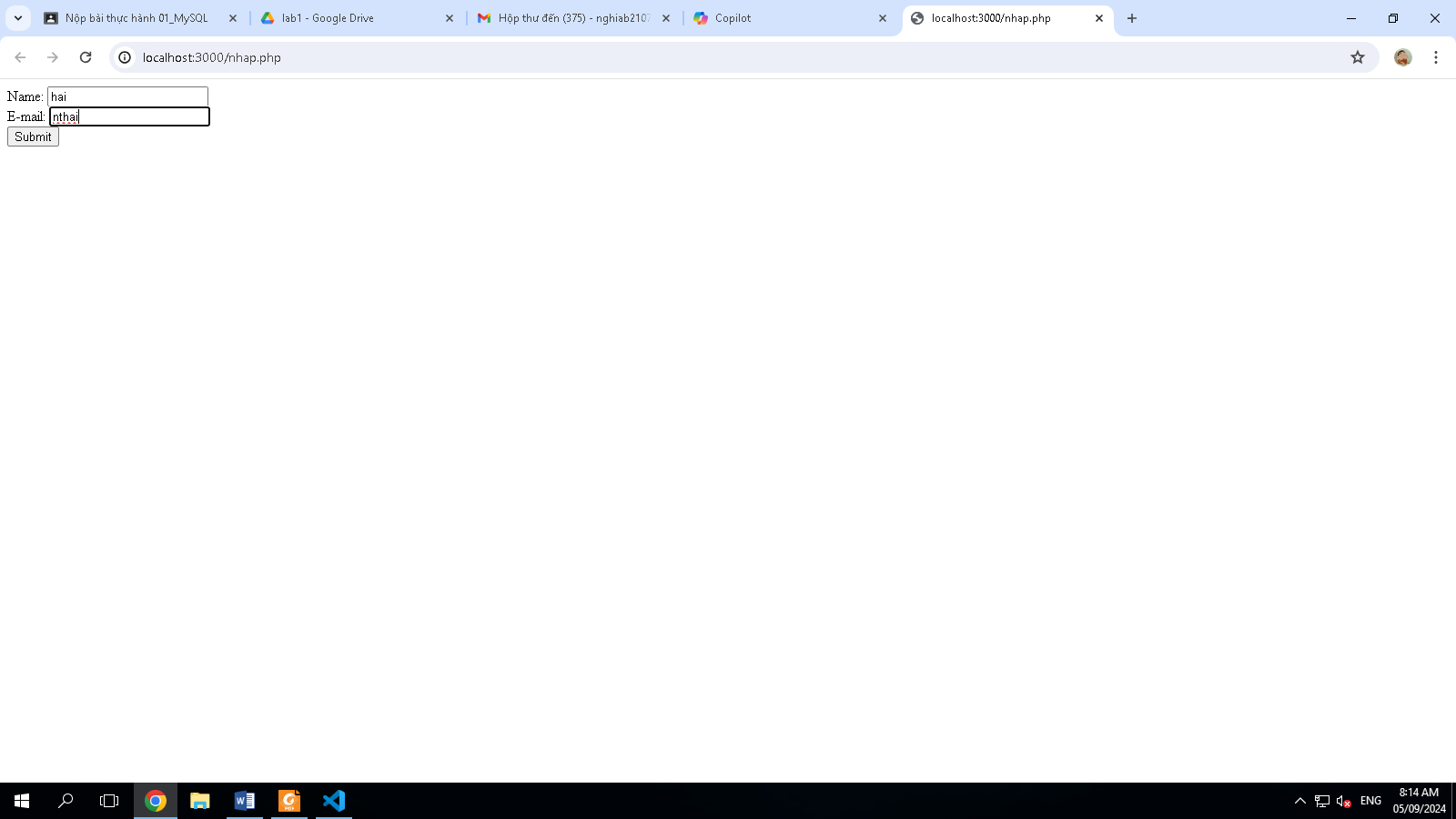


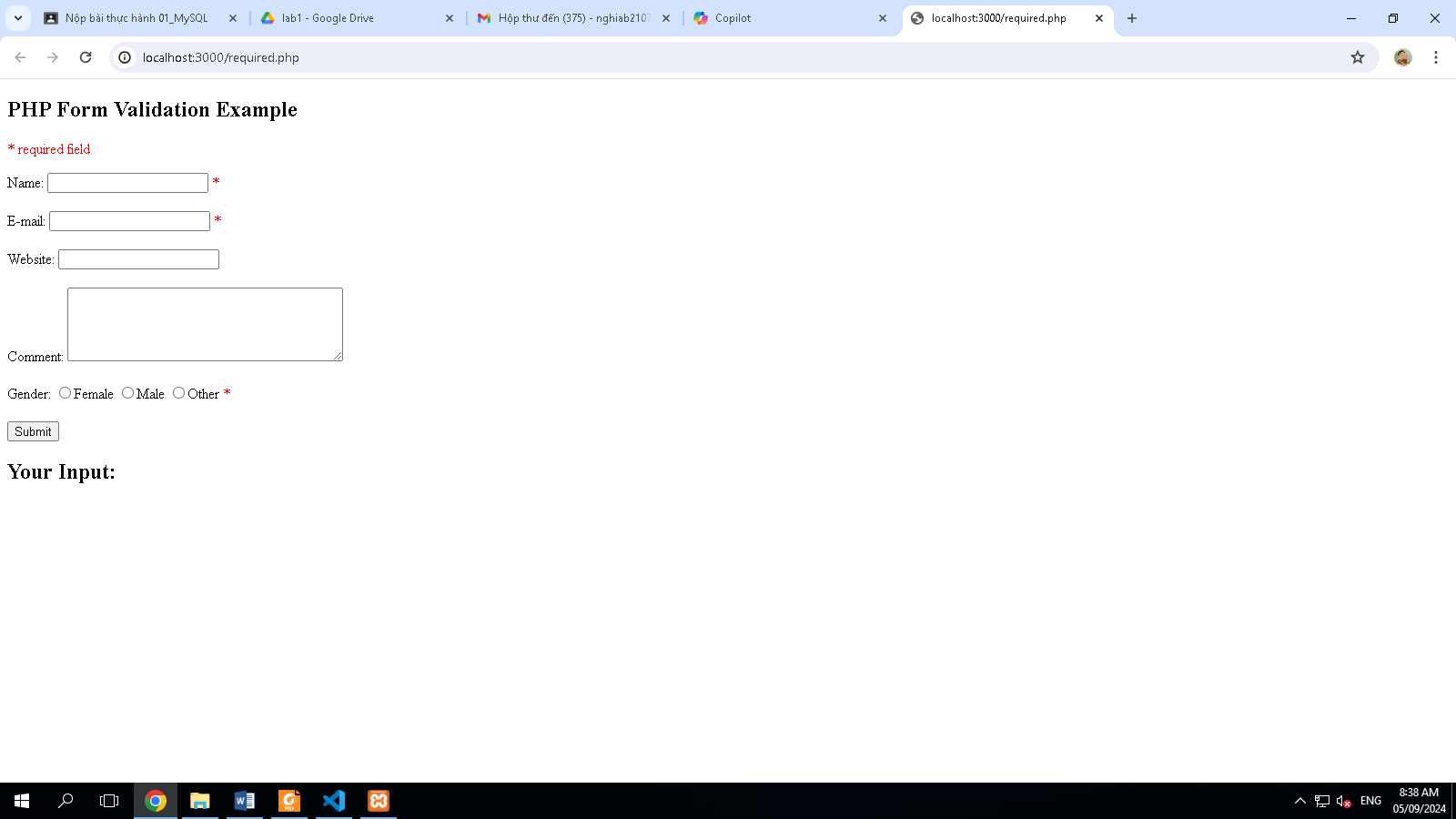


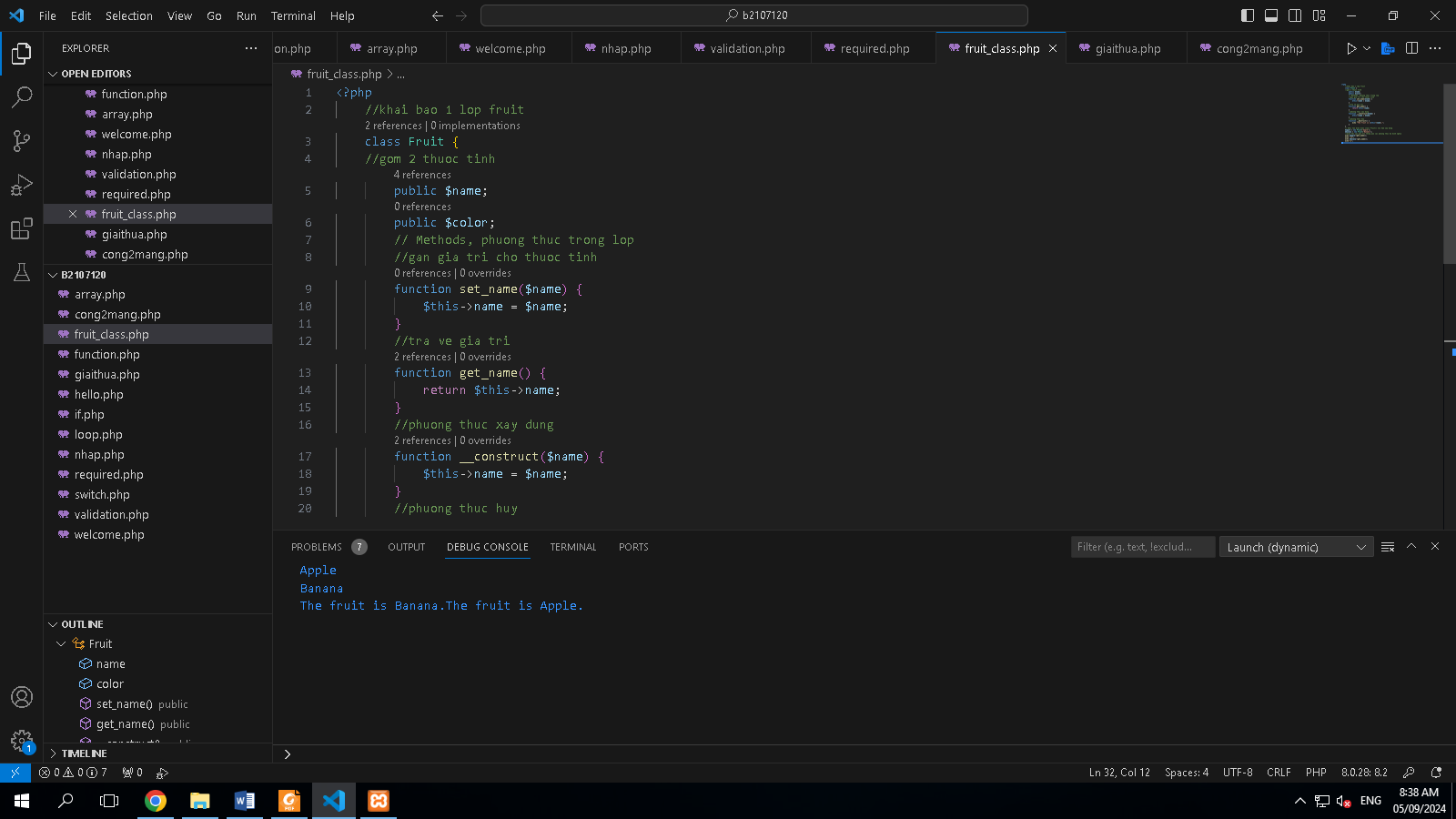












**2. Hãy cho biết cách khai báo biến, và gán trị cho biến trong PHP. Liệt kê các chú ý khi đặt  
tên biến. Cho các ví dụ về khai báo biến, và gán giá trị kiểu chuỗi, kiểu số, ngày tháng.**

Trả lời:  Biến trong PHP được khai báo bằng cách sử dụng ký hiệu $ trước tên biến. Sử dụng dấu = để gán giá trị cho biến.Các lưu ý khi đặt tên biến:

1. Bắt đầu bằng ký tự hoặc dấu gạch dưới: Tên biến phải bắt đầu bằng một chữ cái hoặc dấu gạch dưới (\_), không được bắt đầu bằng số.
2. Chỉ chứa các ký tự chữ cái, số và dấu gạch dưới: Tên biến chỉ được chứa các ký tự chữ cái (a-z, A-Z), số (0-9) và dấu gạch dưới (\_).
3. Phân biệt chữ hoa và chữ thường: PHP phân biệt chữ hoa và chữ thường, vì vậy $name và $Name là hai biến khác nhau.
4. Không chứa ký tự đặc biệt: Tên biến không được chứa các ký tự đặc biệt như @, #, !, %, ^, &, \*, (, ), -, +, =, {, }, [, ], |, \, :, ;, ", ', <, >, ,, ., ?, /.

**Ví dụ:**

**Kiểu chuỗi:**

<?php

$name= "Nguyễn Văn A";

?>

**Kiểu số:**

<?php

$age = 25;

$height = 1.75; // Số thực

?>

**Kiểu ngày tháng:**

<?php

$date = "2024-09-05";

?>

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

**Trả lời:**

Bắt đầu khối lệnh PHP: <?php

Kết thúc khối lệnh PHP: ?>

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

**Trả lời:**

Ghi chú 1 dòng ta sử dụng // hoặc #

Ghi chú nhieufe dòng ta sử dụng /\* \*/

**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.**

**Trả lời:**

Trong PHP, có bốn loại khối lệnh để thực hiện vòng lặp:

* For: Lặp qua một khối lệnh một số lần xác định.
* While: Lặp qua một khối lệnh khi điều kiện được chỉ định là đúng.
* Do…while: Lặp qua một khối lệnh ít nhất một lần, sau đó tiếp tục lặp khi điều kiện được chỉ định là đúng.
* Foreach: Lặp qua mỗi phần tử trong một mảng hoặc mỗi thuộc tính của một đối tượng.

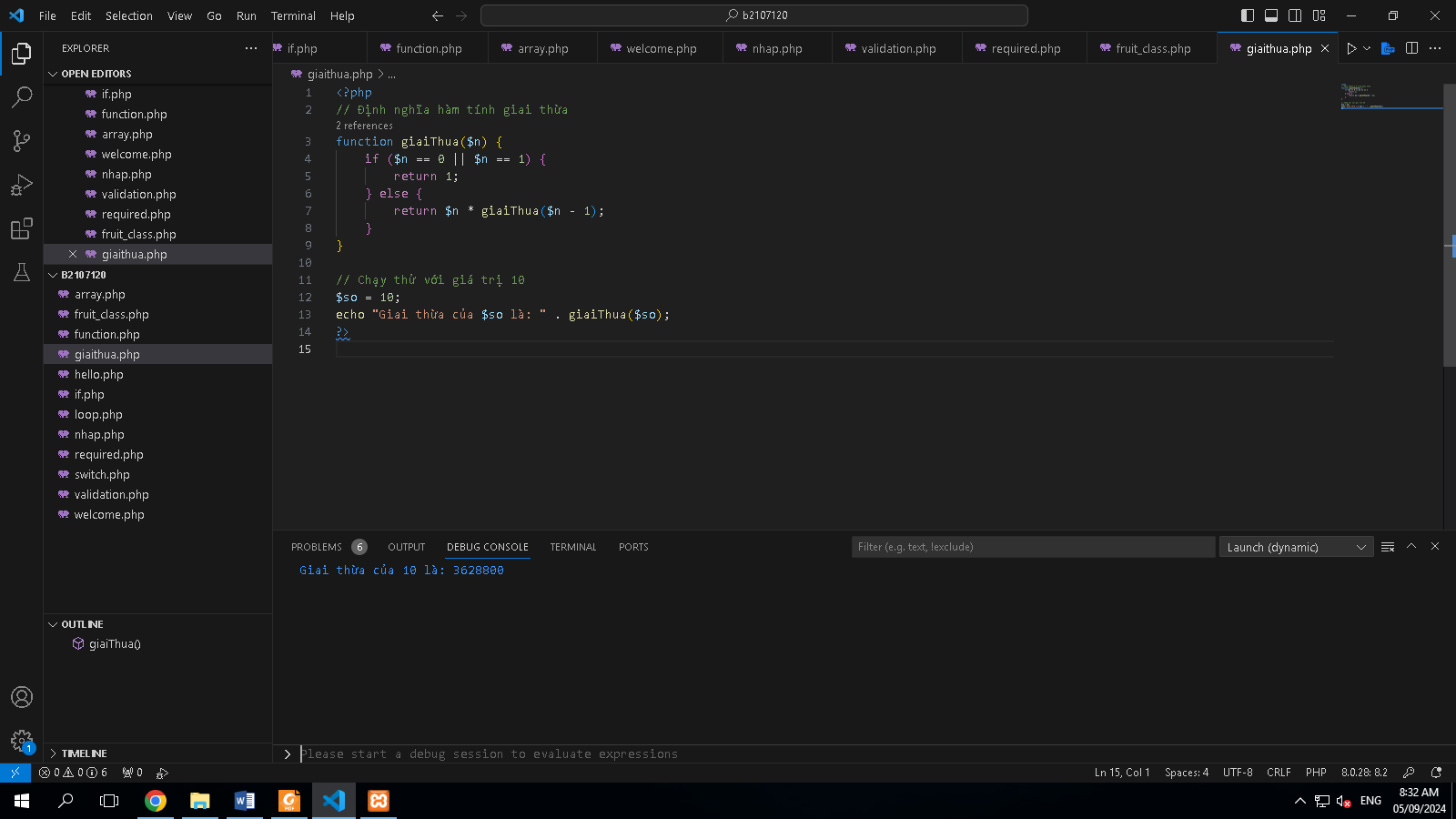
**Khác biệt giữa foreach và các lệnh khác:** foreach được thiết kế để lặp qua các phần tử của mảng hoặc đối tượng, giúp mã nguồn trở nên ngắn gọn và dễ đọc hơn khi làm việc với mảng hoặc đối tượng.

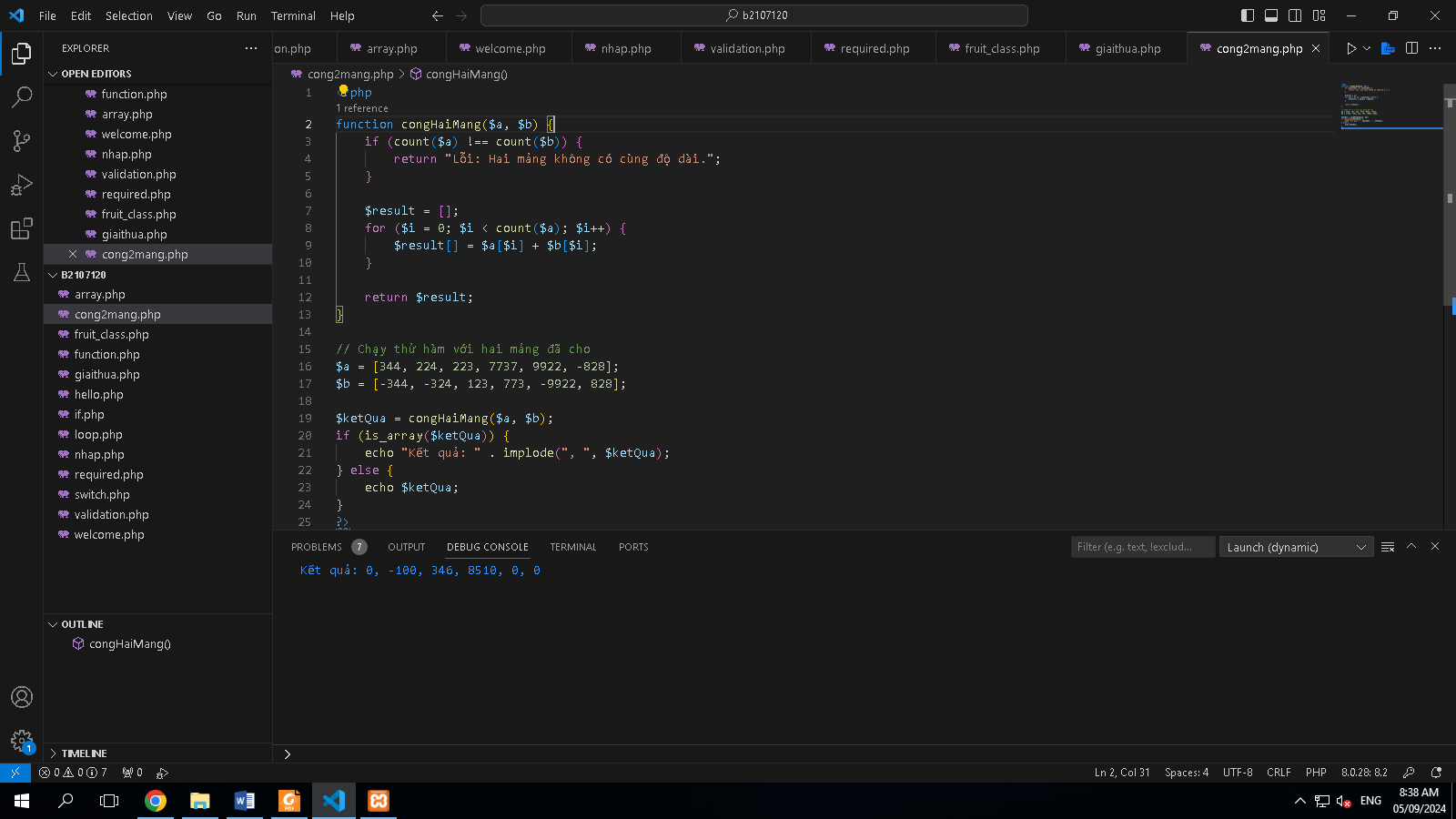
for, while, và do…while yêu cầu bạn phải quản lý biến đếm và điều kiện lặp, trong khi foreach tự động xử lý việc này.

**Cách dùng của lệnh continue và break trong các vòng lặp:**

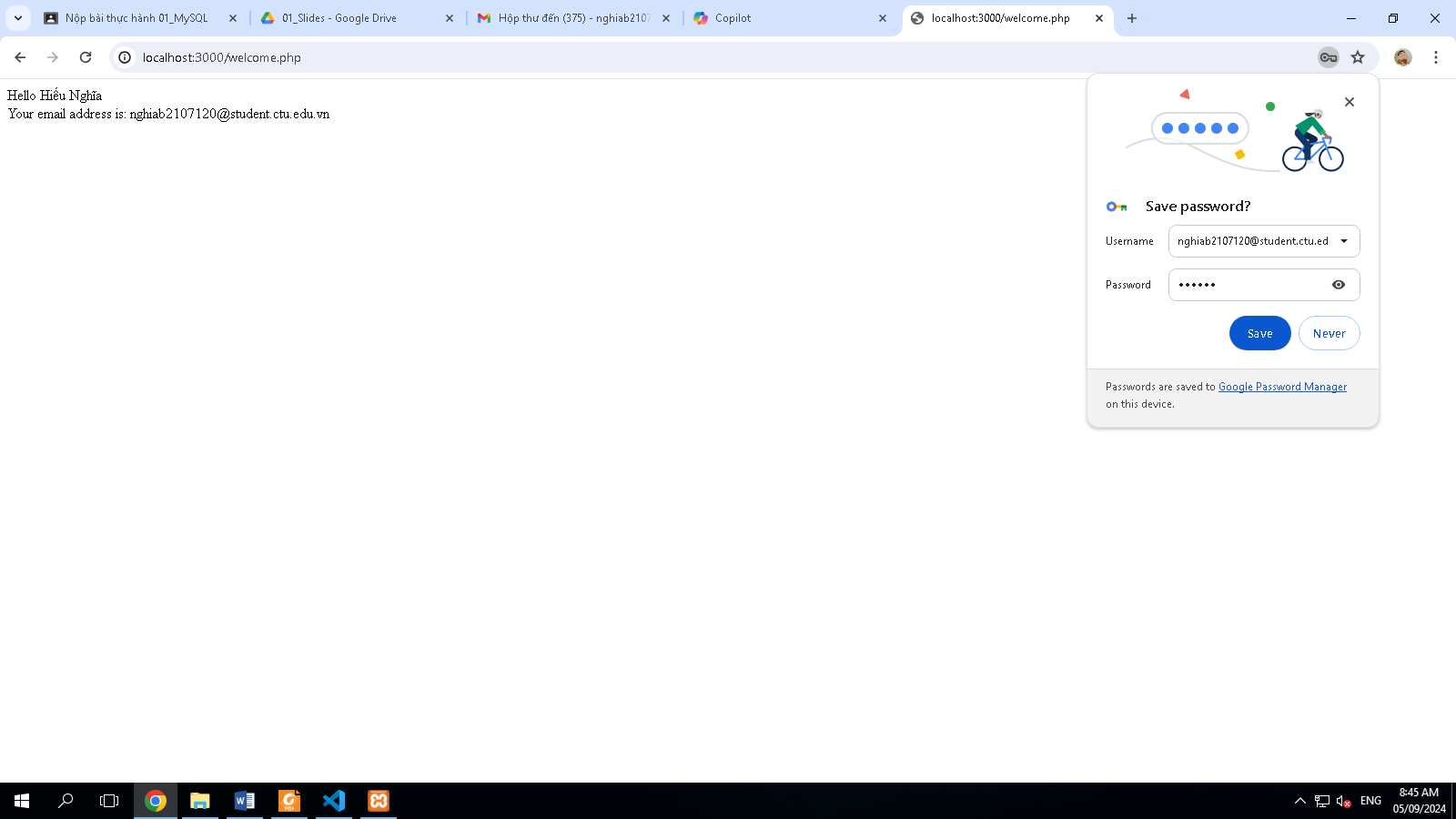
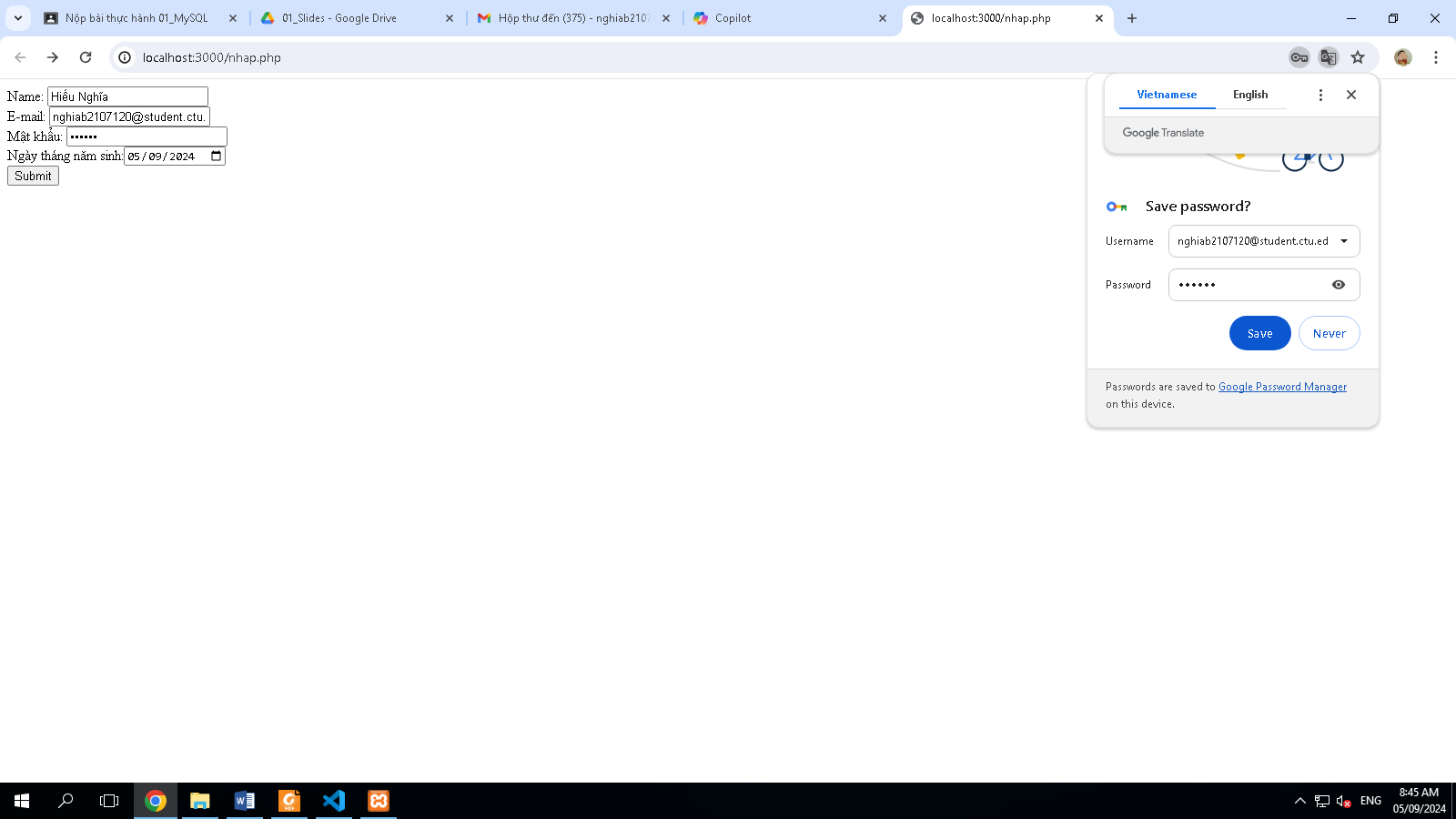
* Continue: Dùng để bỏ qua phần còn lại của vòng lặp hiện tại và tiếp tục với vòng lặp tiếp theo
* Break: Dùng để thoát khỏi vòng lặp ngay lập tức

**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.**

**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.**



**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.**

