

1 dạng mảng ký tự:

```
$countries = ["Japan", "Brazil", "Canada", "Australia", "Denmark"];
```

**Bài 1:** Viết chương trình cho phép người dùng nhập một mảng các ký tự thông qua **textbox**. Sau khi nhập, hãy **sắp xếp mảng theo thứ tự tăng dần**, sau đó hiển thị:

- Mảng đã được sắp xếp.
- Tổng số phần tử trong mảng.

Kết quả cần được hiển thị **ngay bên dưới form nhập liệu**.

**Bài 2:** Viết chương trình nhập một mảng các ký tự từ **textbox**. Sau đó, thực hiện các yêu cầu sau:

- Hiển thị **mảng ban đầu**.
- Sắp xếp mảng theo **thứ tự giảm dần**.
- Hiển thị mảng đã sắp xếp.
- Hiển thị tổng số phần tử của mảng.

Lưu ý: Toàn bộ kết quả cần được **hiển thị ở trang mới** sau khi người dùng gửi dữ liệu.

**Bài 3:** Nhập một mảng các ký tự thông qua **textbox**.

Chương trình cần xác định **phần tử có độ dài lớn nhất** trong mảng, sau đó hiển thị:

- Từ dài nhất.
- Số lượng ký tự của từ đó.

Kết quả được hiển thị **ngay bên dưới form nhập**.

**Bài 4:** Người dùng nhập một mảng các **số nguyên** vào **textbox**.

Chương trình cần tính **tổng của các số lẻ** trong mảng và hiển thị kết quả **bên dưới form**.

**Bài 5:** Tạo chương trình cho phép nhập một mảng các số nguyên từ **textbox**. Sau đó, tính **tổng các phần tử có chỉ số chẵn** (vị trí 0, 2, 4, ...) trong mảng.

Hiển thị:

- Mảng ban đầu.
- Tổng các phần tử có chỉ số chẵn.

Kết quả được **hiển thị ở trang mới** sau khi người dùng gửi dữ liệu.

## **Bài 6: Quản lý danh sách sinh viên bằng PHP và MySQL**

*Yêu cầu:*

### **1. Tạo cơ sở dữ liệu và bảng:**

- Tạo cơ sở dữ liệu có tên là `SinhVienDB` (`utf8mb4_general_ci`).
- Trong cơ sở dữ liệu này, tạo bảng `SinhVien` với các cột:
  - `MaSV` (mã sinh viên, `int`, `PK`)
  - `HoTen` (họ tên, `varchar(25)`)
  - `DienThoai` (số điện thoại, `varchar(10)`)

- DiaChi (địa chỉ, varchar(20))
- Diem (điểm, tinyint)

Hãy nhập trước 3 bản ghi sinh viên để kiểm tra hiển thị dữ liệu ban đầu.

## 2. Trang danh sách sinh viên:

Viết mã PHP để hiển thị **danh sách tất cả sinh viên** có trong bảng SinhVien. Trong bảng hiển thị, thêm một cột **Học bổng** được tính theo quy tắc sau:

- Nếu điểm của sinh viên  $\geq 8$ , học bổng = 1000.
- Ngược lại, học bổng = 0.
- Trên trang này cũng cần có **liên kết (link)** để người dùng chuyển sang trang **nhập dữ liệu mới**.

## 3. Trang nhập dữ liệu sinh viên:

- Tạo một **form nhập thông tin sinh viên** (gồm các trường: MaSV, Họ tên, Số điện thoại, Địa chỉ).
- Sau khi người dùng nhập và gửi dữ liệu, chương trình sẽ **lưu thông tin vào cơ sở dữ liệu**, sau đó **chuyển về trang danh sách**, nơi người dùng có thể thấy sinh viên vừa được thêm vào.

## 4. Chức năng tìm kiếm sinh viên theo mã sinh viên:

- Tạo một form cho phép người dùng **nhập vào mã sinh viên (MaSV)**.
- Khi người dùng bấm nút “Tìm”, chương trình sẽ:
  - Hiển thị **thông tin chi tiết** của sinh viên nếu tìm thấy.
  - Nếu không có kết quả, hiển thị thông báo: **"Không có MaSV cần tìm."**

## 5. Chức năng xóa sinh viên:

- Tạo form cho phép người dùng nhập vào MaSV để xóa.
- Khi nhấn nút “Xóa”, chương trình sẽ hiển thị hộp thoại:

**“Bạn có chắc chắn muốn xóa sinh viên này không?”**

- Nếu người dùng chọn **“Có”**, sinh viên sẽ được xóa khỏi cơ sở dữ liệu.
- Nếu chọn **“Không”**, thao tác sẽ bị hủy.
- Sau khi xóa thành công, trang sẽ quay lại danh sách để thể hiện rõ rằng **sinh viên đã bị xóa**.

Gợi ý:

Bạn có thể tích hợp các form (**thêm, tìm kiếm, xóa**) ngay trên **trang danh sách sinh viên**, để người dùng có thể dễ dàng thao tác và quan sát thay đổi dữ liệu ngay lập tức.