

1. Khái niệm & Khởi tạo (Initialization)

- **Định nghĩa:** Tập hợp các phần tử có thứ tự, lưu trữ đa dạng kiểu dữ liệu.
- **Khai báo:** let mảng = [];
(Mảng rỗng là điểm bắt đầu phổ biến).
- **Cấu trúc:** Mỗi phần tử đi kèm một **Chỉ số (Index)** bắt đầu từ 0.

2. Các thuộc tính & Phương thức cơ bản

- **.length:** Trả về tổng số phần tử (Dùng để in tổng số sách, số bạn đọc).
- **.push(giá_trị):** Thêm phần tử vào cuối mảng (Dùng khi nhập sách mới).
- **.splice(vị_trí, số_lượng):** Xóa phần tử tại vị trí xác định (Dùng khi loại bỏ sách hỏng).
- **Truy cập:** mảng[i] để lấy hoặc sửa giá trị tại vị trí i.

3. Mảng song song (Parallel Arrays)

- **Đặc điểm:** Nhiều mảng có cùng độ dài, các phần tử ở cùng chỉ số i có liên quan mật thiết (Ví dụ: booksId[i] đi kèm với booksName[i]).
- **Ứng dụng:** Quản lý thông tin đa chiều (Mã - Tên - Tình trạng - Số lượng).
- **Lưu ý:** Khi xóa hoặc sửa ở một mảng, phải thực hiện tương ứng trên các mảng còn lại để tránh lệch dữ liệu.

4. Thuật toán Duyệt & Xử lý (Logic lồng ghép)

- **Vòng lặp for:** Xương sống để duyệt qua toàn bộ danh sách.
- **Tìm kiếm (Search):**
 - Sử dụng biến cờ (isDuplicate) để kiểm tra trùng lặp mã ID.
 - Dùng .includes() và .toLowerCase() để tìm kiếm chuỗi không phân biệt hoa thường.
- **Thống kê (Statistics):** * Sử dụng biến đếm (count++) kết hợp lệnh if.
 - Kỹ thuật tìm Min/Max (Giả định phần tử đầu là nhỏ nhất/lớn nhất rồi so sánh).
- **Phân loại (Classification):** Dùng if...else if để đưa ra các cảnh báo hệ thống (Tốt/Nhắc nhở/Nghiêm trọng).

5. Kỹ năng bổ trợ (Input/Output)

- **prompt():** Nhận dữ liệu từ người dùng.
- **parseInt():** Ép kiểu dữ liệu về số nguyên (Quan trọng khi tính toán số ngày quá hạn/tồn kho).

- **console.log()**: Xuất báo cáo chi tiết.
- **console.warn/error**: Phân cấp mức độ nghiêm trọng của thông tin.