**Minh họa về bài toán "Mua hàng"**

Bạn được yêu cầu xây dựng một hệ thống đơn giản để quản lý giỏ hàng trực tuyến. Hệ thống này cần cho phép người dùng thêm sản phẩm vào giỏ, tính tổng số tiền phải trả, và hiển thị thông tin chi tiết của từng sản phẩm trong giỏ hàng. Cụ thể, bạn cần xây dựng 3 lớp: **Product**, **CartItem**, và **Cart** để quản lý các chức năng của hệ thống giỏ hàng.

### Yêu cầu cụ thể:

**Class Product**:

* 1. Đại diện cho sản phẩm trong cửa hàng.
  2. Thuộc tính:
     1. id (int): Mã sản phẩm (duy nhất cho mỗi sản phẩm).
     2. name (string): Tên của sản phẩm.
     3. price (float): Giá của sản phẩm.
  3. Phương thức:
     1. \_\_construct($id, $name, $price): Phương thức khởi tạo nhận mã, tên và giá sản phẩm.
     2. getId(): Trả về mã sản phẩm.
     3. getName(): Trả về tên sản phẩm.
     4. getPrice(): Trả về giá sản phẩm.

**Class CartItem**:

* 1. Đại diện cho một mặt hàng trong giỏ hàng.
  2. Thuộc tính:
     1. product (Product): Đối tượng sản phẩm.
     2. quantity (int): Số lượng sản phẩm.
  3. Phương thức:
     1. \_\_construct($product, $quantity = 1): Phương thức khởi tạo nhận đối tượng sản phẩm và số lượng (mặc định là 1).
     2. getProduct(): Trả về đối tượng sản phẩm.
     3. getQuantity(): Trả về số lượng sản phẩm trong giỏ.
     4. setQuantity($quantity): Thiết lập số lượng mới cho sản phẩm.
     5. getTotalPrice(): Tính tổng giá của sản phẩm (số lượng \* giá của sản phẩm).

**Class Cart**:

* 1. Quản lý toàn bộ giỏ hàng.
  2. Thuộc tính:
     1. items (array): Mảng chứa các đối tượng CartItem.
  3. Phương thức:
     1. addProduct($product, $quantity = 1): Thêm một sản phẩm vào giỏ với số lượng nhất định. Nếu sản phẩm đã tồn tại trong giỏ, cập nhật số lượng.
     2. removeProduct($productId): Xóa sản phẩm khỏi giỏ hàng dựa trên productId.
     3. getTotal(): Tính tổng số tiền của tất cả các sản phẩm trong giỏ.
     4. displayCart(): Hiển thị danh sách sản phẩm trong giỏ và tổng số tiền cần thanh toán.