

# Dò bài cũ



1. Constructor là gì?

2. Đặc điểm của constructor

3. Tính bao đóng

4. Access modifier

# Constructor

- Là một **phương thức đặc biệt** được sử dụng để tạo đối tượng
- Có 2 cách khai báo
  - Constructor **mặc định** (không tham số)
  - Constructor **có tham số**

Đặc điểm của  
Constructor



# Đặc điểm của constructor



- **Trùng tên** với **class**, không có kiểu trả về
- Trong một **class** có thể có **nhiều constructor**.
- Mặc định không khai báo **constructor** thì JAVA **cung cấp constructor mặc định** không tham số.
- Có thể gọi 1 **constructor** khác nhưng phải đặt đầu tiên trong **body**.
- **this(xx)** gọi **constructor** của class hiện tại
- **super(yyy)** gọi **constructor** của **class** cha.

# Tính bao đóng



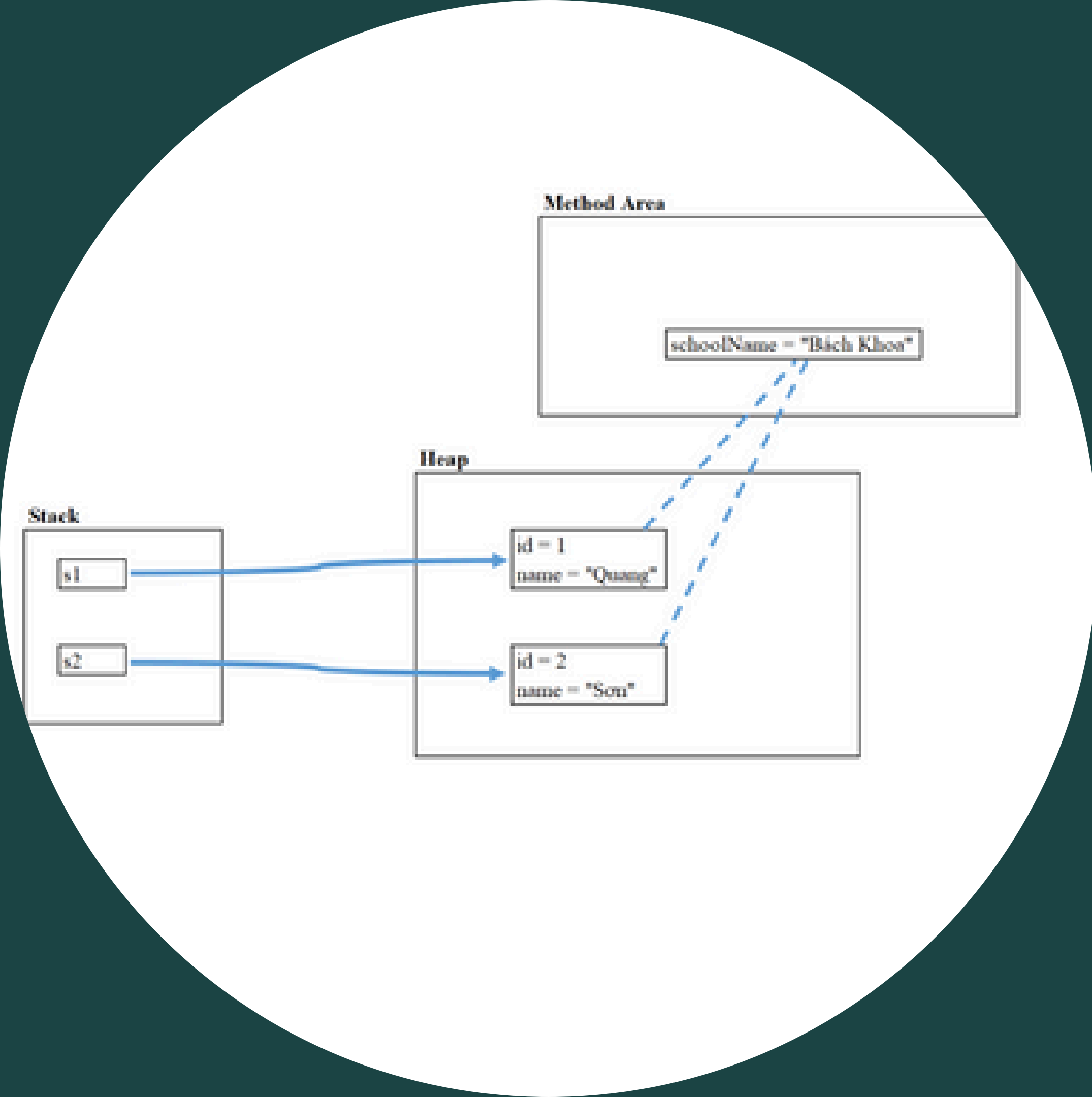
- Là kỹ thuật **che dấu** thông tin của đối tượng.
  - **Không** cho phép thay đổi **trực tiếp giá trị** thuộc tính của đối tượng
- Để đạt được tính **bao đóng**
  - **Access modifier**
  - **Getter, Setter**

# Access modifier



- Giúp xác định **mức độ truy cập** của lớp, constructor, biến, phương thức.

Access Modifier	Trong lớp	Trong package	Ngoài package bởi lớp con	Ngoài package
Private	Y	N	N	N
Default	Y	Y	N	N
Protected	Y	Y	Y	N
Public	Y	Y	Y	Y



# Static



# NỘI DUNG CHÍNH

- 1
- 2
- 3

Static là gì

Ràng buộc khi sử dụng static

Các loại biến



# Static là gì

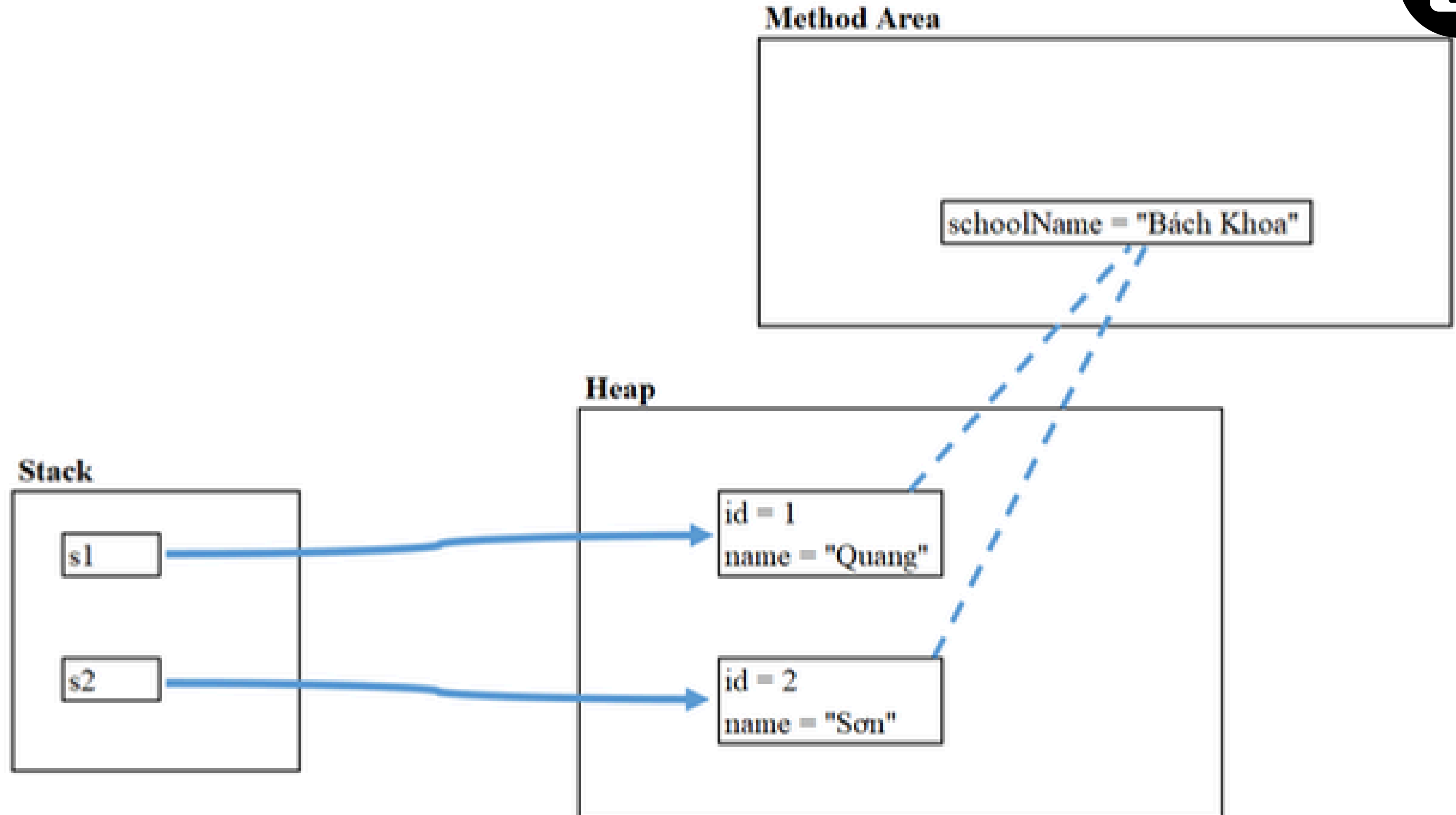
Cơ chế lưu trữ



- Là từ khóa dùng để **khai báo** thuộc tính và phương thức **thuộc** về **class**
- **Truy xuất** thành phần **static** (thuộc tính, phương thức) thông qua **tên class**, hoặc có thể thông qua **đối tượng**.
- Mục đích
  - Định nghĩa thuộc tính, phương thức **chung** của **toàn bộ đối tượng**
  - Tạo ra các lớp **tiện ích**.
  - Tiết kiệm bộ nhớ



# Cơ chế ưu trữ



# Ràng buộc khi sử dụng static



- **Phương thức static**
  - Chỉ **có thể** gọi phương thức và thuộc tính **static**.
  - **Không thể** gọi phương thức và thuộc tính **non-static**.
  - **Không thể** sử dụng **this** hoặc **super**
- **Phương thức non-static**
  - **Có thể** gọi phương thức và thuộc tính **static**.
- **Có thể** khởi tạo giá trị **biến static** thông qua **khối static**.

# Các loại biến



- Biến **toàn cục** (global, instance, đối tượng)
  - Khai báo trong một **class**, bên ngoài constructor, method và block {}
- Biến **cục bộ** (local)
  - Khai báo bên trong constructor, method hoặc block {}
- Biến **static**
  - Được khai báo bằng từ khóa static



DATABASE

**THANK  
YOU**