

 Cập nhật tháng 8 năm 2024

[Bài đọc] Từ khóa `static` và `final`

1. `static` là gì

- Từ khóa `static` dùng để khai báo thành phần chung cho tất cả các đối tượng (instances) của lớp, thay vì riêng lẻ cho từng đối tượng

2. Dùng `static` ở đâu?

- Bạn có thể dùng `static` cho:
 - Biến (`static variable`): Còn gọi là biến lớp (`class variable`)
 - Phương thức (`static method`): Gọi được mà không cần tạo đối tượng
 - Khối `static` (`static block`): Chạy khi lớp được nạp vào bộ nhớ
 - Lớp lồng nhau (`static nested class`)
- Ví dụ:



Student.java

```
public class Student {
    static String school = "Đại học ABC"; // dùng chung cho tất cả sinh viên
    String name;

    public Student (String name) {
        this.name = name;
    }

    public void show() {
        System.out.println(name + " - " + school);
    }

    public static void showSchool() {
        System.out.println(school);
    }
}
```



Main.java

```
public class Main {
    public static void main(String[] args) {
        Student sv1 = new Student("An");
        Student sv2 = new Student("Bình");

        sv1.show(); // An - Đại học ABC
        sv2.show(); // Bình - Đại học ABC

        Student.showSchool(); // Gọi mà không cần tạo đối tượng
    }
}
```

3. final là gì

- **final** nghĩa là không thay đổi được sau khi đã gán. Dùng để tạo hằng số, phương thức không bị override, hoặc lớp không bị kế thừa

4. Dùng final ở đâu?

- **Biến (final variable)**: Gán giá trị 1 lần duy nhất
- **Phương thức (final method)**: Không thể override trong lớp con
- **Lớp (final class)**: Không thể bị kế thừa
- Ví dụ:
final variable



Example.java

```
public class Example {  
    final int MAX = 100;  
  
    public void change() {  
        // MAX = 200; // Lỗi vì MAX là final  
    }  
}
```

final method



Example.java

```
class Cha {  
    final void xinChao() {  
        System.out.println("Chào từ lớp Cha");  
    }  
}  
  
class Con extends Cha {  
    // void xinChao() {} // Lỗi: không thể override phương thức final  
}
```

final class

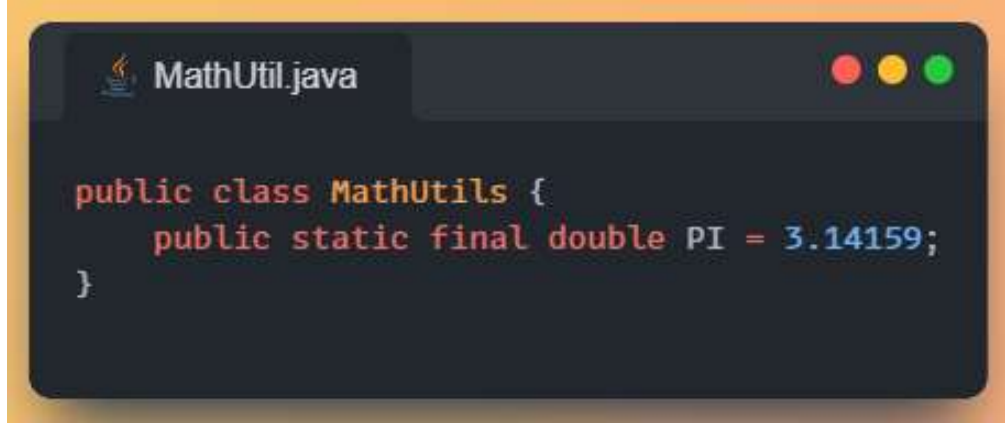


Example.java

```
final class HìnhTron {  
    // code ở đây  
}  
  
// class HìnhTronMoi extends HìnhTron {} // Lỗi: không thể kế thừa
```

5. Kết hợp sử dụng static và final

- Khi dùng cả hai, bạn sẽ tạo ra một hằng số cấp lớp:



6. Tổng kết

Từ khóa	Mục đích chính	Có thể dùng với
static	Chia sẻ giữa các đối tượng	biến, phương thức, khối, lớp lồng
final	Không thay đổi	biến, phương thức, lớp

Tài nguyên đọc thêm:

- static: <https://www.geeksforgeeks.org/static-keyword-java/>
- final: <https://www.geeksforgeeks.org/final-keyword-in-java/>

Danh sách các bài học

