



# Framework là gì?

Mentor: Lê Nhật Tùng



## 1. Giới thiệu chung

### *Định nghĩa của Framework*

Trong lập trình, **Framework** là **một tập hợp các thư viện, công cụ, mô hình thiết kế và quy tắc** mà lập trình viên có thể sử dụng để xây dựng và phát triển các ứng dụng, hệ thống hoặc phần mềm một cách nhanh chóng và hiệu quả.

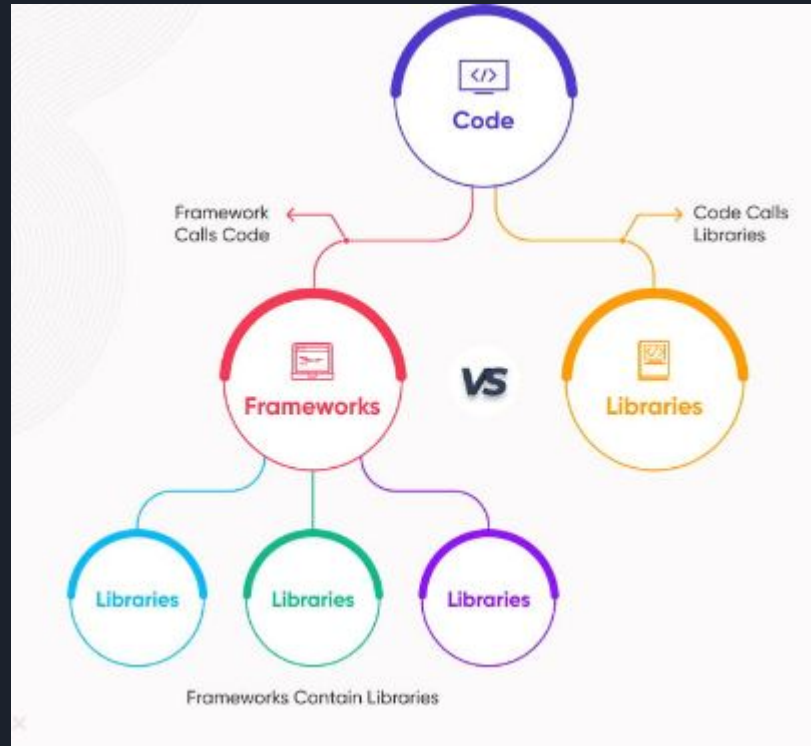


Làm sao để sắp xếp một mảng số nguyên tăng dần/ giảm dần?


```
public static void sortAscending(int[] array) {
    int n = array.length;
    for (int i = 0; i < n-1; i++) {
        for (int j = 0; j < n-i-1; j++) {
            if (array[j] > array[j+1]) {
                // Hoán đổi các phần tử nếu thứ tự sai
                int temp = array[j];
                array[j] = array[j+1];
                array[j+1] = temp;
            }
        }
    }
}
```

```
public static void sortAscending(Integer[] array) {
    List<Integer> list = Arrays.asList(array);
    Collections.sort(list);
}

public static void sortDescending(Integer[] array) {
    List<Integer> list = Arrays.asList(array);
    Collections.sort(list, Collections.reverseOrder());
}
```



<https://www.netsolutions.com/insights/what-is-a-framework-in-programming/>




Làm sao để lưu  
một đối tượng  
xuống CSDL??

```
import java.sql.Connection;
import java.sql.DriverManager;
import java.sql.PreparedStatement;
import java.sql.SQLException;

public class StudentDAO {
    private static final String DB_URL = "jdbc:mysql://localhost:3306/mydatabase";
    private static final String DB_USERNAME = "username";
    private static final String DB_PASSWORD = "password";

    public void saveStudent(Student student) {
        Connection conn = null;
        PreparedStatement stmt = null;

        try {
            conn = DriverManager.getConnection(DB_URL, DB_USERNAME, DB_PASSWORD);
            String query = "INSERT INTO students (student_id, full_name, score) VALUES (?, ?, ?)";
            stmt = conn.prepareStatement(query);
            stmt.setString(1, student.getStudentId());
            stmt.setString(2, student.getFullName());
            stmt.setDouble(3, student.getScore());
            stmt.executeUpdate();
        } catch (SQLException e) {
            e.printStackTrace();
        } finally {
            try {
                if (stmt != null)
                    stmt.close();
                if (conn != null)
                    conn.close();
            } catch (SQLException e) {
                e.printStackTrace();
            }
        }
    }
}
```



Làm sao để lưu  
một đối tượng  
xuống CSDL??

```
import org.hibernate.Session;
import org.hibernate.SessionFactory;
import org.hibernate.Transaction;
import org.hibernate.cfg.Configuration;

public class StudentDAO {
    private static SessionFactory sessionFactory;

    public void saveStudent(Student student) {
        Transaction transaction = null;
        try (Session session = getSessionFactory().openSession()) {
            transaction = session.beginTransaction();
            session.save(student);
            transaction.commit();
        } catch (Exception e) {
            if (transaction != null) {
                transaction.rollback();
            }
            e.printStackTrace();
        }
    }

    private SessionFactory getSessionFactory() {
        if (sessionFactory == null) {
            try {
                Configuration configuration = new Configuration().configure();
                sessionFactory = configuration.buildSessionFactory();
            } catch (Exception e) {
                e.printStackTrace();
            }
        }
        return sessionFactory;
    }
}
```

```

public void saveStudent(Student student) {
    Connection conn = null;
    PreparedStatement stmt = null;

    try {
        conn = DriverManager.getConnection(DB_URL, DB_USERNAME, DB_PASSWORD);
        String query = "INSERT INTO students (student_id, full_name, score) VALUES (?, ?, ?)";
        stmt = conn.prepareStatement(query);
        stmt.setString(1, student.getId());
        stmt.setString(2, student.getFullName());
        stmt.setDouble(3, student.getScore());
        stmt.executeUpdate();
    } catch (SQLException e) {
        e.printStackTrace();
    } finally {
        try {
            if (stmt != null)
                stmt.close();
            if (conn != null)
                conn.close();
        } catch (SQLException e) {
            e.printStackTrace();
        }
    }
}

```

```

public void saveStudent(Student student) {
    Transaction transaction = null;
    try (Session session = sessionFactory().openSession()) {
        transaction = session.beginTransaction();
        session.save(student);
        transaction.commit();
    } catch (Exception e) {
        if (transaction != null) {
            transaction.rollback();
        }
        e.printStackTrace();
    }
}

```

# Triển khai một API đơn giản với PYTHON

main.py



Save

Run

```
1 from flask import Flask, jsonify, request
2
3 app = Flask(__name__)
4
5 # Định nghĩa một route và phương thức xử lý
6 @app.route('/api/students', methods=['GET'])
7 def get_students():
8     # Lấy danh sách sinh viên từ CSDL hoặc bất kỳ nguồn dữ liệu nào khác
9     students = [
10         {'student_id': '001', 'full_name': 'John Doe', 'score': 8.5},
11         {'student_id': '002', 'full_name': 'Jane Smith', 'score': 9.0}
12     ]
13
14     return jsonify(students)
15
16 # Định nghĩa một route và phương thức xử lý
17 @app.route('/api/students', methods=['POST'])
18 def create_student():
19     # Lấy dữ liệu sinh viên từ request body
20     data = request.get_json()
21
22     # Lưu thông tin sinh viên vào CSDL hoặc bất kỳ nguồn dữ liệu nào khác
23     # ...
24
25     return jsonify({'message': 'Student created successfully'})
26
27 if __name__ == '__main__':
28     app.run()
--
```





# 1. Giới thiệu chung

## *Vai trò và lợi ích*

### Cung cấp cấu trúc và khung làm việc:

Framework cung cấp một cấu trúc và khung làm việc sẵn có cho việc phát triển phần mềm. Nó định nghĩa các quy tắc, mô hình và công cụ giúp người phát triển xây dựng ứng dụng một cách có hệ thống và có hiệu suất cao.

# What does software development speed depend on?

Complexity



Framework



Code Quality



People



Resources





## 1. Giới thiệu chung

### *Vai trò và lợi ích*

#### Tăng năng suất phát triển:

Framework cung cấp các thư viện, công cụ và tiện ích để giảm bớt công việc lặp lại và tăng năng suất phát triển. Người phát triển có thể tận dụng các chức năng đã được xây dựng sẵn, giảm thời gian và công sức cần thiết để triển khai các tính năng phổ biến.



Code Reusability



# 1. Giới thiệu chung

## *Vai trò và lợi ích*

### Chuẩn hóa và tái sử dụng mã nguồn:

Framework thường xác định các quy tắc và chuẩn mà người phát triển phải tuân thủ. Điều này giúp tạo ra mã nguồn dễ đọc, dễ bảo trì và dễ tái sử dụng. Người phát triển có thể sử dụng lại các thành phần, module và thư viện được xây dựng sẵn trong Framework để giảm thiểu việc viết mã từ đầu.

Login Page - Mozilla Firefox

Login Page



localhost:8080/springSecurityLoginOut/login

## Login with Username and Password

User:

irfan

Password:

....

Login



# 1. Giới thiệu chung

## *Vai trò và lợi ích*

### **Bảo mật:**

Framework thường có các cơ chế bảo mật tích hợp và kiểm tra tự động giúp ngăn chặn các lỗ hổng bảo mật và giảm rủi ro. Nó cung cấp các công cụ và thư viện cho việc kiểm thử, gỡ lỗi và kiểm tra hiệu năng để đảm bảo chất lượng ứng dụng.





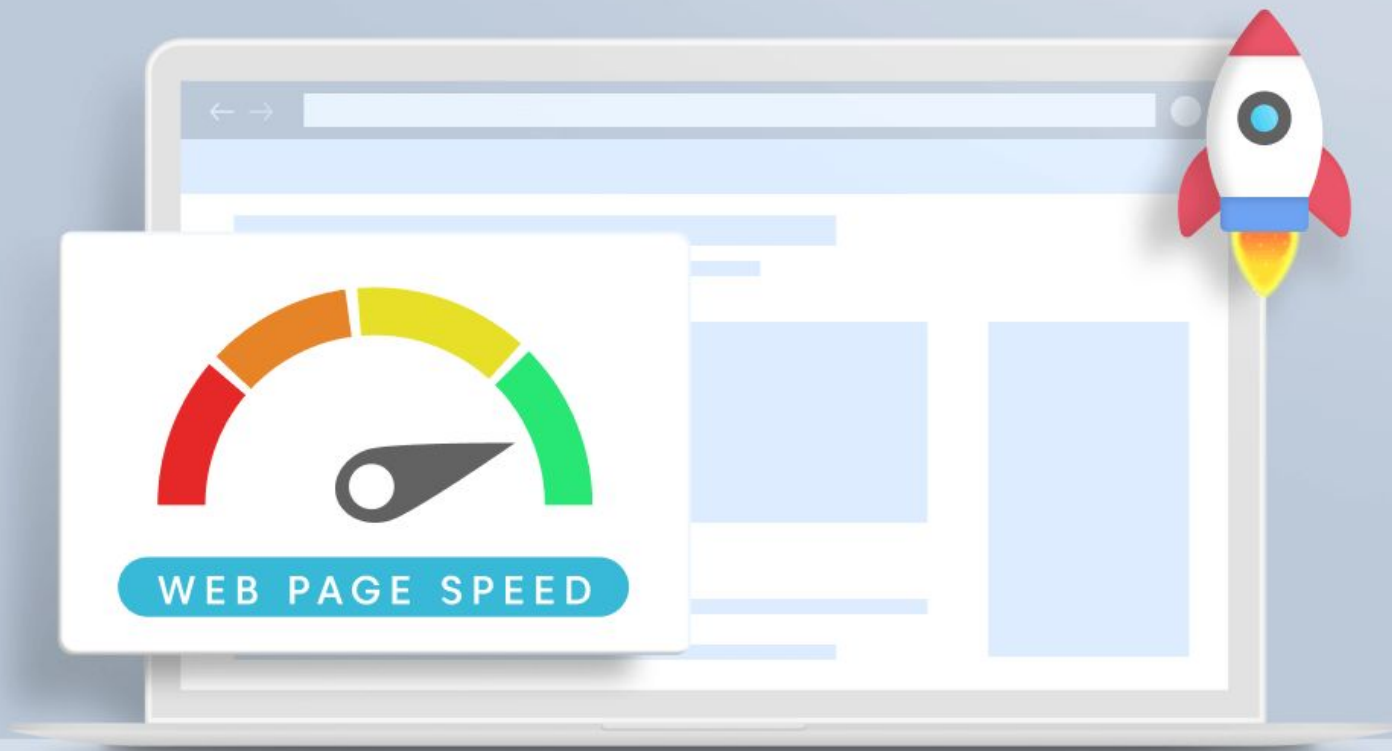


## 1. Giới thiệu chung

### *Vai trò và lợi ích*

#### Hỗ trợ từ cộng đồng và tài liệu:

Framework thường có một cộng đồng lớn và nhiệt tình xung quanh nó. Người phát triển có thể tìm kiếm hỗ trợ, chia sẻ kinh nghiệm và nhận được các tài liệu, ví dụ và hướng dẫn từ cộng đồng. Điều này giúp giải quyết các vấn đề phát triển một cách nhanh chóng và hiệu quả.



WEB PAGE SPEED



# 1. Giới thiệu chung

## *Vai trò và lợi ích*

Cải thiện hiệu suất và khả năng mở rộng:

Framework cung cấp các công cụ và kỹ thuật tối ưu hóa để cải thiện hiệu suất ứng dụng. Nó cung cấp khả năng mở rộng để xử lý

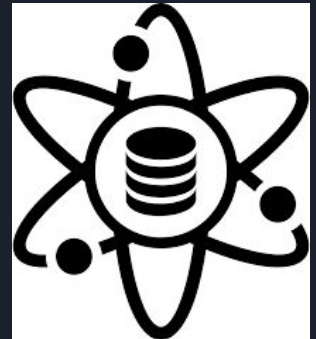
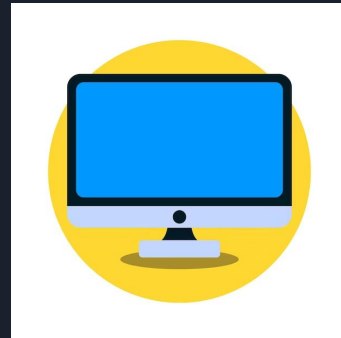


## 2. Các thành phần cơ bản của một Framework

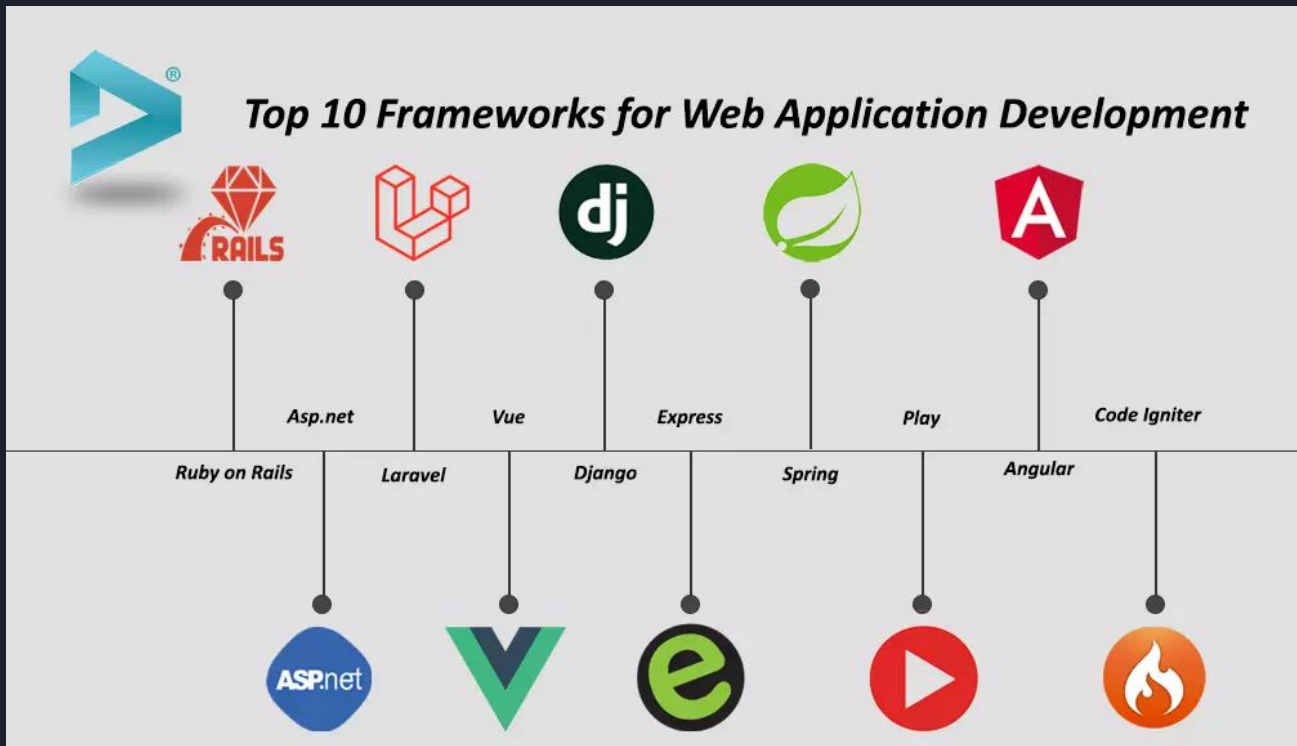
- Thư viện (Library)
- Mô hình thiết kế (Design pattern)
- Cấu trúc thư mục (Directory structure)
- Các công cụ hỗ trợ (Tooling)
- Cách thức hoạt động (Workflow)

### 3. Các loại Framework phổ biến trong lập trình

Web Frameworks	Mobile Frameworks
Desktop Frameworks	Data Science Frameworks

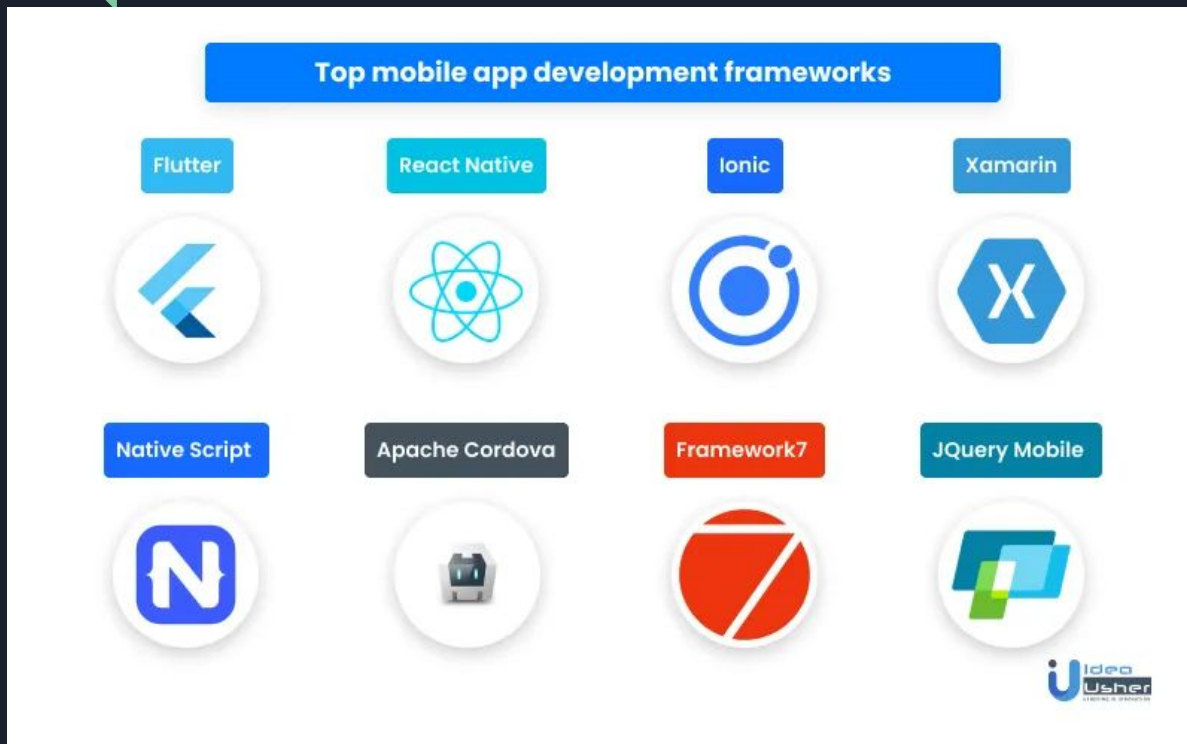


### 3. Các loại Framework phổ biến trong lập trình Web Frameworks



<https://www.decipherzone.com/blog-detail/top-web-frameworks>

### 3. Các loại Framework phổ biến trong lập trình Mobile Frameworks



<https://ideausher.com/blog/top-mobile-app-development-frameworks/>

### 3. Các loại Framework phổ biến trong lập trình Desktop Frameworks



Lazarus



X O J O

<https://medium.com/geekculture/10-tools-for-developing-desktop-applications-with-user-interfaces-f3806ef0d841>



### 3. Các loại Framework phổ biến trong lập trình Data Science Frameworks



**django**



APACHE  
**Spark**<sup>™</sup>



**Flink**



APACHE  
**STORM**<sup>™</sup>  
Distributed • Resilient • Real-time



scikit  
**learn**



APACHE  
**kafka**<sup>™</sup>  
A distributed streaming platform



**hadoop**



**Flask**  
web development,  
one drop at a time

<https://www.kdnuggets.com/2017/05/frameworks-offer-data-scientists-programming-languages-lack.html>



## 4. Cách lựa chọn Framework phù hợp

A. Yêu cầu dự án

B. Các yếu tố cần xem xét khi lựa chọn

1. Hỗ trợ cộng đồng
2. Hiệu năng và khả năng mở rộng
3. Tính di động và đa nền tảng
4. Thời gian và nguồn lực

## 5. Cách bắt đầu học/ sử dụng Framework

### 1. Xác định mục tiêu

Giúp bạn tập trung vào những kiến thức và kỹ năng cần thiết để đạt được mục tiêu của mình.

### 2. Tham gia lớp học, nghiên cứu tài liệu

Tìm hiểu thông qua tài liệu hướng dẫn, tài liệu chính thức, sách, video hoặc các nguồn tài liệu trực tuyến khác; hiểu cách Framework hoạt động, các khái niệm, cú pháp và các tính năng quan trọng.

### 3. Thực hành

Tạo các ví dụ đơn giản, theo các hướng dẫn và ví dụ trong tài liệu. Thử các tính năng và chức năng khác nhau của Framework để hiểu rõ hơn về cách sử dụng.

## 5. Cách bắt đầu học/ sử dụng Framework

### 4. Dự án thực tế

Hãy thử xây dựng một dự án thực tế sử dụng Framework.

Dự án có thể là một ứng dụng đơn giản, Trong quá trình xây dựng, bạn sẽ gặp phải các thách thức và học hỏi từ những vấn đề thực tế.

### 5. Tham gia cộng đồng và tìm kiếm sự giúp đỡ

Tham gia diễn đàn, nhóm người dùng, trang web chia sẻ kiến thức hoặc các cộng đồng trên mạng xã hội.

### 6. Tiếp tục học tập và nâng cao kỹ năng

Cập nhật kiến thức của mình về Framework, theo dõi các bản cập nhật mới và học các phương pháp tiếp cận mới.



# CÂU THẦN CHÚ

1. Bắt đầu!
2. Không bỏ cuộc!
3. Giữ cho mọi thứ đơn giản!