

---

# WHAT IS FRAMEWORK?

A framework, or software framework, is a platform for developing software applications. It provides a foundation on which software developers can build programs for a specific platform.

**For example**, a framework may include predefined classes and functions that can be used to process input, manage hardware devices, and interact with system software. This streamlines the development process since programmers don't need to reinvent the wheel each time they develop a new application.



# DIFFERENCE BETWEEN A LIBRARY AND A FRAMEWORK

## Library

- Library is a set of reusable functions used by computer programs.
- You are in full control when you call a method from a library and the control is then returned.
- It's incorporated seamlessly into existing projects to add functionality that you can access using an API.
- They are important in program linking and binding process.
- Example: jQuery is a JavaScript library that simplifies DOM manipulation.

## Framework

- Framework is a piece of code that dictates the architecture of your project and aids in programs.
- The code never calls into a framework, instead the framework calls you.
- It cannot be seamlessly incorporated into an existing project. Instead it can be used when a new project is started.
- They provide a standard way to build and deploy applications.
- Example: AngularJS is a JavaScript based framework for dynamic web applications.



# 10 MOST POPULAR WEB FRAMEWORKS

The **framework** plays an essential role in the field of **web development** and **web applications**. So you must know any of these frameworks if you are thinking of working in the web development field.

There are many frameworks available in the market. Every framework has its own advantages and disadvantages. So choosing the best framework on which you should work can be difficult.

- **Django** - Python Framework
- **Spring** - Java Framework
- **Laravel** - PHP Framework
- **Rails Framework (Ruby on Rails)**
- **React JS Framework**
- **Angular JS Framework**
- **Express Framework**
- **Vue.js Framework**
- **JQuery**
- **Flask** - Python Framework



# SHOULD YOU USE A FRAMEWORK?

## Five Advantages of Using a Framework

- I. Rapid Development
- II. A Framework Makes Your Application More Secure
- III. Easier Maintenance
- IV. Stronger Teamwork
- V. The Community Is There for You

## Five Disadvantages of Using a Framework

- I. Slower Execution
- II. General Solutions for Specific Problems
- III. A Framework Takes Time to Learn and Master
- IV. Limited Visibility and Control
- V. A Framework Introduces an External Dependency



# WHEN TO AND WHEN NOT TO USE A FRAMEWORK

## General needs vs. specific functions

Besides a framework, there are other solutions that are available to develop both Web sites and Web applications: CMSs (Content Management System) and their supplemental modules, as well as packaged professional solutions (CRM, e-commerce solutions, etc.). If their native features line up perfectly with your immediate and future needs, no question about it: Go for it! There is no point in reinventing the wheel.

On the other hand, when there are specific needs, specific business rules or even the need to mix building blocks (content and e-commerce, for example), a framework solution is necessary: it allows an application to be developed that perfectly matches your current needs, while still being innovative.

***“ A framework can be tailored to all requirements but that is not always necessary, especially when the requirements are of a more generic nature. Before making your selection, map out your current and future requirements in detail, and then compare the various solutions available on the market. ”***