

Suppose you're a web developer in a company and need to decide if you'll use vanilla (plain) Python for a project, or a framework like Django instead. What are the advantages and drawbacks of each?

Vanilla Python - The advantage is that it's versatile and can be used for different types of features, particularly math, data science, and machine learning. The disadvantage is that it's very basic so it would take more manual coding to implement basic web development tasks.

Django - Better for an app that needs to be developed quickly. Django comes with many built in web development features that save the developer a lot of time, and allows for quick scalability with minimal code rewriting/tweaking. It is best for large applications or ones that need CDN support. Django is very strict about structure so there's no way to stray from its rules or to make up your own, or to pick and choose certain features. Django is a better choice for large applications that actually need the tools that it offers.

In your own words, what is the most significant advantage of Model View Template (MVT) architecture over Model View Controller (MVC) architecture?

The best advantage of MVT is that it saves the developer tons of time by handling the controller output on its own rather than requiring the developer to manually write it. It also allows for code reuseability.

What do you want to learn about Django?

What do you want to get out of this Achievement?

Where or what do you see yourself working on after you complete this Achievement?

I want to learn how I can use Django to host user generated content and to draw patterns and make recommendations based off of it. After I finish this achievement, I want to make a simple personal diet app that I can use to log calorie and macro information, and have a feature that makes recipe recommendations based off of nutritional goals and dieting habits.