What is Django?

What we can do with Dajngo?

Why Django?

Django is a free open source web application framework. There are two important terms:

The first one is free open source and the second one is web application framework.

**Open source:** is a project which is available to all and anyone can contribute

to it. So it can be a framework it can be a library or it can be a program.

So even you can contribute to Django through some financial donation or maybe giving input through codes.

**Web application framework:**

What is *framework*? Framework is something which gives structure to a system.

So, *web application framework* is going to give us structure a pre-defined plan for our

web application.

What a web application needs?

So web application might need to control your URL'S, maybe handling your database, user authentication, forms that is taking input and giving output.

So there are several common tasks that we need with web application and the framework is a structure, a plan that helps us to execute these common tasks easily.

That's where Django come into picture.

So if we are using any framework it is much easier to develop anything because we know a structure, we know a plan, how to execute certain things.

And that is why Django has so much importance in the industry.

There are several common tasks that Django handles very well.

These can be some repetitive tasks or these can be simple tasks.

The next important point is MVC and MVT.

That is model view controller or model view template for Django.

For any other framework such as for PHP, for Java, for JavaScript all of them

follow MVC, whereas Python and Django follow MVT that is model view templates.

So, there is an architecture which Django follows and we call it MVT.

Now Django is not the only framework for Python:

We have Flask

We have WEB2PY

We have CherryPy

We have Bottle

And there are tons of other different frameworks but they are low level web development framework. That means they have several other use cases.

Some of them are quite useful while creating API. Some are quite useful for doing certain tasks.

*Whereas Django is the only framework which can handle heavy task altogether.*

What is Django?

Django is a free open source web application framework.

What we can do with Django?

We can actually build back-end of a website.

That means it is a web framework so we can design back-end of a website.

Website is divided into three parts.

*Frontend, backend & database:*

Front end means how the website is going to look.

Html, css and JavaScript - basic knowledge of these three can help you

a lot to create some static pages without any functionality just for the view purpose.

There might be certain animation color of a website, font and such.

For high end part you might go with different JavaScript frameworks like Angular, Vue, React and lot more.

All the functionalities of a website is controlled by backend. Here is a place where Django comes into picture. And if we are able to control complete Django, we can control complete backend.

So, backend means all the functionalities and the third one is database.

How you are going to store all the data that you are getting from a user or you are going to display to a user. Everything is stored in a database.

For frontend we are going to use html, css & javascript with the help of bootstrap.

For backend, we will use Django.

And for Database we are going to use SQLite3, which is an inbuilt database with Python.

Since we are going to use Django, everything would be quite easy and smooth to integrate.

Now coming back to our question number three, why Django?

Some really strong and onto the point reason, the most important one is that Django is fast, the development phase of Django is quite fast.

The second one is :

There are several libraries that are auto built with Django that makes your development quite smooth.

The third one is that it is secure.

The importance of data is really high. Django provides a secure application, keeps your website and your data safe.

And that's where Django has high advantage. There are regular security updates with every version of Django.

Django is around for 11 years and it has grown really well during these eleven years.

There are some significant improvements which is admin panel by default.

In any other framework you have to build your own admin panel from scratch. Whereas Django delivers a default admin panel which you can use with your application even your smallest application is going to have admin panel by default and so you don’t have to code for it separately.

So this is going to save a lot of time and this is one of the most important features.

Next important feature is library support. If you want to work with API - there is Django

REST framework. If you want to work with CMS - there is fractal, mezzanine.

If you want to work with some messaging app there are several other libraries.

You can work with maps, you can work with payment and all of them have their different liabilities.