

Q:1

Why Django should be used for web-development?

Explain how you can create a project in Django?

Python is a powerful high-level programming language that can be possibly used in a lot of fields. These fields can range from data science to automation and web development. It also has amazing libraries and frameworks that include pandas, NumPy, PyTorch, selenium, OpenCV, bottle, pyramid, flask, etc. One such framework in Python that is used for web development majorly is Django.

Django is a full-stack web framework that is open-sourced and follows an MVT (Model View Template) type of architecture. It is comprised of a set of components and modules that aids in faster development and is used by a few of the world's top companies like Instagram, Mozilla, Spotify, Quora, YouTube, Reddit, Pinterest, Dropbox, bitly, Google, Disqus, etc. Today we are going to discuss why is Django is the most loved, preferred, and believed to be the best framework for web development. We will look at its features one by one and understand why it is so popular, so let's get started!

firstly, Django is based on Python programming language which is comparatively easier as compared to other high-level programming languages like Java or C++. It includes pluggable modules and libraries which reduce the time taken in development to a huge extent as you are not writing code from scratch, but rather using already available code.

Q:2

How to check installed version of django?

Django is one of the most popular framework for creating web applications. Its initial release was back in 2005, and since then, more than 20 stable versions have been released. Having that many versions available to choose from, we often require methods to check our version in Django, either for deployment or dependency compatibility purposes. So how to check the Django version?

- Django version can be checked using multiple approaches.

1. Using the django-admin command line (terminal commands)

```
$ django-admin version
```

```
4.0.3
```

```
$ python manage.py version
```

```
4.0.3
```

```
$ python -m django version
```

```
4.0.3
```

2. Using django module (Python console commands)

```
>>> import django
```

```
>>> django.VERSION(4, 0, 3, 'final', 0)
```

Q:3

Explain what does django-admin.py make messages command is used for?

Manage.py in Django is a command-line utility that works similar to the django-admin command. The difference is that it points towards the project's settings.py file. This manage.py utility provides various commands that you must have while working with Django. Some of the most commonly used commands are –

`python manage.py startapp`

`python manage.py makemigrations`

`python manage.py migrate`

`python manage.py runserver`

Interestingly we can create our own Custom Management Commands to fulfill a wide variety of requirements ranging from interacting with our application using the command line to serve as an interface to execute Cron Jobs. We are going to create a custom Management Command which gives us the stats or metrics of new articles published, comments on those articles on a particular day.

Q:4

What is DjangoURLs? Make program to create django ursl.

URL is a path through which a specific web-based application and one particular page in that web application can be reached. So for any web-oriented application setting these url paths is a very key necessity. The same applies to Django also, So from a Django perspective setting the necessary URL's are handled in the urls.py section of the framework and there are several techniques through which the URL's are maintained through the application. This information on how classifiable Django handles its URL's and what are the techniques involved for keeping these URL's assembled are described below.

How to Create a Django URL?

How to create a Django url is given below:

- 1. Creating a url USING PATH()**
- 2. Creating a Django URL USING RE_PATH()**
- 3. Creating a Django url Through INCLUDE**
- 4. Setting Converters in URLS**
- 5. Creating a Django URL Through URL()**
- 6. Django URL Error Codes**

Q:5

What is queryset ? write program to create a new post object in database:

Once we've created our data models, Django automatically gives us a database-abstraction API that lets us create, retrieve, update and delete existing objects. This is very useful and time saving.

Current implementation

Make sure to have the project up and running and let's get started by checking our views.py file and the class IndexView inside the main app directory:

```
class IndexView(TemplateView):  
    template_name = 'index.html'  
    def get_context_data(self, **kwargs):  
        context = super(IndexView,  
self).get_context_data(**kwargs)  
        context['blogposts'] = BlogPost.objects.all()  
        return context
```

We have already used the Django QuerySet API! Yes, since we use the BlogPost.objects.all() we are basically retrieving all BlogPost objects and saving it to the context so we can loop through all the objects in our template file.

Q:6

Mention what command line can be used to load data into django?

Django is a robust high-level web framework that comes with a large amount of usability and inbuilt functionality right out of the box. With simple steps to start writing clean and powerful code, it uses the Model View Template structure.

Many advocates that the real power behind Django lies in its inbuilt Object-relational Mapping (ORM) allow developers to code “models” (tables) as class objects through object-oriented programming. These class objects represent the database structure and can be mapped to a CREATE statement in SQL. This creates the defined models as tables in an SQL lite database without having to write complex SQL code.

One of the biggest benefits of using Django database models is the interchangeability between SQL databases. The same models can be used when switching between an SQL lite database for your development environment and a PostgreSQL database for production.

Even more so, the ability to code database statements in python code is useful when creating new entries inside a database. However, before doing this, first, we must define the database tables.

Q:7

Explain what does `django-admin.py` make messages command is used for?

To do this, add a `management/commands` directory to the application. Django will register a `manage.py` command for each Python module in that directory whose name doesn't begin with an underscore. For example:

`polls/`

`_init_.py`

`models.py`

`management/`

`_init_.py`

`commands/`

`_init_.py`

`_private.py`

`closepoll.py`

`tests.py`

`views.py`

In this example, the `closepolls` command will be made available to any project that includes the `polls` application in `INSTALLED_APPS`.

The `_private.py` module will not be available as a management command.

