

1. what does the 'yesno' filter do in templates?

Ans: It used to display one of two specific string base value on give value. By using this filter the value be showed dynamically.

`{{ value | yesno : 'Yes, No, Maybe' }}`

2. what is the difference between path() and re\_path( ) ?

Ans: Path() is used for simple, readable, straight pattern url.

`path('teacher|', teacherpage, name='teacher')`

re\_path() it is used when we need to use regular expressions to define one more specific or complex, flexible URL patterns.

from django.urls import path

```
urlpatterns = [
    path('job', job, name='job'),
    re_path(r'^example/(?P<slug>[w-]+)$', view),
]
```

Q. What is the purpose of a URL in Django?

- to map urls to view
- connect url to a specific view and handle the corresponding HTTP request.

```
path('Student/', studentpage, name='student')
```

4. Explain the django admin site?

Ans: admin site is a built in web based interface that is powerful and customizable.

- creating, editing and deleting data
- viewing data for a specific model
- searching and filtering data
- Registering model in admin site

5. which file is responsible for defining the database settings for a Django project?

Ans: The settings.py file is responsible for this.

```
DATABASE = {
    'default': {
        'ENGINE': 'django.db.backends.sqlite3',
        'NAME': BASE_DIR / 'db.sqlite3',
    }
}
```

06. what does the date filter do?

Ans: Date filter is used to format dates according to require format. Showing the date in dynamically

```
{% my_date | date : "Y-m-d" %}
```

07. what is a Django form and its purpose?

Ans. It is a built in form in django.

- create form by using django form
- display the form
- validates data
- Render Html form easily and input fields
- Save data to Models

```
from django.contrib import forms  
class jobForm(forms.Form):  
    name=forms.CharField(max_length=50)
```

08. How is a form submitted in a templates?

Ans : In Django Form in a templates  
Form submitted 'POST' method is used.

```
<form method='POST', action="">  
    {%- csrf_token %}  
    {{ form.as_p }}  
    <button type='submit'> Submit </button>  
</form>
```

09. What is a static file URL ?

Ans. Static files like CSS, javascripts,  
image files are stored in static folder.

```
{% load static %}  
<link rel='stylesheet' type='css' href='{%  
    static "css/style.css" %}'>
```

10. what is the purpose of serializers in DRF ?

Ans :

- convert complex model data to JSON, XML
- save data to Models

11. which file is responsible for defining the database settings for Django project?

Ans. settings.py file is responsible for defining the database settings for django project.

12. where should store static files in Django?

Ans. During development in Django, STATICFILES\_DIRS setting in settings.py to additional directories for static files.

STATIC - STATIC\_URL = '/static/'  
STATICFILES\_DIRS = [BASE\_DIR / "static", ]

13. How is a URL patterns in urls.py file?

Ans. In urls.py -

```
from django.urls import path  
from jobapp import views
```

```
urlpatterns = [
```

```
    path('home/', homepage, name='home')
```

14. what is the purpose of reverse() function:

Ans. It create URLs dynamically along with that it helps to avoid hard coding from the source code.

15. What does the 'date' filter do in Django?

Ans: In Django templates, the length filter is used to get the length of variable.  
- works with string, list, Queryset etc and return the Queryset.

{% with list = [1, 2, 3, 4, 5] %}  
The length of the list is: {{list|length}}  
{% endwith %}

16. How do you create a new database record (object) using Django ORM?

Ans: For creating a new database record (object) using Django's ORM (Object-Relational-Mapping). The steps are given below-

1. import the model class which represents the database table.

```
from django.db import models  
class JobModel(models.Model):  
    name = models.CharField(max_length=50)  
    salary = models.CharField(max_length=100)
```

2. create an instance of the Model class and call the save() method on the instance

```
job-app  
from jobModel.models import JobModel  
new_data = JobModel.objects.create(  
    name='Rahim',  
    salary='20K'  
)  
new_data.save()
```

17. What is the role of get\_context() method?

Ans. In Django, get\_context() method is a Django's widget classes.

- It helps to prepare the context dictionary for rendering the widget's template.
- It collects all necessary information such as
  1. widget name (name)
  2. current value (value)
  3. HTML attributes (attrs)
- Returns the data in a structure dictionary format.
- The templates uses this context to generate the widget's HTML output.

```
def get_context(name, value, attrs):  
    context = {  
    }  
    return context
```

18. what is the role of 'is\_authenticated' attribute in Django user authentication?

Ans: This attribute is a built-in attribute of the User object provided by the authentication system.

- This attribute is used to check the user is authenticate or not.

if request.user.is\_authenticated:

else:

19.

⇒ `get()` → specific or single data

⇒ `filter()` → One type data from model / multiple data

### ① `get()` and `filter()` use

Ans: From Django Model `get()` ~~can~~

can be used for getting specific

or single data . and `filter()`

`Queryset` is used for finding  
multiple record on ~~as~~  
One type data from Model.

④ filter().delete() command can be used

deleting multiple records.

Model.objects.filter().delete()

④ makemigrations : It creates a migrate

file based on the changes of Model.  
It doesn't affect data schema.

④ Migrate : make the changes of are

it can say that it applies migrations files into database. It is affect our data system.

python manage.py makemigrations

python manage.py migrate

Q) For set up a database in Django :

```
DATABASE = {
    'default': {
        'ENGINE': 'django.db.backends.sqlite3',
        'NAME': BASE_DIR / 'db.sqlite3',
    }
}
```

Q) The main features of Django

1. MVT - Architecture (Model - View - Template)
2. Built-in Authenticate system
3. Django Form
4. Admin Interface
5. Strong security (CSRF, SQL Injection, clickjacking, XSS or cross site scripting)

## ■ Strong Security

1. CSRF → Cross Site Request Forgery
2. XSS → Cross Site Scripting
3. SQL injection protection
4. Clickjacking protection

## ■ The role of settings.py in Django project

- connecting database
- install apps
- set up custom user
- set up static file

④ Django ORM means Object relational Mapping.

=> db.site queryset is ORM

Model.objects.all()

Model.objects.filter()

⑤ The role of 'is-authenticate' attribute

If user.is\_authenticate:  
(it returns True)

It checked a user logged in or not.

It also check user's authenticity.

if yes then it return True , otherwise  
it return False .

def view (request):

if request.user.is\_authenticated:

True

else:

False

## ④ Django Model in context of web

development it represents the database tabel table. Django Model is a class and make relationship with another Model. auto\_now means it update the timestamp

auto\_now\_add means it update the timestamps when use of create

### ⑤ The save() function in Django :

It save the model instance when

it created or updated. It used to

persist changes to a model instance

in database. It is a method of

Django's model class.

save() function

1. create new records

2. updates existing Records

Q The purpose of models.py file

Ans: It represents the database table.

⇒ setup relationship with other model (foreignkey)

class Modelname(models.Model)

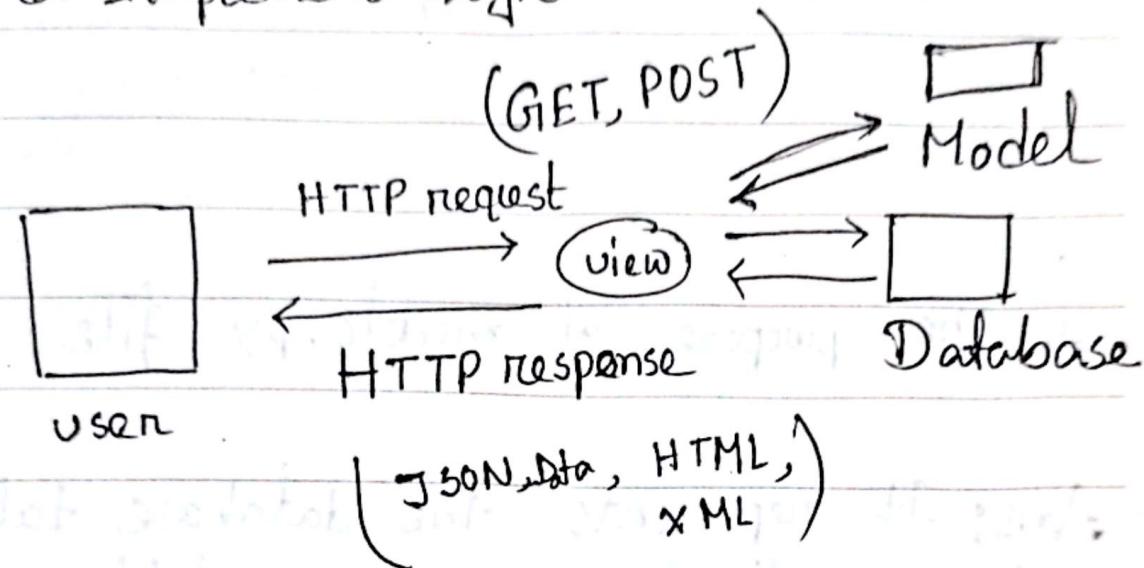
name = Models.CharField('')

Q what is a view in Django?

1. It received HTTP Request (GET, POST)  
and return HTTP Response  
(JSON, HTML Template)

## 2. Interact with database

### 3. Implement logic



■ The purpose of delete() function is

to delete the model instance.

For deleting multiple data filter( )

queryset can be used.

```
data = Model.objects.filter().delete()
```

The using purpose of reverse() URL function in Django is that -

it create URLs dynamically along

with that it avoid hard coding from

source code.

example :

context = 123

myurl = reverse('StudentPage', kwargs=[context])

DATE / /

OM OT OW OT OF OS OS

→ (regular)

NOTES

④ path() & re-path() Straight pattern  
⇒ path() is used for simple and URL  
⇒ re-path() is used when we need to  
use regular expressions to define  
more specific or complex

path(' ', function\_name, name=''),

re\_path(r'^example/(?P<slug>[w-]+)\$',  
dynamic\_view),

④ urls.py

any views or function work with  
the url re pattern.

## ④ Reverse() Function in url pattern :

- helps to avoid hard coding url
  - dynamical url generated to reduce hard coding
  - makes code more flexible, maintainable especially url patterns
- ⇒ Reverse() returns the url string