

Answer the following

1. What are views in Django? State the types of Views in Django.

Views in Django are Python functions or classes that receive web requests and return web responses. A view function controls what data is displayed and how it is presented to the user, usually via an HTML template.

Types of Views in Django:

1. Function-Based Views (FBVs):

- **These are views written as standard Python functions.**
- **Suitable for simple logic and straightforward handling of requests.**

```
from django.http import HttpResponseRedirect
```

```
def my_view(request):  
    return HttpResponseRedirect("Hello, world!")
```

Class-Based Views (CBVs):

- **These are views written using Python classes.**
- **They allow more modular, reusable, and extensible code by providing methods for handling HTTP verbs (get, post, etc.).**

```
from django.views import View  
from django.http import HttpResponseRedirect
```

```
class MyView(View):  
    def get(self, request):  
        return HttpResponseRedirect("Hello from a class-based view!")
```

2. What is the difference between render() and redirect() functions in Django views?

Feature `render()` `redirect()`

Purpose Renders an HTML template with a context Redirects to another URL

Returns <code>HttpResponse</code> with the rendered template	<code>HttpResponseRedirect</code> to a specific URL
--	---

Usage To display a page To navigate the user to a different route

Example <code>render(request, 'template.html', context)</code>	<code>redirect('home')</code> or <code>redirect('/home/')</code>
--	---

3. You can pass URL parameters using the **URLconf (URL configuration)** and capture values with path converters or regular expressions.

Example using `path()`:

How do you pass URL parameters to a Django view

urls.py

```
from django.urls import path
```

```
from . import views
```

```
urlpatterns = [
```

```
    path('article/<int:id>/', views.article_detail, name='article_detail'),
```

```
]
```

```
# views.py
```

```
def article_detail(request, id):
```

```
    return HttpResponse(f"Article ID: {id}")
```