**PEJELASAN KODING**

* **MODEL**

|  |
| --- |
| # Create your models here.  class jadwalModels(models.Model):  Tanggal = models.CharField(max\_length=70)  Imsyak = models.TimeField()  Subuh = models.TimeField()  Terbit = models.TimeField()  Duha = models.TimeField()  Zuhur = models.TimeField(default='05:')  Asar = models.TimeField()  Magrib = models.TimeField()  Isya = models.TimeField()  def \_\_str\_\_(self):  return self.Tanggal |
| Model ini memiliki beberapa field yang didefinisikan menggunakan modul models dari Django. Tipe CharField digunakan untuk field Tanggal, yang dapat menampung string dengan panjang maksimum 70 karakter. Tipe TimeField digunakan untuk field lainnya, yang menyimpan waktu dalam sehari. Atribut default diatur untuk field Zuhur, dengan nilai default adalah '05:'.  Metode \_\_str\_\_ adalah metode khusus dalam kelas Python yang mendefinisikan bagaimana instance dari kelas tersebut harus direpresentasikan sebagai sebuah string. Dalam hal ini, metode ini mengembalikan nilai field Tanggal sebagai sebuah string, yang akan digunakan sebagai representasi default dari instance model ini ketika ditampilkan di antarmuka admin Django atau di konteks lain di mana representasi string diperlukan. |

* **URL**

|  |
| --- |
| from django.contrib import admin  from django.urls import path, include  from imsyakApp.views import \*  from rest\_framework import routers  router = routers.DefaultRouter()  router.register('Jadwal\_Imsyak', jadwalViews)  urlpatterns = [  path('admin/', admin.site.urls),  path('', include(router.urls))  ] |
| Kode yang Anda berikan adalah file urls.py yang terdapat dalam sebuah aplikasi Django, yang mengatur konfigurasi routing URL.  Pertama, kita mengimpor modul admin dan path dari Django, serta modul jadwalViews dari aplikasi imsyakApp dan modul routers dari rest\_framework.  Selanjutnya, kita membuat sebuah objek router dari kelas DefaultRouter yang disediakan oleh Django REST Framework. Objek router ini digunakan untuk mengatur routing URL dari view jadwalViews dengan menggunakan method register.  Di dalam list urlpatterns, kita menambahkan dua path yaitu /admin/ dan path kosong ''. Path /admin/ digunakan untuk mengakses halaman admin Django, sedangkan path kosong digunakan untuk meng-include semua URL yang telah diregistrasi di router, sehingga semua URL yang terdaftar pada router akan tersedia di path ini. |

* **VIEW**

|  |
| --- |
| from django.shortcuts import render  from imsyakApp.models import jadwalModels  from rest\_framework.response import Response  from imsyakApp.serializers import jadwalSerializer  from rest\_framework import viewsets  # rest\_framework  from rest\_framework.decorators import api\_view  from rest\_framework.response import Response  # Create your views here.  class jadwalViews(viewsets.ModelViewSet):  queryset = jadwalModels.objects.all()  serializer\_class = jadwalSerializer |
| Pertama, kita mengimpor beberapa modul, yaitu render dari django.shortcuts, jadwalModels dari imsyakApp.models, Response dan viewsets dari rest\_framework, serta jadwalSerializer dari imsyakApp.serializers.  Kemudian, kita mendefinisikan sebuah kelas jadwalViews yang merupakan subkelas dari viewsets.ModelViewSet. Kelas ini menyediakan beberapa metode seperti list, create, retrieve, update, dan destroy yang digunakan untuk menangani permintaan HTTP GET, POST, GET detail, PUT, dan DELETE.  Pada atribut queryset, kita menetapkan semua objek dari model jadwalModels sebagai objek yang akan digunakan sebagai sumber data oleh view ini.  Pada atribut serializer\_class, kita menetapkan kelas jadwalSerializer sebagai kelas serializer yang akan digunakan untuk mengubah objek model menjadi representasi JSON dan sebaliknya. |

* **ADMIND**

|  |
| --- |
| from django.contrib import admin  from . models import jadwalModels  # Register your models here.  admin.site.register(jadwalModels) |
| Kode yang Anda berikan adalah file admin.py dari sebuah aplikasi Django, yang digunakan untuk mendaftarkan model pada halaman admin.  Pada kode tersebut, kita mengimpor modul admin dari Django dan model jadwalModels dari aplikasi saat ini.  Kemudian, kita mendaftarkan model jadwalModels pada halaman admin dengan menggunakan fungsi register pada objek admin.site. Setelah model didaftarkan, halaman admin akan menampilkan tabel untuk model jadwalModels beserta aksi yang dapat dilakukan seperti menambah, mengedit, dan menghapus objek model. |

* **SERIALIZER**

|  |
| --- |
| from imsyakApp.models import jadwalModels  from rest\_framework import serializers  class jadwalSerializer(serializers.ModelSerializer):  class Meta:  model = jadwalModels  fields = ['id','Tanggal','Imsyak','Subuh','Terbit','Duha','Zuhur', 'Asar','Magrib','Isya'] |
| Kode yang Anda berikan adalah file serializers.py dari sebuah aplikasi Django, yang berisi definisi serializer untuk model jadwalModels.  Kode tersebut mendefinisikan sebuah kelas jadwalSerializer yang merupakan subkelas dari serializers.ModelSerializer. Kelas ini menentukan bagaimana objek model jadwalModels akan diubah menjadi representasi JSON.  Pada atribut Meta, kita menentukan model yang akan digunakan sebagai sumber data (model = jadwalModels) dan atribut-atribut dari model yang akan dimasukkan ke dalam representasi JSON (fields = [...]).Dalam hal ini, kita memilih semua atribut (id, Tanggal, Imsyak, Subuh, Terbit, Duha, Zuhur, Asar, Magrib, dan Isya) sebagai atribut yang akan dimasukkan ke dalam representasi JSON. |