تکمیل مدل پکیج ها

آماده سازی مدل پست و ترسیم آن در سایت دی بی دیاگرام و ترسیم مدل یوزر برای دسترسی های پرایمری کی

ساخت اپلیکیشن جدید برای پست

class post(models.Model):

*# image*

*#author*

    title = models.CharField(*max\_length*=255)

    content = models.TextField()

*# tag*

*# category*

    coounted\_viwes = models.IntegerField(*default*=0)

    status = models.BooleanField(*default*=False)

    published\_date = models.DateTimeField(*null*=True)

    created\_date = models.DateTimeField(*auto\_now\_add*=True)

    updated\_date = models.DateTimeField(*auto\_now*=True)

معرفی متا کلاس برای مدل

*from* django.db *import* models

class post(models.Model):

    title = models.CharField(*max\_length*=255)

    content = models.TextField()

    coounted\_viwes = models.IntegerField(*default*=0)

    status = models.BooleanField(*default*=False)

    published\_date = models.DateTimeField(*null*=True)

    created\_date = models.DateTimeField(*auto\_now\_add*=True)

    updated\_date = models.DateTimeField(*auto\_now*=True)

    class Meta:

        ordering = ['-created\_date']

    def \_\_str\_\_(*self*):

*return* self.title

و دکوراتورadmin.site.register کاستومایز کردن پنل ادمین برای مدل پست و سپس رجیستر کردن با حالت

@admin.register(post)

class PostAdmin(admin.ModelAdmin):

    list\_display = ('id','title', 'status', 'created\_date')*# چه فیلد هایی از آن سطر در صفحه نمایش داده شوند*

    list\_filter = ('status', )*# افزودن یک گزینه فیلتر بر اساس استتوس ها و یا هر فیلد دیگری*

    search\_fields = ('title', )*# یک سرچ فیلد میگذارد که یک عبارت را مثلا در ستون تایتل ها سرچ میکند*

    empty\_value\_display = '-empty-' *# اگر فیلدی خالی بود خالی نشان نده و بجای آن عبارتش را نشان بده*

    fields = ('title', 'content', 'published\_date') *# وقتی بر روی یک سطر کلیک میکنیم کدام بخش ها قابل تغییر باشند*

    ordering = ('created\_date',  )*# بر اساس تاریخ ایجاد صفحه ادمین را سورت میکند*

*#admin.site.register(post, PostAdmin)*

حال وقتی مدل را ساختیم چند پست با وضعیت انتشار ترو و فالس ایجاد میکنیم و میخواهیم حالا در صفحه به دو صورت زیر پست های در حال انتشار را ایجاد میکنیم

<!DOCTYPE html>

<html lang="en">

<head>

    <meta charset="UTF-8">

    <meta http-equiv="X-UA-Compatible" content="IE=edge">

    <meta name="viewport" content="width=device-width, initial-scale=1.0">

    <title>Document</title>

</head>

<body>

    {% *for* post in posts %}

        {% *if* post.status == 1 %}

            {{post.title}}

        {% *else* %}

            {{post.created\_date}}

        {% *endif* %}

    {% *endfor* %}

</body>

</html>

با استفاده از فیلتراسیون مدل

def b\_home(*req*):

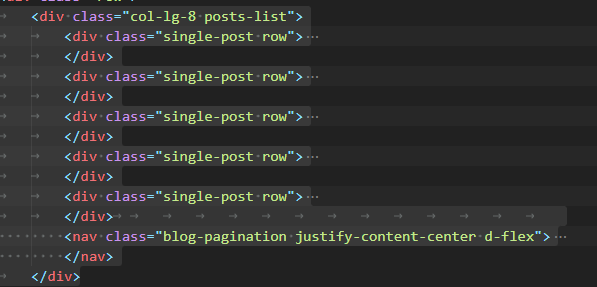
    posts=post.objects.filter(*status*=1)

    context = {

        'posts':posts

    }

قبل از اینکه پست ها را در صفحه اصلی نمایش دهیم در یک صفحه تستی نمایش میدهیم و سپس محتویات را به صفحه اصلی میبریم و سپس طبق فرم زیر آنرا میسازیم



urlارسال پارامتر از طریق