2.6.4\_models的设计和使用要点

# 以下为在默认设置下添加的设置

# 关于数据库配置的详细介绍请参照：2.6.5\_数据库设置和同步

settings.py:

DATABASES = {  
 'default': {  
 'ENGINE': 'django.db.backends.mysql',  
 'NAME': 'blogdb',  
 'USER': 'root',  
 'PASSWORD': '1234',  
 'HOST': '',  
 'PORT': '',  
 }  
}

# 自定义用户model  
AUTH\_USER\_MODEL = 'blog.User'

models.py:

# -\*- coding:utf-8 -\*-  
**from** django.db **import** models  
**from** django.contrib.auth.models **import** AbstractUser  
  
# Create your models here.  
# 用户模型  
# 第一种：采用继承的方式扩展用户信息（本系统）  
# 第二种：采用关联的方式扩展用户信息  
**class User**(AbstractUser):  
 avatar = models.ImageField(upload\_to='avatar/%Y/%m', default='avatar/default.png', max\_length=200, blank=True, null=True, verbose\_name='用户头像')  
 qq = models.CharField(max\_length=20, blank=True, null=True, verbose\_name='QQ号码')  
 mobile = models.CharField(max\_length=11, blank=True, null=True, unique=True, verbose\_name='手机号码')  
  
 **class Meta**:  
 verbose\_name = '用户'  
 verbose\_name\_plural = verbose\_name  
 ordering = ['-id']  
  
 **def** \_\_unicode\_\_(self):  
 **return** self.username  
  
# 标签  
**class Tag**(models.Model):  
 name = models.CharField(max\_length=30, verbose\_name='标签名称')  
  
 **class Meta**:  
 verbose\_name = '标签'  
 verbose\_name\_plural = verbose\_name  
 ordering = ['id']  
  
 **def** \_\_unicode\_\_(self):  
 **return** self.name  
  
# 文章分类  
**class Category**(models.Model):  
 name = models.CharField(max\_length=30, verbose\_name='分类名称')  
 index = models.IntegerField('显示顺序(从小到大)',default=999)  
  
 **class Meta**:  
 verbose\_name = '分类'  
 verbose\_name\_plural = verbose\_name  
 ordering = ['index', 'id']  
  
 **def** \_\_unicode\_\_(self):  
 **return** self.name  
  
# 文章模型  
**class Article**(models.Model):  
 title = models.CharField(max\_length=50, verbose\_name='文章标题')  
 desc = models.CharField(max\_length=50, verbose\_name='文章描述')  
 content = models.TextField(verbose\_name='文章内容')  
 click\_count = models.IntegerField(default=0, verbose\_name='点击次数')  
 is\_recommend = models.BooleanField(default=False, verbose\_name='是否推荐')  
 date\_publish = models.DateTimeField(auto\_now\_add=True, verbose\_name='发布时间')  
 user = models.ForeignKey(User, verbose\_name='用户')  
 category = models.ForeignKey(Category, blank=True, null=True, verbose\_name='分类')  
 tag = models.ManyToManyField(Tag, verbose\_name='标签')  
  
 **class Meta**:  
 verbose\_name = '文章'  
 verbose\_name\_plural = verbose\_name  
 ordering = ['-date\_publish']  
  
 **def** \_\_unicode\_\_(self):  
 **return** self.title  
  
# 评论模型  
**class Comment**(models.Model):  
 content = models.TextField(verbose\_name='评论内容')  
 date\_publish = models.DateTimeField(auto\_now\_add=True, verbose\_name='发布时间')  
 user = models.ForeignKey(User, blank=True, null=True, verbose\_name='用户')  
 article = models.ForeignKey(Article, blank=True, null=True, verbose\_name='文章')  
 pid = models.ForeignKey('self', blank=True, null=True, verbose\_name='父级评论')  
  
 **class Meta**:  
 verbose\_name = '评论'  
 verbose\_name\_plural = verbose\_name  
 ordering = ['-date\_publish']  
  
 **def** \_\_unicode\_\_(self):  
 **return** str(self.id)  
  
# 友情链接  
**class Links**(models.Model):  
 title = models.CharField(max\_length=50, verbose\_name='标题')  
 description = models.CharField(max\_length=200, verbose\_name='友情链接描述')  
 callback\_url = models.URLField(verbose\_name='url地址')  
 date\_publish = models.DateTimeField(auto\_now\_add=True, verbose\_name='发布时间')  
 index = models.IntegerField(default=999, verbose\_name='排列顺序(从小到大)')  
  
 **class Meta**:  
 verbose\_name = '友情链接'  
 verbose\_name\_plural = verbose\_name  
 ordering = ['index', 'id']  
  
 **def** \_\_unicode\_\_(self):  
 **return** self.title  
  
# 广告  
**class Ad**(models.Model):  
 title = models.CharField(max\_length=50, verbose\_name='广告标题')  
 description = models.CharField(max\_length=200, verbose\_name='广告描述')  
 image\_url = models.ImageField(upload\_to='ad/%Y/%m', verbose\_name='图片路径')  
 callback\_url = models.URLField(null=True, blank=True, verbose\_name='回调url')  
 date\_publish = models.DateTimeField(auto\_now\_add=True, verbose\_name='发布时间')  
 index = models.IntegerField(default=999, verbose\_name='排列顺序(从小到大)')  
  
 **class Meta**:  
 verbose\_name = u'广告'  
 verbose\_name\_plural = verbose\_name  
 ordering = ['index', 'id']  
  
 **def** \_\_unicode\_\_(self):  
 **return** self.title

index.html:

<div class="logo">  
 <h1>{{ SITE\_NAME }}</h1>  
 <p>{{ SITE\_DESC }}</p>  
</div>