课程列表页

子应用创建

```
python ../../manage.py startapp Course
```

数据模型创建

```
class CourseCategory(models.Model):
        """
        课程分类
        """
    name = models.CharField(max_length=64, unique=True,verbose_name="分类名称")

    class Meta:
        db_table="ly_course_category"
        verbose_name = "课程分类"
        verbose_name_plural = "课程分类"

    def __str__(self):
        return "%s" % self.name

class Course(models.Model):
    """
    专题课程
    """
    course_type = (
      (0, '付费'),
      (1, 'VIP专享'),
      (2, '学位课程')
    )
    level_choices = (
      (0, '初级'),
      (1, '中级'),
      (2, '高级')
    )
    name = models.CharField(max_length=128, verbose_name="课程名称")
    course_img = models.CharField(max_length=255,verbose_name="封面图片",blank=True,null=True)
    course_type = models.SmallIntegerField(choices=course_type,verbose_name="付费类型")
    brief = models.TextField(max_length=2048,verbose_name="课程概述",null=True,blank=True)
    level = models.SmallIntegerField(choices=level_choices, default=1,verbose_name="等级划分")
    pub_date = models.DateField(verbose_name="发布日期",auto_now_add=True)
    period = models.IntegerField(verbose_name="建议学习周期(day)", default=7)
    orders = models.IntegerField(verbose_name="课程排序")
    attachment_path = models.CharField(max_length=128, verbose_name="课件路径",
blank=True, null=True)
    status_choices = (
      (0, '上线'),
      (1, '下线'),
      (2, '预上线')
    )
    status = models.SmallIntegerField(choices=status_choices,
default=0,verbose_name="课程状态")

 course_category=models.ForeignKey("CourseCategory",on_delete=models.CASCADE,null=True,
blank=True,verbose_name="课程分类")
    students= models.IntegerField(verbose_name="学习人数", default=0)
    lessons = models.IntegerField(verbose_name="课时数量", default=0)
    pub_lessons = models.IntegerField(verbose_name="课时更新数量", default=0)

    class Meta:
        db_table="ly_course"
        verbose_name = "专题课程"
        verbose_name_plural = "专题课程"

    def __str__(self):
        return "%s" % self.name


class Teacher(models.Model):
    """讲师、导师表"""
    role_choices = (
      (0, '讲师'),
      (1, '导师'),
    )
    name = models.CharField(max_length=32,verbose_name="讲师title")
    role = models.SmallIntegerField(choices=role_choices, default=0,berbose_name="讲师身
份")
    title = models.CharField(max_length=64, verbose_name="职位、职称")
    signature = models.CharField(max_length=255, help_text="导师签名", blank=True,
null=True)
    image = models.CharField(max_length=128, berbose_name="讲师封面")
    brief = models.TextField(max_length=1024,verbose_name="讲师描述")

    class Meta:
        db_table="ly_teacher"
        verbose_name = "讲师导师"
        verbose_name_plural = "讲师导师"

    def __str__(self):
        return "%s" % self.name



class CourseChapter(models.Model):
    """课程章节"""
    course = models.ForeignKey("Course",
related_name='coursechapters',on_delete=models.CASCADE,verbose_name="课程名称")
    chapter = models.SmallIntegerField(verbose_name="第几章", default=1)
    name = models.CharField(max_length=128,verbose_name="章节标题")
    summary = models.TextField(verbose_name="章节介绍", blank=True, null=True)
    pub_date = models.DateField(verbose_name="发布日期", auto_now_add=True)

    class Meta:
        db_table="ly_course_chapter"
        verbose_name = "课程章节"
        verbose_name_plural = "课程章节"

    def __str__(self):
        return "%s:(第%s章)%s" % (self.course, self.chapter, self.name)


class CourseLesson(models.Model):
    """课程课时"""
    section_type_choices = (
      (0, '文档'),
      (1, '练习'),
      (2, '视频')
    )
    chapter = models.ForeignKey("CourseChapter",
related_name='coursesections',on_delete=models.CASCADE,berbose_name="课程章节")
    name = models.CharField(max_length=128, verbose_name="课时标题")
    orders = models.PositiveSmallIntegerField(verbose_name="课时排序")
    section_type = models.SmallIntegerField(default=2,
choices=section_type_choices,vernbose_name="课时种类")
    section_link = models.CharField(max_length=255, blank=True,
null=True,verbose_name="课时链接" help_text="若是video，填vid,若是文档，填link")
    duration = models.CharField(verbose_name="视频时长", blank=True, null=True,
max_length=32)  # 仅在前端展示使用
    pub_date = models.DateTimeField(verbose_name="发布时间", auto_now_add=True)
    free_trail = models.BooleanField(verbose_name="是否可试看", default=False)

    class Meta:
        db_table="ly_course_lesson"
        verbose_name = "课程课时"
        verbose_name_plural = "课程课时"

    def __str__(self):
        return "%s-%s" % (self.chapter, self.name)
```