

# 传智播客

## 《Python 程序开发案例教程》

### 教学设计

课程名称：Python 程序开发案例教程

授课年级：2019 年级

授课学期：2019 学年第一学期

教师姓名：某某老师

2019 年 09 月 09 日

课题名称	第 8 章 模块	计划学时	3 学时
内容分析	面向对象是程序开发领域的重要思想，这种思想模拟了人类认识客观世界的逻辑，是当前计算机软件工程学的主流方法；类是面向对象的实现手段。Python 在设计之处就已经是一门面向对象语言，了解面向对象编程思想对于学习 Python 开发至关重要。		
教学目标及基本要求	<ol style="list-style-type: none"> <li>1. 了解模块的概念及其导入方式</li> <li>2. 掌握常见标准模块的使用</li> <li>3. 了解模块导入的特性</li> <li>4. 掌握自定义模块的使用</li> <li>5. 掌握包的结构及其导入方式</li> <li>6. 了解第三方模块的下载安装</li> </ol>		
教学重点	<ol style="list-style-type: none"> <li>1. 模块的导入方式</li> <li>2. 自定义模块</li> <li>3. 包的导入</li> </ol>		
教学难点	<ol style="list-style-type: none"> <li>1. 常见的标准模块</li> <li>2. <code>__all__</code> 属性</li> <li>3. <code>__name__</code> 属性</li> <li>4. 包的结构</li> </ol>		
教学方式	教学采用教师课堂讲授为主，使用教学 PPT 讲解		
教学过程	<p style="text-align: center;"><b>第一课时</b></p> <p style="text-align: center;"><b>（模块的概述、自定义模块、模块的导入特性）</b></p> <p><b>一、创设情境，引出面向对象</b></p> <p>（1）教师通过提出需求，引出什么是模块。</p> <p>（2）明确学习目标</p> <ul style="list-style-type: none"> <li>● 要求学生了解什么是模块</li> <li>● 要求学生掌握模块的导入方式</li> </ul>		

- 要求学生掌握常见的标准模块
- 要求学生掌握自定义模块
- 要求学生了解\_\_all\_\_属性
- 要求学生了解\_\_name\_\_属性

## 二、进行重点知识的讲解

(1) 教师根据课件，讲述模块的概述。

在 Python 程序中，每个 py 文件都可以视为一个模块，通过在当前 py 文件中导入其它 py 文件，可以使用被导入文件中定义的内容，Python 中的模块可分为三类，分别是内置模块、第三方模块和自定义模块。

(2) 教师根据课件，讲述模块的导入方式。

Python 模块的导入方式分为使用 import 导入和使用 from...import...导入两种。

(3) 教师根据课件，讲述常见的标准模块。

Python 内置了许多模块，例如 sys、os、random 和 time 模块。

(4) 教师根据课件，讲述自定义模块。

一般在进行程序开发时，不会将所有代码都放在一个文件中，而是将耦合度较低的多个功能写入不同的文件中，制作成模块，并在其他文件中以导入模块的方式使用自定义模块的内容。

(5) 教师根据课件，讲述\_\_all\_\_属性。

Python 模块的开发通常会定义一个\_\_all\_\_属性，该属性实际上是一个元组，该元组中包含的元素决定了在使用 from...import\*语句导入模块内容时通配符\*所包含的内容。如果\_\_all\_\_中只包含模块的部分内容，那么 from...import\*语句只会将\_\_all\_\_中包含的部分内容导入程序。

(6) 教师根据课件，讲述\_\_name\_\_属性。

在较大型的项目开发中，一个项目通常由多名开发人员共同开发，每名开发人员负责不同的模块。为了保证自己编写的程序在整合后可以正常运行，开发人员通常需在整合前额外编写测试代码，对自己负责的模块进行测试。Python 中设置了\_\_name\_\_属性，\_\_name\_\_属性通常与 if 条件语句一起使用，若当前模块是启动模块，则其\_\_name\_\_的值为“\_\_main\_\_”；若该模块被其它程序导入，则\_\_name\_\_的值为文件名。

### 三、归纳总结，布置作业/随堂练习

(1) 回顾上课前的学习目标，并对本节课的内容进行总结。

教师总结本节课需要了解什么是模块，掌握模块的导入方式、常见的标准模块、自定义模块，了解 `__all__` 属性与 `__name__` 属性。

(2) 布置随堂练习，检查学生掌握情况。

根据博学谷和随堂练习资源，给学生布置随堂练习，检测学生的掌握程度，并对学生出现的问题进行解决。

(3) 使用博学谷系统下发课后作业。

### 第二课时

#### (包的结构、包的导入、第三方模块的下载与安装、实例 1：随机生成验证码、实例 2：绘制多角星)

#### 一、回顾上节课的内容，继续讲解本课时的知识

(1) 教师对学生们的疑问进行统一答疑。

(2) 回顾总结上节课内容，继续介绍本课时的内容。

上节课我们学习了模块的概念、模块的导入方式、常见的标准模块、自定义模块、`__all__` 属性、`__name__` 属性，本节课将带领大家学习包结构、包的导入、第三方模块的下载与安装。

(3) 明确学习目标

- 要求学生了解包的结构
- 要求学生掌握包的导入方式
- 要求学生了解第三方模块的下载与安装
- 要求学生掌握访问限制

#### 二、进行重点知识的讲解

(1) 教师根据课件，讲解包的结构。

为了更好地组织 Python 代码，开发人员通常会根据不同业务将模块进行归类划分，并将功能相近的模块放到同一目录下。如果想要导入该目录下的模块，就需要先导入包。Python 中的包是一个包含 `__init__.py` 文件的目录，该目录下好包含一些模块以及子模块。

(2) 教师根据课件，讲解包的导入。

包的导入语模块的导入方法大致相同，亦可使用 `import` 或 `from...import...实`

现。

(3) 教师根据课件，讲解第三方模块的下载与安装。

程序开发中不仅需要大量的标准模块，而且还会根据业务需求使用第三方模块。在使用第三方模块之前，需要使用包管理工具 `pip` 下载和安装第三方模块。

(4) 教师根据课件，讲解实例 1：随机生成验证码。

教师根据教材资源实现实例 1：随机生成验证码，为学生讲解使用内置的 `random` 模块实现过程。

(5) 教师根据课件，讲解实例 2：绘制多角星。

教师根据教材资源实现实例 2：绘制多角星，为学生讲解使用第三方 `Turtle` 模块实现过程。

### 三、归纳总结，布置作业

(1) 回顾学习目标，总结本节课需要了解包的结构，掌握包的导入，了解包的导入、第三方模块的下载与安装。

(2) 布置随堂练习，检查学生掌握情况。

根据博学谷和随堂练习资源，给学生布置随堂练习，检测学生的掌握程度，并对学生出现的问题进行解决。

(3) 使用博学谷系统下发课后作业。

### 第三课时 (上机练习)

上机练习主要针对本章中需要重点掌握的知识点，以及在程序中容易出错的内容进行练习，通过上机练习可以考察同学对知识点的掌握情况，对代码的熟练程度。

**上机一：（练习教材示例代码以及完成实例 1 与实例 2 程序）**

**形式：独立完成**

**要求：**

(1) 要求学生能够熟练掌握教材中示例代码。

(2) 要求学生能够自己实现实例 1 与实例 2 程序。

思考题 和习题	见教材第 8 章配套的习题
教 学 后 记	