



# 信息工程系

## 教 案

---

课程名称： Python 编程基础

教 师： 陈旭文

总 学 时： 54

理论学时： 18

实训学时： 36

上课班级： 电子信息工程技术 251、电子信息工  
程技术（3+证书）251

授课学期： 2025-2026 学年第二学期

课题名称	第 1 章 开启 Python 学习之旅	计划课时	3 课时
教学引入	<p>在当今数字化和技术驱动的世界中，编程成为了一项越来越重要的技能。而在众多的编程语言中，Python 凭借其卓越的特性和广泛的应用领域赢得了很多业内人士的青睐。Python 语言以其简洁直观的语法、易于学习和使用的特点，成为众多初学者的首选。不仅如此，Python 在数据科学、人工智能、Web 开发等领域都有着广泛的应用，无论是企业还是个人都离不开这门多才多艺的编程语言。</p>		
教学目标	<ul style="list-style-type: none"> <li>● 了解 Python 的发展史，能够说出 Python 语言的发展过程</li> <li>● 了解 Python 的优缺点，能够说出 Python 语言的优点和缺点</li> <li>● 熟悉 Python 的应用领域，能够列举 Python 语言常用的应用领域</li> <li>● 掌握 Python 解释器的安装方法，能够独立在计算机中安装 Python 解释器</li> <li>● 掌握 IDLE 工具的使用方法，能够熟练使用 IDLE 工具编写并运行代码</li> <li>● 掌握 PyCharm 工具的安装及使用方法</li> </ul>		
教学重点	<ul style="list-style-type: none"> <li>● Python 解释器的安装</li> <li>● 开发第一个 Python 程序</li> <li>● 数据的表示--变量</li> <li>● 基本输入输出</li> </ul>		
教学难点	数据的表示--变量，基本输入输出		
思政要素	<ol style="list-style-type: none"> <li>1. 结合 Python 在人工智能、大数据等领域的应用，激发学生科技报国的家国情怀与创新意识</li> <li>2. 培养严谨规范、认真细致的编程习惯，树立工匠精神</li> <li>3. 依托开源文化，引导学生尊重知识产权、崇尚开放协作与共享精神</li> </ol>		
教学方式	PPT 讲授演示 + 实例编写调试		
教学过程	<p style="text-align: center;"><b>第一课时</b></p> <p style="text-align: center;"><b>（Python 的特点、Python 的应用领域、Python 解释器的安装）</b></p> <p>一、创设情景，导入新课</p> <p>教师通过给学生展示一些人工智能产品，比如智能客服机器人、指纹识别等，引导学生了解到开发这些产品的编程语言多数基于 Python，从而导入我们本节课要介绍的主题 Python。</p> <p>二、新课讲解</p> <p><b>知识点 1-Python 的特点</b></p> <p>教师通过 PPT 讲解 Python 的特点</p> <ul style="list-style-type: none"> <li>● 简单易学</li> <li>● 免费开源</li> <li>● 可移植性</li> <li>● 丰富的库</li> <li>● 良好的中文支持</li> </ul> <p><b>知识点 2-Python 的应用领域</b></p> <p>教师通过 PPT 讲解 Python 的应用领域。</p> <ul style="list-style-type: none"> <li>● Web 开发</li> <li>● 科学计算与数据分析</li> </ul>		

- 自动化运维
- 网络爬虫
- 游戏开发
- 人工智能

### 知识点 3-Python 解释器的安装

教师通过 PPT 讲解 Python 解释器的安装。

- (1) 访问 Python 官网的下载页面
- (2) 进入 Python 解释器安装包的下载页面
- (3) 下载相应版本的安装包，下载完成后，找到并双击安装包
- (4) 验证 Python 解释器是否安装成功

### 三、归纳总结

教师回顾本节课所讲的内容，并通过测试题的方式引导学生解答问题并给予指导。

## 第二课时

(PyCharm 的安装与使用、开发第一个 Python 程序、良好的编程约定)

### 一、复习巩固

教师通过上节课作业的完成情况，对学生吸收不好的知识点进行再次巩固讲解。

### 二、通过需求引入的方式导入新课

工欲善其事必先利其器，工匠想要使他的工作做好，一定要先让工具锋利。在开发 Python 程序过程中也是如此，便捷的开发工具不仅可以提高工作效率，还可以减少开发过程可能出现的错误，那么在开发 Python 程序过程中，经常会使用哪些开发工具呢？从而引导学生了解更便捷的开发工具，开发人员使用较多的一款开发工具 PyCharm，从而导入本节课要介绍的内容。

### 三、新课讲解

#### 知识点 1-PyCharm 的安装与使用

教师通过 PPT 讲解 PyCharm 的安装与使用

- (1) Professional 版本的特点
  - 提供 Python IDE 的所有功能，支持 Web 开发
  - 支持 Django、Flask、Google App 引擎、Pyramid 和 web2py
  - 支持 JavaScript、CoffeeScript、TypeScript、CSS 和 Cython 等
  - 支持远程开发、Python 分析器、数据库和 SQL 语句
- (2) Community 版本的特点
  - 轻量级的 Python IDE，只支持 Python 开发
  - 免费、开源、集成 Apache2 的许可证
  - 智能编辑器、调试器，支持重构和错误检查，集成 VCS 版本控制
- (3) PyCharm 的安装
- (4) PyCharm 的使用

#### 知识点 2-开发第一个 Python 程序

教师通过 PPT 结合实操的形式讲解开发第一个 Python 程序。

- (1) 通过 PPT 介绍任务案例需求。
- (2) 通过 PPT 介绍任务案例分析思路。
- (3) 在 PyCharm 中实现模拟手机充值的场景代码。

### 知识点 3-良好的编程约定

教师通过 PPT 讲解良好的编程约定。

#### (1) 代码布局

- 缩进
- 行的最大长度
- 空白行
- 多语句换行

#### (2) 空格要求

- 逗号、冒号、分号前不要加空格
- 函数的左括号前不要加空格
- 序列的左括号前不要加空格
- 运算符左右各加一个空格

#### (3) 代码注释

- 块注释
- 行内注释
- 文档字符串

#### (4) 命名规范

- 模块名、包名应简短，一般全为小写字母，多个单词用下划线分隔
- 函数名一般全为小写字母，多个单词用下划线分隔
- 类名一般使用大写字母开头的单个或多个单词
- 常量名一般全为大写字母，多个单词用下划线分隔

### 实例操作

#### (1) 模拟手机充值 (P12)。

```
>>> phone = input('请输入要充值的手机号码: ')
请输入要充值的手机号码: 13800000011
>>> money = input('请输入充值金额: ')
请输入充值金额: 50
>>> print('手机号码: '+phone+'成功充值'+money+'元')
手机号码: 13800000011成功充值50元
```

#### (2) 模拟用户登录 (P15)

```
>>> user_name = input('请输入账号: ')
请输入账号: admin
>>> password = input('请输入密码: ')
请输入密码: 123456
>>> print('登录成功! ')
登录成功!
```

### 四、归纳总结

教师回顾本节课所讲的内容，并通过测试题的方式引导学生解答问题并给予指导。

### 第三课时

(数据的表示—变量、基本输入输出、实例 1: 打印购物小票、实例 2: 打印功

## 能菜单)

### 一、复习巩固

教师通过上节课作业的完成情况,对学生吸收不好的知识点进行再次巩固讲解。

### 二、通过直接引入的方式导入新课

在上一节中我们学习了,PyCharm 的安装与使用、开发第一个 Python 程序、良好的编程约定,那么在本节中将对数据的表示--变量、基本输入输出、实例 1: 打印购物小票、实例 2: 打印功能菜单进行介绍。

### 三、新课讲解

#### 知识点 1-数据的表示—变量

教师通过 PPT 讲解数据的表示——变量。

- (1) 变量的定义格式
- (2) 变量名的定义规范
  - 变量名由字母、数字和下划线组成,且不以数字开头
  - 变量名区分大小写
  - 变量名不允许使用关键字
- (3) 关键字

#### 知识点 2-基本的输入输出

教师通过 PPT 讲解基本的输入输出。

- (1) input()函数
- (2) print()函数

#### 知识点 3-实例 1: 打印购物小票

教师通过 PPT 结合实操的形式讲解实例 1: 打印购物小票。

- (1) 通过 PPT 介绍任务案例需求。
- (2) 通过 PPT 介绍任务案例分析思路。
- (3) 在 PyCharm 中实现打印购物小票代码。

#### 知识点 4-实例 2: 打印功能菜单

教师通过 PPT 解释实操的形式讲解实例 2: 打印功能菜单

- (1) 通过 PPT 介绍任务案例需求。
- (2) 通过 PPT 介绍任务案例分析思路。
- (3) 在 PyCharm 中实现打印功能菜单代码。

### 四、归纳总结

教师回顾本节课所讲的内容,并通过测试题的方式引导学生解答问题并给予指导。

## 实训操作

上机一: (考察知识点为 Python 解释器的安装、PyCharm 的安装与使用、基本的输入输出)

#### 题目:

- (1) 在计算机中安装 Python 解释器,并验证是否安装成功
- (2) 在计算中安装 PyCharm,并在 PyCharm 中编写 Python 代码
- (3) 练习 input()函数和 print()函数的使用

上机二: (考察知识点为实例 1: 打印购物小票)

	<p><b>题目：</b> 根据任务描述的要求，编写代码实现实例 1：打印购物小票。</p> <p><b>上机三：（考察知识点为实例 2：打印功能菜单）</b></p> <p><b>题目：</b> 根据任务描述的要求，编写代码实现实例 2：打印功能菜单。</p>
教学后记	

课题名称	第 2 章 数据类型与字符串	计划课时	6 课时
教学引入	<p>在计算机编程中，数据是程序处理的核心。Python 中提供了丰富多样的数据类型，简单的数据类型有数字类型和字符串，其中数字类型用于表示数值形式的数据，它可以联合运算符进行算术或逻辑等操作；字符串用于表示文本形式的数据，它能够对文本进行一些诸如分割、拼接、查找和替换操作，在程序中应用十分广泛。本章将对数字类型、字符串和运算符的相关内容进行讲解，并通过实例帮助大家熟练运用它们。</p>		
教学目标	<ul style="list-style-type: none"> <li>● 掌握数字类型，能够在程序中正确表示不同数字类型的数据</li> <li>● 掌握运算符的用法，能够使用运算符进行数值运算</li> <li>● 掌握运算符优先级，能够在数值运算中正确使用运算符</li> <li>● 掌握字符串的创建方式，能够准确创建字符串类型的变量</li> <li>● 掌握格式化字符串的方式，能够使用%、format()和 f-string 这 3 种方式格式化字符串</li> <li>● 掌握字符串的常见操作，能够使用方法实现字符串的常见操作</li> <li>● 掌握字符串的索引和切片，能够使用索引和切片访问字符串的字符或子串</li> <li>● 掌握类型转换函数的使用，能够使用类型转换函数对不同类型的数据进行转换</li> </ul>		
教学重点	<ul style="list-style-type: none"> <li>● 算术运算符、比较运算符、赋值运算符、逻辑运算符、成员运算符、运算符优先级</li> <li>● 字符串的创建、格式化、常见操作、索引与切片</li> <li>● 类型转换函数</li> </ul>		
教学难点	位运算符；字符串的索引与切片		
思政要素	<ol style="list-style-type: none"> <li>1. 结合 BMI 指数计算、超市收银等生活实操实例，培养学以致用编程应用意识与实践能力</li> <li>2. 依托语法规范学习，锤炼严谨细致、逻辑缜密的思维与编程素养</li> <li>3. 借不良词语替换案例，引导树立文明用语意识和正确网络价值观</li> <li>4. 融合数学与编程知识，培养跨学科思维和自主探索的创新精神</li> <li>5. 通过上机调试实操，锻炼知错纠错能力和持之以恒的钻研精神</li> </ol>		
教学方式	PPT 讲授演示 + 实例编写调试		
教学过程	<p style="text-align: center;"><b>第一课时</b> <b>（数字类型、算术运算符、比较运算符）</b></p> <p>一、复习巩固 教师通过上节课作业的完成情况，对学生吸收不好的知识点进行再次巩固讲解。</p> <p>二、通过直接引入的方式导入新课 Python 中提供了丰富多样的数据类型，例如，数字类型和字符串，其中数字类型用于表示数值形式的数据，它可以联合运算符进行算术或逻辑等操作。本节将对数字类型、算术运算符、比较运算符进行讲解。</p> <p><b>知识点 1-数字类型</b> 教师通过 PPT 讲解数字类型</p>		

- 整型
- 浮点型
- 复数类型
- 布尔类型

### 知识点 2-算术运算符

教师通过 PPT 讲解算术运算符。

#### (1) 算术运算符

- 加: +
- 减: -
- 乘: \*
- 除: /
- 整除: //
- 取余: %
- 幂: \*\*

#### (2) Python 中不同类型的对象进行运算的转换规律

- ①布尔类型在进行算术运算时，将 False 和 True 被视为数值 0 或 1
- ②整型与浮点型进行运算时，将整型转化为浮点型
- ③其它类型与复数类型进行运算时，将其他类型转换为复数类型

### 知识点 3-比较运算符

教师通过 PPT 讲解比较运算符。

- ==
- !=
- >
- <
- >=
- <=

### 三、归纳总结

教师回顾本节课所讲的内容，并通过测试题的方式引导学生解答问题并给予指导。

## 第二课时

### (赋值运算符、逻辑运算符、成员运算符、位运算符)

#### 一、复习巩固

教师通过上节课作业的完成情况，对学生吸收不好的知识点进行再次巩固讲解。

#### 二、通过直接引入的方式导入新课

在上一节中，我们介绍了数字类型、算术运算符、比较运算符，在本节中将对赋值运算符、逻辑运算符、位运算符进行介绍。

#### 三、新课讲解

### 知识点 1-赋值运算符

教师通过 PPT 讲解赋值运算符。

- 等: =
- 加等: +=
- 减等: -=

- 乘等：\*=
- 除等：/=
- 整除等：//=
- 取余等：%=
- 幂等：\*\*=

#### 知识点 2-逻辑运算符

教师通过 PPT 讲解逻辑运算符。

- 逻辑或运算：or
- 逻辑与运算：and
- 逻辑非运算：not

#### 知识点 3-成员运算符

教师通过 PPT 讲解成员运算符。

(1) 代码布局

- in: 如果给定值在字符串、列表、元组、集合、字典中, 返回 True, 否则返回 False。
- not in: 如果给定值不在字符串、列表、元组、集合、字典中, 返回 True, 否则返回 False。

#### 知识点 4-位运算符

教师通过 PPT 讲解位运算符。

- 按位左移：<<
- 按右左移：>>
- 按位与：&
- 按位或：|
- 按位异或：^
- 按位取反：~

#### 四、归纳总结

教师回顾本节课所讲的内容, 并通过测试题的方式引导学生解答问题并给予指导。

### 第三课时

(运算符优先级、实例 1: 计算 BMI 指数、实例 2: 计算三角形的面积)

#### 一、复习巩固

教师通过上节课作业的完成情况, 对学生吸收不好的知识点进行再次巩固讲解。

#### 二、通过直接引入的方式导入新课

在上一节中介绍了赋值运算符、逻辑运算符、成员运算符、位运算符, 在本节中将对运算符优先级、实例 1: 计算 BMI 指数、实例 2: 计算三角形的面积进行介绍。

#### 三、新课讲解

##### 知识点 1-运算符优先级

教师通过 PPT 讲解数据的表示——变量。

(1) 运算符优先级由高到低顺序

- +=, -=, \*=, /=, //=, %=, \*\*=
- or

- and
- not
- in, not in
- <, <=, >, >=, !=, ==
- |
- ^
- &
- <<, >>
- +, -
- \*, /, %
- ~
- \*\*

(2) 使用圆括号改变表达式的执行顺序

#### 知识点 2-实例 1: 计算 BMI 指数

教师通过 PPT 结合实操的形式讲解实例 1: 计算 BMI 指数。

- (1) 通过 PPT 介绍任务案例需求。
- (2) 通过 PPT 介绍任务案例分析思路。
- (3) 在 PyCharm 中实现实例 1: 计算 BMI 指数代码。

#### 知识点 3-实例 2: 计算三角形的面积

教师通过 PPT 结合实操的形式讲解实例 2: 计算三角形的面积。

- (1) 通过 PPT 介绍任务案例需求。
- (2) 通过 PPT 介绍任务案例分析思路。
- (3) 在 PyCharm 中实现实例 2: 计算三角形的面积代码。

#### 四、归纳总结

教师回顾本节课所讲的内容, 并通过测试题的方式引导学生解答问题并给予指导。

### 第四课时

#### (实例 3: 判断水仙花数、字符串的创建、字符串的格式化)

##### 一、复习巩固

教师通过上节课作业的完成情况, 对学生吸收不好的知识点进行再次巩固讲解。

##### 二、通过直接引入的方式导入新课

在上一节中介绍了运算符优先级、实例 1: 计算 BMI 指数、实例 2: 计算三角形的面积, 在本节中将对实例 3: 判断水仙花数、字符串的创建、字符串的格式化进行讲解。

##### 三、新课讲解

#### 知识点 1-实例 3: 判断水仙花数

教师通过 PPT 结合实操的形式讲解实例 3: 判断水仙花数。

- (1) 通过 PPT 介绍任务案例需求。
- (2) 通过 PPT 介绍任务案例分析思路。
- (3) 在 PyCharm 中实现实例 3: 判断水仙花数代码。

#### 知识点 2-字符串的创建

教师通过 PPT 讲解字符串的创建。

- (1) 使用单引号创建字符串
- (2) 使用双引号创建字符串
- (3) 使用三引号创建字符串

### 知识点 3-字符串的格式化

教师通过 PPT 讲解字符串的格式化。

- (1) 利用符号%对字符串进行格式化。

- %s
- %d
- %o
- %x
- %X
- %e
- %f

- (2) 利用 format()方法对字符串进行格式化。


- 保留 n 位小数
- 数字补齐
- 显示百分比


- (3) 利用 f-strings 对字符串进行格式化。


#### 四、归纳总结

教师回顾本节课所讲的内容，并通过测试题的方式引导学生解答问题并给予指导。

五、作业：完成三个实例操作

 第2章 实例1 计算BMI指数.pdf

 第2章 实例2 计算三角形的面积.pdf

 第2章 实例3 判断水仙花数.pdf

### 第五课时

(字符串的常见操作、字符串的索引与切片、实例 4：制作名片)

#### 一、复习巩固

教师通过上节课作业的完成情况，对学生吸收不好的知识点进行再次巩固讲解。

#### 二、通过直接引入的方式导入新课

在上一节中介绍了实例 3：判断水仙花数、字符串的创建、字符串的格式化，在本节中将对字符串的常见操作、字符串的索引与切片、实例 4：制作名片进行讲解。

#### 三、新课讲解

### 知识点 1-字符串的常见操作

教师通过 PPT 结合实操的形式讲解字符串的常见操作。

- (1) 字符串拼接：+
- (2) 字符串替换：replace()
- (3) 字符串分割：split()
- (4) 取出字符串两侧空格：strip()

### 知识点 2-字符串的索引与切片

教师通过 PPT 讲解字符串的索引与切片。

(1) 索引

- ①什么是索引
- ②正向索引
- ③反向索引

(2) 切片

- ①什么是切片
- ②切片格式
- ③切片的使用

### 知识点 3-实例 4: 制作名片

教师通过通过 PPT 结合实操的形式讲解实例 4: 制作名片。

- (1) 通过 PPT 介绍任务案例需求。
- (2) 通过 PPT 介绍任务案例分析思路。
- (3) 实例 4: 制作名片的代码。

#### 四、归纳总结

教师回顾本节课所讲的内容,并通过测试题的方式引导学生解答问题并给予指导。

## 第六课时

(实例 5: 替换不良词语、类型转换函数、实例 6: 模拟超市收银抹零行为)

#### 一、复习巩固

教师通过上节课作业的完成情况,对学生吸收不好的知识点进行再次巩固讲解。

#### 二、通过直接引入的方式导入新课

在上一节中介绍了字符串的常见操作、字符串的索引与切片、实例 4: 制作名片,在本节中将对实例 5: 替换不良词语、类型转换函数、实例 6: 模拟超市收银抹零行为进行讲解。

#### 三、新课讲解

##### 知识点 1-实例 5: 替换不良词语

教师通过 PPT 结合实操的形式讲解实例 5: 替换不良词语。

- (1) 通过 PPT 介绍任务案例需求。
- (2) 通过 PPT 介绍任务案例分析思路。
- (3) 在 PyCharm 中实现实例 5: 替换不良词语代码。

##### 知识点 2-类型转换函数

教师通过 PPT 讲解类型转换函数。

- int()函数
- float()函数
- complex()函数
- str()函数

##### 知识点 3-实例 6: 模拟超市收银抹零行为

教师通过通过 PPT 结合实操的形式讲解实例 6: 模拟超市收银抹零行为。

- (1) 通过 PPT 介绍任务案例需求。
- (2) 通过 PPT 介绍任务案例分析思路。

	<p>(3) 实例 6: 模拟超市收银抹零行为的代码。</p> <p style="text-align: center;"><b>案例操作</b></p> <p><b>实例 1: 计算 BMI 指数、实例 2: 计算三角形的面积、实例 3: 判断水仙花数</b></p> <p><b>实例 4: 制作名片、实例 5: 替换不良词语、实例 6: 模拟超市收银抹零行</b></p>
教学后记	

课题名称	第3章 流程控制	计划课时	9 课时
教学引入	程序中的语句默认自上而下顺序执行。流程控制意指在程序执行时，通过一些特定的指令更改程序中语句的执行顺序，使程序产生跳跃、回溯等现象。本章将对 Python 中的条件语句、循环语句和跳转语句进行讲解。		
教学目标	<ul style="list-style-type: none"> <li>● 掌握条件语句的使用，能够根据需求选择合适的语句处理判断情况</li> <li>● 掌握条件嵌套的使用，能够使用条件嵌套处理多重判断情况</li> <li>● 掌握循环语句的使用，能够使用 for 语句或 while 语句实现循环操作</li> <li>● 掌握循环嵌套的使用，能够使用循环嵌套处理多层循环的操作</li> <li>● 掌握跳转语句，能够使用 break 和 continue 语句控制循环的执行情况</li> </ul>		
教学重点	<ul style="list-style-type: none"> <li>● if 语句、if-else 语句、if-elif-else 语句</li> <li>● for 语句、while 语句、break 语句、continue 语句</li> </ul>		
教学难点	条件嵌套的格式；while 循环嵌套；for 循环嵌套		
思政要素	<ol style="list-style-type: none"> <li>1. 培养条理清晰的逻辑思维与系统思考能力</li> <li>2. 树立用编程解决实际问题的学以致用理念</li> <li>3. 强化网络安全意识与信息保护责任观念</li> <li>4. 培养遵守规则、规范执行的编程素养</li> </ol>		
教学方式	PPT 讲授演示 + 实例编写调试		
教学过程	<p style="text-align: center;"><b>第一课时</b></p> <p style="text-align: center;"><b>(if 语句、if-else 语句、if-elif-else 语句、实例 1: 判断回文数)</b></p> <p>一、复习巩固 教师通过上节课作业的完成情况，对学生吸收不好的知识点进行再次巩固讲解。</p> <p>二、通过需求引入的方式导入新课 程序开发中经常会用到条件判断，比如，用户登录的时候，需判断用户输入的用户名和密码是否都正确，进而决定用户是否能够成功登录。类似这种需求的功能，都可以使用条件语句实现。Python 中的条件语句可使程序产生分支，根据分支数量的不同，可以分为单分支 if 语句、双分支 if-else 语句和多分支 if-elif-else 语句。接下来，本节将针对 if 语句、if-else 语句、if-elif-else 语句、实例 1: 判断回文数的内容进行详细讲解。</p> <p><b>知识点 1-if 语句</b> 教师通过 PPT 讲解 if 语句</p> <ol style="list-style-type: none"> <li>(1) if 语句格式</li> <li>(2) if 语句执行流程</li> <li>(3) if 语句的使用</li> </ol> <p><b>知识点 2-if-else 语句</b> 教师通过 PPT 讲解 if-else 语句。</p> <ol style="list-style-type: none"> <li>(1) if-else 语句格式</li> <li>(2) if-else 语句的执行流程</li> <li>(3) if-else 语句的使用</li> </ol>		

### 知识点 3-if-elif-else 语句

教师通过 PPT 讲解 if-elif-else 语句。

- (1) if-elif-else 语句格式
- (2) if-elif-else 语句的执行流程
- (3) if-elif-else 语句的使用

### 知识点 4-实例 1: 判断回文数

教师通过 PPT 结合实操的形式讲解实例 1: 判断回文数。

- (1) 通过 PPT 介绍任务案例需求。
- (2) 通过 PPT 介绍任务案例分析思路。
- (3) 在 PyCharm 中实现实例 1: 判断回文数代码。

## 三、归纳总结

教师回顾本节课所讲的内容,并通过测试题的方式引导学生解答问题并给予指导。

## 第二、三课时

(实例 2: 奖金发放、实例 3: 判断身体胖瘦程度、条件嵌套的格式)

### 一、复习巩固

教师通过上节课作业的完成情况,对学生吸收不好的知识点进行再次巩固讲解。

### 二、通过直接引入的方式导入新课

在上一节中,我们介绍了 if 语句、if-else 语句、if-elif-else 语句、实例 1: 判断回文数,在本节中将对实例 2: 奖金发放、实例 3: 判断身体胖瘦程度、条件嵌套的格式进行介绍。

### 三、新课讲解

#### 知识点 1-实例 2: 奖金发放

教师通过 PPT 结合实操的形式讲解实例 2: 奖金发放。

- (1) 通过 PPT 介绍任务案例需求。
- (2) 通过 PPT 介绍任务案例分析思路。
- (3) 在 PyCharm 中实现实例 2: 奖金发放代码。

#### 知识点 2-实例 3: 判断身体胖瘦程度

教师通过 PPT 讲解实例 3: 判断身体胖瘦程度。

- (1) 通过 PPT 介绍任务案例需求。
- (2) 通过 PPT 介绍任务案例分析思路。
- (3) 在 PyCharm 中实现实例 3: 判断身体胖瘦程度代码。

#### 知识点 3-条件嵌套的格式

教师通过 PPT 讲解条件嵌套的格式。

- (1) 讲解条件嵌套格式
- (2) 讲解条件嵌套的说明
- (3) 演示条件嵌套的使用

## 四、归纳总结

教师回顾本节课所讲的内容,并通过测试题的方式引导学生解答问题并给予指导。

## 案例操作

实例 1 判断回文数、实例 2：奖金发放、实例 3：判断身体胖瘦程度

#### 第四课时

(实例 4：模拟乘客进站流程、实例 5：物流计费系统、for 语句、实例 6：数据加密)

##### 一、复习巩固

教师通过上节课作业的完成情况，对学生吸收不好的知识点进行再次巩固讲解。

##### 二、通过直接引入的方式导入新课

在上一节中介绍了实例 2：奖金发放、实例 3：判断身体胖瘦程度、条件嵌套的格式，在本节中将对实例 4：模拟乘客进站流程、实例 5：物流计费系统、for 语句、实例 6：数据加密进行介绍。

##### 三、新课讲解

###### 知识点 1-实例 4：模拟乘客进站流程

教师通过 PPT 结合实操的形式讲解实例 4：模拟乘客进站流程。

- (1) 通过 PPT 介绍任务案例需求。
- (2) 通过 PPT 介绍任务案例分析思路。
- (3) 在 PyCharm 中实现实例 4：模拟乘客进站流程代码。

###### 知识点 2-实例 5：物流计费系统

教师通过 PPT 结合实操的形式讲解实例 5：物流计费系统。

- (1) 通过 PPT 介绍任务案例需求。
- (2) 通过 PPT 介绍任务案例分析思路。
- (3) 在 PyCharm 中实现实例 5：物流计费系统代码。

###### 知识点 3-for 语句

教师通过 PPT 讲解 for 语句。

- (1) 讲解 for 语句的格式。
- (2) 演示 for 语句的使用。

###### 知识点 4-实例 6：数据加密

教师通过 PPT 结合实操的形式讲解实例 6：数据加密。

- (1) 通过 PPT 介绍任务案例需求。
- (2) 通过 PPT 介绍任务案例分析思路。
- (3) 在 PyCharm 中实现实例 6：数据加密的代码。

##### 四、归纳总结

教师回顾本节课所讲的内容，并通过测试题的方式引导学生解答问题并给予指导。

#### 案例操作

实例 4：模拟乘客进站流程、实例 5：物流计费系统

#### 第五、六课时

(实例 7：逢七拍手游戏、while 语句、实例 8：登录系统账号检测)

##### 一、复习巩固

教师通过上节课作业的完成情况，对学生吸收不好的知识点进行再次巩固讲解。

##### 二、通过直接引入的方式导入新课

在上一节中介绍了实例 4：模拟乘客进站流程、实例 5：物流计费系统、for 语句、实例 6：数据加密，在本节中将对实例 7：逢七拍手游戏、while 语句、实例 8：登录系统账号检测进行讲解。

### 三、新课讲解

#### 知识点 1-实例 7：逢七拍手游戏

教师通过 PPT 结合实操的形式讲解实例 7：逢七拍手游戏。

- (1) 通过 PPT 介绍任务案例需求。
- (2) 通过 PPT 介绍任务案例分析思路。
- (3) 在 PyCharm 中实现实例 3：逢七拍手游戏代码。

#### 知识点 2-while 语句

教师通过 PPT 讲解 while 语句。

- (1) 讲解 while 语句的格式
- (2) 讲解 while 语句的执行流程
- (3) 演示 while 语句的使用

#### 知识点 3-实例 8：登录系统账号检测

教师通过 PPT 结合实操的形式讲解实例 8：登录系统账号检测。

- (1) 通过 PPT 介绍任务案例需求。
- (2) 通过 PPT 介绍任务案例分析思路。
- (3) 在 PyCharm 中实现实例 8：登录系统账号检测代码。

### 四、归纳总结

教师回顾本节课所讲的内容，并通过测试题的方式引导学生解答问题并给予指导。

### 案例操作

逢七拍手游戏

### 第七、八课时

(while 循环嵌套、for 循环嵌套、实例 9：九九乘法表、break 语句)

#### 一、复习巩固

教师通过上节课作业的完成情况，对学生吸收不好的知识点进行再次巩固讲解。

#### 二、通过直接引入的方式导入新课

在上一节中介绍了实例 7：逢七拍手游戏、while 语句、实例 8：登录系统账号检测，在本节中将对 while 循环嵌套、for 循环嵌套、实例 9：九九乘法表、break 语句进行讲解。

### 三、新课讲解

#### 知识点 1-while 循环嵌套

教师通过 PPT 讲解 while 循环嵌套。

- (1) 讲解 while 循环嵌套的格式
- (2) 讲解 while 循环嵌套的执行流程
- (3) 演示 while 循环嵌套的使用

#### 知识点 2-for 循环嵌套

教师通过 PPT 讲解 for 循环嵌套。

- (1) 讲解 for 循环嵌套的格式
- (2) 讲解 for 循环嵌套的执行流程

(3) 演示 for 循环嵌套的使用

### 知识点 3-实例 9：九九乘法表

教师通过通过 PPT 结合实操的形式讲解实例 9：九九乘法表。

- (1) 通过 PPT 介绍任务案例需求。
- (2) 通过 PPT 介绍任务案例分析思路。
- (3) 在 PyCharm 中实现实例 9：九九乘法表的代码。

### 知识点 4-break 语句

教师通过通过 PPT 讲解 break 语句。

- (1) 通过 PPT 介绍 break 语句的作用。
- (2) 通过 PPT 介绍 break 语句的格式。
- (3) 演示 break 语句的使用。

## 四、归纳总结

教师回顾本节课所讲的内容，并通过测试题的方式引导学生解答问题并给予指导。

### 案例操作

#### 九九乘法表

### 第九课时

(continue 语句、实例 10：猜数游戏)

#### 一、复习巩固

教师通过上节课作业的完成情况，对学生吸收不好的知识点进行再次巩固讲解。

#### 二、通过直接引入的方式导入新课

在上一节中介绍了 while 循环嵌套、for 循环嵌套、实例 9：九九乘法表、break 语句，在本节中将对 continue 语句、实例 10：猜数游戏进行讲解。

#### 三、新课讲解

### 知识点 1-continue 语句

教师通过 PPT 讲解 continue 语句。

- (1) 通过 PPT 介绍 continue 语句的作用。
- (2) 通过 PPT 介绍 continue 语句的格式。
- (3) 演示 continue 语句的使用。

### 知识点 2-实例 10：猜数游戏

教师通过 PPT 结合实操的形式讲解实例 10：猜数游戏。

- (1) 通过 PPT 介绍任务案例需求。
- (2) 通过 PPT 介绍任务案例分析思路。
- (3) 在 PyCharm 中实现实例 10：猜数游戏代码。

### 案例汇总

实例 1：判断回文数、实例 2：奖金发放、实例 3：判断身体胖瘦程度、实例 4：模拟乘客进站流程、实例 5：物流计费系统

实例 6：数据加密、实例 7：逢七拍手游戏、实例 8：登录系统账号检测、

	实例 9：九九乘法表、实例 10：猜数游戏
教学后记	

课题名称	第 4 章 列表与元组	计划课时	6 课时
教学引入	列表与元组是 Python 提供的两种非常重要的数据类型，它们属于序列类型，可以存储任意数量、任何类型的数据，并且支持访问、遍历等一系列操作，本章将对列表和元组这两种数据类型进行介绍。		
教学目标	<ul style="list-style-type: none"> <li>● 掌握列表的创建方式，能够使用[]和 list()函数创建列表</li> <li>● 掌握访问列表元素的方式，能够通过索引和切片访问列表元素</li> <li>● 掌握列表的遍历，能够通过 for 语句遍历列表的元素</li> <li>● 掌握列表的排序，能够通过 sort()和 reverse()方法对列表的元素实现排序操作</li> <li>● 掌握列表的增删改操作，能够选择合适的方式添加、删除、修改列表元素</li> <li>● 熟悉嵌套列表的创建与访问方式，能够创建嵌套列表并访问列表元素</li> <li>● 掌握元组的创建方式，能够使用()和 tuple()函数创建元组</li> <li>● 掌握访问元组元素的方式，能够通过索引和切片访问元组元素</li> </ul>		
教学重点	<ul style="list-style-type: none"> <li>● 列表的创建方式、访问列表元素、列表的遍历、列表的排序</li> <li>● 添加列表元素、删除列表元素、修改列表元素</li> <li>● 元组的创建方式、访问元组元素</li> </ul>		
教学难点	嵌套列表的创建与访问		
思政要素	<ol style="list-style-type: none"> <li>1. 培养数据整理与分类的结构化思维</li> <li>2. 树立按需选择数据类型的严谨编程意识</li> <li>3. 锤炼细致操作、精准处理数据的实操能力</li> <li>4. 激发用序列类型解决生活问题的应用思维</li> </ol>		
教学方式	PPT 讲授演示 + 实例编写调试		
教学过程	<p style="text-align: center;"><b>第一课时</b></p> <p style="text-align: center;"><b>（列表的创建方式、访问列表元素、实例 1：刮刮乐、列表的遍历）</b></p> <p>一、复习巩固 教师通过上节课作业的完成情况，对学生吸收不好的知识点进行再次巩固讲解。</p> <p>二、通过需求引入的方式导入新课 在开发程序过程中，经常需要保存程序产生的临时数据或永久数据，那么在 Python 中如何保存数据呢？列表是 Python 中最灵活的序列类型，它用于存储和管理任意数量、任意类型的元素，比如数字、字符串、列表或其他类型更为复杂的元素，都能够轻松应对。列表是可变的序列类型，它允许开发人员对其中的元素进行添加、修改、删除、排序等一些操作。接下来，本节将针对列表的创建方式、访问列表元素、实例 1：刮刮乐、列表的遍历的内容进行详细讲解。</p> <p><b>知识点 1-列表的创建方式</b> 教师通过 PPT 讲解列表的创建方式。</p> <p>（1）使用中括号“[]”创建列表</p> <p>（2）使用 list()函数创建列表</p> <p><b>知识点 2-访问列表元素</b></p>		

教师通过 PPT 讲解访问列表元素。

- (1) 使用索引访问列表元素
- (2) 使用切片访问列表元素

### 知识点 3-实例 1: 刮刮乐

教师通过 PPT 结合实操的形式讲解实例 1: 刮刮乐。

- (1) 通过 PPT 介绍任务案例需求。
- (2) 通过 PPT 介绍任务案例分析思路。
- (3) 在 PyCharm 中实现实例 1: 刮刮乐代码。

### 知识点 4-列表的遍历

教师通过 PPT 讲解访问列表元素。

- (1) 列表是一个可迭代对象
- (2) 通过 for 语句遍历其中的元素

## 三、归纳总结

教师回顾本节课所讲的内容,并通过测试题的方式引导学生解答问题并给予指导。

## 四、布置作业

教师通过高校教辅平台 (<http://tch.ityx.com>) 布置本节课作业以及下节课的预习作业。

## 第二、三课时

### (列表的排序、实例 2: 商品价格区间排序、添加列表元素)

## 一、复习巩固

教师通过上节课作业的完成情况,对学生吸收不好的知识点进行再次巩固讲解。

## 二、通过直接引入的方式导入新课

在上一节中,我们介绍了列表的创建方式、访问列表元素、实例 1: 刮刮乐、列表的遍历,在本节中将对列表的排序、实例 2: 商品价格区间排序、添加列表元素进行介绍。

## 三、新课讲解

### 知识点 1-列表的排序

教师通过 PPT 讲解列表的排序。

- (1) 演示 sort()方法的使用
- (2) 演示 sorted()函数的使用
- (3) 演示 reverse()方法的使用

### 知识点 2-实例 2: 商品价格区间排序

教师通过 PPT 讲解实例 2: 商品价格区间排序。

- (1) 通过 PPT 介绍任务案例需求。
- (2) 通过 PPT 介绍任务案例分析思路。
- (3) 在 PyCharm 中实现实例 2: 商品价格区间排序代码。

### 知识点 3-添加列表元素

教师通过 PPT 讲解条件嵌套的格式。

- (1) 演示 append()方法的使用
- (2) 演示 extend()方法的使用
- (3) 演示 insert()方法的使用

#### 四、归纳总结

教师回顾本节课所讲的内容，并通过测试题的方式引导学生解答问题并给予指导。

#### 五、布置作业

教师通过高校教辅平台 (<http://tch.ityxb.com>) 布置本节课作业以及下节课的预习作业。

### 案例操作

**实例 1：刮刮乐、实例 2：商品价格区间排序、实例 3：好友管理系统**

### 第四、五课时

**(删除列表元素、修改列表元素、实例 3：好友管理系统、嵌套列表的创建与访问)**

#### 一、复习巩固

教师通过上节课作业的完成情况，对学生吸收不好的知识点进行再次巩固讲解。

#### 二、通过直接引入的方式导入新课

在上一节中介绍了列表的排序、实例 2：商品价格区间排序、添加列表元素，在本节中将对删除列表元素、修改列表元素、实例 3：好友管理系统、嵌套列表的创建与访问进行介绍。

#### 三、新课讲解

##### 知识点 1-删除列表元素

教师通过 PPT 讲解删除列表元素。

- (1) 演示 del 语句的使用
- (2) 演示 remove()方法的使用
- (3) 演示 pop()方法的使用

##### 知识点 2-修改列表元素

教师通过 PPT 讲解删除列表元素。

- (1) 演示通过索引获取指定位置的元素并对该元素重新赋值

##### 知识点 3-实例 3：好友管理系统

教师通过 PPT 结合实操的形式讲解实例 3：好友管理系统。

- (1) 通过 PPT 介绍任务案例需求。
- (2) 通过 PPT 介绍任务案例分析思路。
- (3) 在 PyCharm 中实现实例 3：好友管理系统代码。

##### 知识点 3-嵌套列表的创建与访问

教师通过 PPT 讲解嵌套列表的创建与访问。

- (1) 讲解如何创建嵌套列表。
- (2) 讲解如何访问嵌套列表元素。

#### 四、归纳总结

教师回顾本节课所讲的内容，并通过测试题的方式引导学生解答问题并给予指导。

#### 五、布置作业

教师通过高校教辅平台 (<http://tch.ityxb.com>) 布置本节课作业以及下节课的预习作业。

	<p style="text-align: center;"><b>第六课时</b></p> <p style="text-align: center;"><b>(实例 4: 随机分配办公室、元组的创建方式、访问元组元素、实例 5: 中文数字对照表)</b></p> <p>一、复习巩固 教师通过上节课作业的完成情况,对学生吸收不好的知识点进行再次巩固讲解。</p> <p>二、通过直接引入的方式导入新课 在上一节中介绍了删除列表元素、修改列表元素、实例 3: 好友管理系统、嵌套列表的创建与访问,在本节中将对实例 4: 随机分配办公室、元组的创建方式、访问元组元素、实例 5: 中文数字对照表进行讲解。</p> <p>三、新课讲解</p> <p><b>知识点 1-实例 4: 随机分配办公室</b> 教师通过 PPT 结合实操的形式讲解实例 4: 随机分配办公室。</p> <ol style="list-style-type: none"> <li>(1) 通过 PPT 介绍任务案例需求。</li> <li>(2) 通过 PPT 介绍任务案例分析思路。</li> <li>(3) 在 PyCharm 中实现实例 4: 随机分配办公室代码。</li> </ol> <p><b>知识点 2-元组的创建方式</b> 教师通过 PPT 讲解元组的创建方式。</p> <ol style="list-style-type: none"> <li>(1) 使用圆括号“()”创建元组</li> <li>(2) 使用 tuple()函数创建元组</li> </ol> <p><b>知识点 3-访问元组元素</b> 教师通过 PPT 讲解访问元组元素。</p> <ol style="list-style-type: none"> <li>(1) 使用索引访问单个元素</li> <li>(2) 使用切片访问元组元素</li> </ol> <p><b>知识点 5-实例 5: 中文数字对照表</b> 教师通过 PPT 结合实操的形式讲解实例 5: 中文数字对照表。</p> <ol style="list-style-type: none"> <li>(1) 通过 PPT 介绍任务案例需求。</li> <li>(2) 通过 PPT 介绍任务案例分析思路。</li> <li>(3) 在 PyCharm 中实现实例 5: 中文数字对照表代码。</li> </ol> <p>四、归纳总结 教师回顾本节课所讲的内容,并通过测试题的方式引导学生解答问题并给予指导。</p> <p>五、布置作业 教师通过高校教辅平台 (<a href="http://tch.ityxb.com">http://tch.ityxb.com</a>) 布置本节课作业以及下节课的预习作业。</p> <p style="text-align: center;"><b>案例操作</b></p> <p style="text-align: center;"><b>实例 3: 好友管理系统、实例 4: 随机分配办公室</b></p>
教学后记	

课题名称	第 5 章 字典和集合	计划课时	6 课时
教学引入	除了之前章节介绍的数据类型，Python 还提供了两种实用且强大的数据类型，分别是字典和集合。字典通过键值对（key-value）的形式存储数据，大大提高了数据查找的效率；而集合作为无序且不包含重复数据的类型，它为处理数据唯一性提供了简洁便捷的方法。因此，字典和集合在解决许多实际问题时发挥着重要作用。本章将带大家学习字典和集合这两种类型。		
教学目标	<ul style="list-style-type: none"> <li>● 掌握字典的创建方式，能够使用 {} 和 dict() 函数创建字典</li> <li>● 掌握访问字典的方式，能够通过键访问字典中其对应的值</li> <li>● 掌握字典元素的添加和修改方式，能够通过 update() 方法或字典的键添加或修改元素</li> <li>● 掌握字典元素的删除方式，能够根据需求选择合适的方法从字典中删除元素</li> <li>● 掌握字典元素的查询方式，能够通过 items()、keys()、values() 方法查询字典的所有元素、所有键和所有值</li> <li>● 掌握集合的创建方式，能够使用 {} 和 set() 函数创建集合</li> <li>● 掌握集合元素的基本操作，能够添加、删除和清空集合的元素</li> <li>● 熟悉集合操作符的用法，能够通过操作符对集合进行联合、取交集、差补和对称差分操作</li> </ul>		
教学重点	<ul style="list-style-type: none"> <li>● 字典的创建方式、通过键访问字典、字典元素的添加和修改、字典元素的删除、字典元素的查询</li> <li>● 集合的创建方式，集合元素的添加、删除和清空</li> </ul>		
教学难点	集合类型的操作符 实例 3：生词本		
思政要素	<ol style="list-style-type: none"> <li>1. 借助字典键值对、集合唯一性特性，培养高效分类、精准筛选数据的结构化思维。</li> <li>2. 结合字典增删改查、集合运算操作，锤炼严谨操作、逻辑处理数据的实操素养。</li> <li>3. 依托单词识别、手机通讯录等生活实例，激发用字典集合解决实际问题的应用能力。</li> <li>4. 借助生词本等实例开发，培养学生有序管理信息、积累知识的良好学习与编程习惯。</li> </ol>		
教学方式	PPT 讲授演示 + 实例编写调试		
教学过程	<p style="text-align: center;"><b>第一课时</b></p> <p style="text-align: center;"><b>（字典的创建方式、通过键访问字典、实例 1：单词识别、字典元素的添加和修改）</b></p> <p>一、复习巩固 教师通过上节课作业的完成情况，对学生吸收不好的知识点进行再次巩固讲解。</p> <p>二、通过直接引入的方式导入新课 除了之前章节介绍的数据类型，Python 还提供了两种实用且强大的数据类型，分别是字典和集合。字典通过键值对（key-value）的形式存储数据，大大提高了数据查找的效率，字典在解决许多实际问题时发挥着重要作用。本节将对</p>		

字典的创建方式、通过键访问字典、实例 1：单词识别、字典元素的添加和修改进行介绍。

#### 知识点 1-字典的创建方式

教师通过 PPT 讲解字典的方式。

- (1) 使用大括号“{}”创建字典
- (2) 使用 dict()函数创建字典

#### 知识点 2-通过键访问字典

教师通过 PPT 讲解通过键访问字典。

- (1) 演示通过键获取字典中对应的值

#### 知识点 3-实例 1：单词识别

教师通过 PPT 结合实操的形式讲解实例 1：单词识别。

- (1) 通过 PPT 介绍任务案例需求。
- (2) 通过 PPT 介绍任务案例分析思路。
- (3) 在 PyCharm 中实现实例 1：单词识别代码。

### 三、归纳总结

教师回顾本节课所讲的内容，并通过测试题的方式引导学生解答问题并给予指导。

### 四、布置作业——实例 1：单词识别

教师通过高校教辅平台 (<http://tch.itxyb.com>) 布置本节课作业以及下节课的预习作业。

## 第二、三课时

### (字典元素的删除、字典元素的查询、实例 2：手机通讯录)

#### 一、复习巩固

教师通过上节课作业的完成情况，对学生吸收不好的知识点进行再次巩固讲解。

#### 二、通过直接引入的方式导入新课

在上一节中，我们介绍了字典的创建方式、通过键访问字典、实例 1：单词识别、字典元素的添加和修改，在本节中将对字典元素的删除、字典元素的查询、实例 2：手机通讯录进行介绍。

#### 三、新课讲解

##### 知识点 1-字典元素的删除

教师通过 PPT 讲解字典元素的删除。

- (1) 演示 pop()方法的使用
- (2) 演示 popitem()方法的使用
- (3) 演示 clear()方法的使用

##### 知识点 2-实例 2：字典元素的查询

教师通过 PPT 讲解字典元素的查询。

- (1) 查询字典的所有元素。
- (2) 查询字典的所有键。
- (3) 查询字典的所有值。

##### 知识点 3-实例 2：手机通讯录

教师通过 PPT 结合实操的形式讲解实例 2：手机通讯录。

- (1) 通过 PPT 介绍任务案例需求。

(2) 通过 PPT 介绍任务案例分析思路。

(3) 在 PyCharm 中实现实例 2：手机通讯录代码。

#### 四、归纳总结

教师回顾本节课所讲的内容，并通过测试题的方式引导学生解答问题并给予指导。

#### 五、布置作业——实例 2：手机通讯录

教师通过高校教辅平台 (<http://tch.ityx.com>) 布置本节课作业以及下节课的预习作业。

### 第四课时

(集合的创建方式、集合元素的添加、删除和清空、集合类型的操作符)

#### 一、复习巩固

教师通过上节课作业的完成情况，对学生吸收不好的知识点进行再次巩固讲解。

#### 二、通过直接引入的方式导入新课

在上一节中介绍了字典元素的删除、字典元素的查询、实例 2：手机通讯录，在本节中将对集合的创建方式、集合元素的添加、删除和清空、集合类型的操作符进行介绍。

#### 三、新课讲解

##### 知识点 1-集合的创建方式

教师通过 PPT 讲解集合的创建方式。

(1) 使用大括号“{}”创建集合

(2) 使用 set()函数创建集合

##### 知识点 2-集合元素的添加、删除和清空

教师通过 PPT 讲解集合元素的添加、删除和清空。

(1) 添加元素

- add()方法
- update()方法

(2) 删除元素

- remove()方法
- discard()方法
- pop()方法
- clear()方法

##### 知识点 3-集合类型的操作符

教师通过 PPT 讲解集合类型的操作符。

- (1) 联合操作：|
- (2) 交集操作：&
- (3) 差补操作：-
- (4) 对称差分操作：^

#### 四、归纳总结

教师回顾本节课所讲的内容，并通过测试题的方式引导学生解答问题并给予指导。

### 第五、六课时 (实例 3：生词本)

	<p>一、复习巩固 教师通过上节课作业的完成情况，对学生吸收不好的知识点进行再次巩固讲解。</p> <p>二、通过直接引入的方式导入新课 在上一节中介绍了集合的创建方式、集合元素的添加、删除和清空、集合类型的操作符，在本节中将对生词本进行讲解。</p> <p>三、新课讲解</p> <p><b>知识点 1-实例 3：生词本</b> 教师通过 PPT 结合实操的形式讲解实例 3：生词本。</p> <p>(1) 通过 PPT 介绍任务案例需求。 (2) 通过 PPT 介绍任务案例分析思路。 (3) 在 PyCharm 中实现实例 3：生词本代码。</p> <p>四、归纳总结 教师回顾本节课所讲的内容，并通过测试题的方式引导学生解答问题并给予指导。</p> <p>五、布置作业——<b>实例 3：生词本</b> 教师通过高校教辅平台 (<a href="http://tch.ityx.com">http://tch.ityx.com</a>) 布置本节课作业以及下节课的预习作业。</p> <p style="text-align: center;"><b>操作案例</b></p> <p><b>实例 1：单词识别、实例 2：手机通讯录、实例 3：生词本</b></p>
教学后记	

课题名称	第 6 章 函数	计划课时	6 课时
教学引入	<p>在开发程序时有些功能的逻辑十分相似或完全相同，只要使用这个功能，就需要在相应的位置执行相似或者重复的代码块。如果一个代码块存在问题，那么所有的代码块都要同步修改。这不仅会让程序存在大量重复的代码块，而且增加了代码的维护成本。函数解决了这些问题，它会将相似或重复的代码封装成特定功能的代码模块，使整个程序的结构变得清晰。本章通过 5 个实例对函数相关的知识进行详细讲解。</p>		
教学目标	<ul style="list-style-type: none"> <li>● 掌握函数的定义和调用，能够在程序中定义和调用函数</li> <li>● 掌握参数的传递方式，能够通过多种方式向函数内部传递数据</li> <li>● 熟悉变量作用域，能够区分程序中的全局变量和局部变量</li> <li>● 掌握递归函数的使用，能够运用递归函数解决阶乘的问题</li> <li>● 掌握匿名函数的使用，能够运用匿名函数简化简单函数的定义</li> <li>● 熟悉常用的内置函数，能够使用 chr() 函数返回单个 Unicode 字符的码值</li> </ul>		
教学重点	<ul style="list-style-type: none"> <li>● 函数的定义、函数的调用、位置参数传递、关键字参数传递、默认参数传递</li> <li>● 参数打包传递、参数解包传递</li> <li>● 局部变量、全局变量</li> <li>● 匿名函数、递归函数</li> <li>● 常用的内置函数</li> </ul>		
教学难点	<ul style="list-style-type: none"> <li>● 参数打包传递、参数解包传递</li> <li>● 递归函数</li> </ul>		
思政要素	<ol style="list-style-type: none"> <li>1. 培养代码封装、模块化设计的编程思维</li> <li>2. 锤炼逻辑拆解、分步解决问题的思维能力</li> <li>3. 树立规范编程、高效复用代码的良好习惯</li> <li>4. 激发用函数思想解决实际业务问题的应用能力</li> </ol>		
教学方式	PPT 讲授演示 + 实例编写调试		
教学过程	<p style="text-align: center;"><b>第一课时</b></p> <p style="text-align: center;"><b>(函数的定义、函数的调用、实例 1: 计算器、位置参数传递、关键字参数传递)</b></p> <p>一、复习巩固</p> <p>教师通过上节课作业的完成情况，对学生吸收不好的知识点进行再次巩固讲解。</p> <p>二、通过直接引入的方式导入新课</p> <p>在 Python 中，函数是一种组织好的、可重复使用的代码段，用于实现特定的功能或操作。函数可以被视为有名字的一段代码，这段代码是提前定义好以实现特定的功能，当程序的某个地方需要使用这个功能时，就可以通过名字调用这个函数。本节将对函数的定义、函数的调用、实例 1: 计算器、位置参数传递、关键字参数传递进行介绍。</p> <p><b>知识点 1-函数的定义</b></p> <p>教师通过 PPT 讲解函数的定义。</p> <ol style="list-style-type: none"> <li>(1) 介绍什么是函数</li> <li>(2) 介绍函数的语法格式</li> <li>(3) 演示如何定义函数</li> </ol>		

### 知识点 2-函数的调用

教师通过 PPT 讲解函数的调用。

- (1) 介绍函数调用的格式
- (2) 演示函数的调用

### 知识点 3-实例 1: 计算器

教师通过 PPT 结合实操的形式讲解实例 1: 单词识别。

- (1) 通过 PPT 介绍任务案例需求。
- (2) 通过 PPT 介绍任务案例分析思路。
- (3) 在 PyCharm 中实现实例 1: 计算器代码。

### 知识点 4-位置参数传递

教师通过 PPT 讲解通过键访问字典。

- (1) 介绍什么是位置参数传递
- (2) 演示位置参数传递的使用

### 知识点 5-关键字参数传递

教师通过 PPT 讲解通过键访问字典。

- (1) 介绍什么是关键字参数
- (2) 演示关键字传递的使用

## 三、归纳总结

教师回顾本节课所讲的内容,并通过测试题的方式引导学生解答问题并给予指导。

## 四、布置作业——实例 1: 计算器

教师通过高校教辅平台 (<http://tch.ityx.com>) 布置本节课作业以及下节课的预习作业。

## 第二课时

### (默认参数传递、参数打包传递、参数解包传递、实例 2: 商品折扣计算)

## 一、复习巩固

教师通过上节课作业的完成情况,对学生吸收不好的知识点进行再次巩固讲解。

## 二、通过直接引入的方式导入新课

在上一节中,我们介绍了函数的定义、函数的调用、实例 1: 计算器、位置参数传递、关键字参数传递,在本节中将对默认参数传递、参数打包传递、参数解包传递、实例 2: 商品折扣计算进行介绍。

## 三、新课讲解

### 知识点 1-默认参数传递

教师通过 PPT 讲解默认参数传递。

- (1) 介绍什么是默认参数传递
- (2) 演示默认参数传递的使用

### 知识点 2-参数打包传递

教师通过 PPT 讲解参数打包传递。

- (1) 介绍什么是参数打包传递
- (2) 演示参数打包传递的使用

### 知识点 3-参数解包传递

教师通过 PPT 讲解参数解包传递。

(1) 介绍什么是参数解包传递

(2) 演示参数解包传递的使用

#### 知识点 4-实例 2: 商品折扣计算

教师通过 PPT 结合实操的形式讲解实例 2: 商品折扣计算。

(1) 通过 PPT 介绍任务案例需求。

(2) 通过 PPT 介绍任务案例分析思路。

(3) 在 PyCharm 中实现实例 2: 商品折扣计算代码。

#### 四、归纳总结

教师回顾本节课所讲的内容,并通过测试题的方式引导学生解答问题并给予指导。

#### 五、布置作业——实例 2: 商品折扣计算

教师通过高校教辅平台 (<http://tch.ityxb.com>) 布置本节课作业以及下节课的预习作业。

### 第三、四课时

(局部变量、全局变量、实例 3: 学生信息管理系统、匿名函数)

#### 一、复习巩固

教师通过上节课作业的完成情况,对学生吸收不好的知识点进行再次巩固讲解。

#### 二、通过直接引入的方式导入新课

在上一节中介绍了默认参数传递、参数打包传递、参数解包传递、实例 2: 商品折扣计算,在本节中将对局部变量、全局变量、实例 3: 学生信息管理系统、匿名函数进行介绍。

#### 三、新课讲解

##### 知识点 1-局部变量

教师通过 PPT 讲解局部变量。

(1) 介绍什么是局部变量

(2) 演示局部变量的使用

##### 知识点 2-全局变量

教师通过 PPT 讲解全局变量。

(1) 介绍什么是全局变量

(2) 演示全局变量的使用

##### 知识点 3-实例 3: 学生信息管理系统

教师通过 PPT 结合实操的形式讲解实例 3: 学生信息管理系统。

(1) 通过 PPT 介绍任务案例需求。

(2) 通过 PPT 介绍任务案例分析思路。

(3) 在 PyCharm 中实现实例 3: 学生信息系统管理代码。

##### 知识点 4-匿名函数

教师通过 PPT 讲解全局变量。

(1) 介绍匿名函数的语法格式

(2) 介绍匿名函数与普通函数的区别

(3) 介绍匿名函数的使用

#### 四、归纳总结

教师回顾本节课所讲的内容,并通过测试题的方式引导学生解答问题并给予

指导。

#### 五、布置作业——实例 3：学生信息管理系统

教师通过高校教辅平台 (<http://tch.ityx.com>) 布置本节课作业以及下节课的预习作业。

### 第五、六课时

#### (实例 4：商品销量排序、递归函数、实例 5：失之毫厘，谬以千里、常用的内置函数)

#### 一、复习巩固

教师通过上节课作业的完成情况，对学生吸收不好的知识点进行再次巩固讲解。

#### 二、通过直接引入的方式导入新课

在上一节中介绍了局部变量、全局变量、实例 3：学生信息管理系统、匿名函数，在本节中将对实例 4：商品销量排序、递归函数、实例 5：失之毫厘，谬以千里、常用的内置函数进行讲解。

#### 三、新课讲解

##### 知识点 1-实例 4：商品销量排序

教师通过 PPT 结合实操的形式讲解实例 4：商品销量排序。

- (1) 通过 PPT 介绍任务案例需求。
- (2) 通过 PPT 介绍任务案例分析思路。
- (3) 在 PyCharm 中实现实例 4：商品销量排序代码。

##### 知识点 2-递归函数

教师通过 PPT 讲解递归函数。

- (1) 介绍什么是递归函数
- (2) 介绍满足递归函数的基本条件
- (3) 演示递归函数的使用

##### 知识点 3-实例 5：失之毫厘，谬以千里

教师通过 PPT 结合实操的形式讲解实例 5：失之毫厘，谬以千里。

- (1) 通过 PPT 介绍任务案例需求。
- (2) 通过 PPT 介绍任务案例分析思路。
- (3) 在 PyCharm 中实现实例 5：失之毫厘，谬以千里代码。

##### 知识点 4-常用内置函数

教师通过 PPT 讲解常用内置函数。

- abs()函数
- ord()函数
- chr()函数

#### 四、归纳总结

教师回顾本节课所讲的内容，并通过测试题的方式引导学生解答问题并给予指导。

#### 五、布置作业——实例 5：失之毫厘，谬以千里

教师通过高校教辅平台 (<http://tch.ityx.com>) 布置本节课作业以及下节课的预习作业。

### 操作实训

	实例 1: 计算器、实例 2: 商品折扣计算、实例 3: 学生信息管理系统 实例 4: 商品销量排序、实例 5: 失之毫厘，谬以千里
教学后记	

课题名称	第 7 章 类与面向对象	计划课时	6 课时
教学引入	面向对象是计算机软件开发中的重要思想，它模拟了人类对现实世界的认知逻辑，被广泛应用于软件工程领域。类是面向对象编程的核心概念之一，它在 Python 中也是一种重要的编程手段。由于 Python 最初就是作为一门面向对象的语言设计的，因此掌握面向对象编程对于学习 Python 至关重要。本章将针对面向对象等知识进行详细介绍。		
教学目标	<ul style="list-style-type: none"> <li>● 了解面向对象的思想，能够说出面向对象与面向过程的区别</li> <li>● 熟悉面向对象的基本概念，能够归纳封装、继承和多态的概念</li> <li>● 熟悉类和对象的关系，能够归纳类和对象之间的关系</li> <li>● 掌握类的定义，能够通过关键字 class 定义类</li> <li>● 掌握对象的创建与使用，能够使用对象访问属性或调用方法</li> <li>● 掌握访问限制，能够通过私有成员来限制类外部对成员的访问</li> <li>● 掌握构造方法的使用，能够在构造方法中定义实例属性</li> <li>● 掌握类方法和静态方法的定义，能够在类中定义类方法和静态方法并使用</li> <li>● 掌握单继承、多继承的语法，能够在类中实现单继承和多继承</li> <li>● 掌握方法重写的方式，能够在子类中重写从父类继承的方法</li> <li>● 掌握 super()函数的使用，能够通过 super()函数调用父类被重写的方法</li> <li>● 熟悉多态的特性，能够在程序中以多态的形式调用类中定义的方法</li> </ul>		
教学重点	<ul style="list-style-type: none"> <li>● 类的定义</li> <li>● 对象的创建与使用</li> <li>● 访问限制、构造方法、类方法、静态方法</li> <li>● 单继承、多继承</li> <li>● 方法的重写</li> <li>● super()函数</li> </ul>		
教学难点	多继承 方法的重写 super () 函数		
思政要素	<ol style="list-style-type: none"> <li>1. 培养抽象建模、面向对象的系统编程思维</li> <li>2. 树立封装复用、继承拓展的工程化开发理念</li> <li>3. 锤炼逻辑设计、灵活应用多态特性的思维能力</li> <li>4. 激发结合现实场景进行程序设计的创新能力</li> </ol>		
教学方式	PPT 讲授演示 + 实例编写调试		
教学过程	<p style="text-align: center;"><b>第一课时</b></p> <p style="text-align: center;"><b>（面向对象概述、面向对象的基本概念、类和对象的关系、类的定义）</b></p> <p>一、复习巩固 教师通过上节课作业的完成情况，对学生吸收不好的知识点进行再次巩固讲解。</p> <p>二、通过直接引入的方式导入新课 面向对象（Object-Oriented）是一种软件开发的编程范式和方法论，通过将数据和对数据的操作组织为一个整体，从更高的层次进行软件建模，使得软件的设计更加贴近事物的自然运行模式。本节将对面向对象概述、面向对象的基本概念、类和对象的关系、类的定义进行介绍。</p>		

### 知识点 1-面向对象概述

教师通过 PPT 讲解面向对象概述。

- (1) 介绍什么是面向对象
- (2) 通过对比面向过程编程和面向对象编程引出面向对象优势

### 知识点 2-面向对象的基本概念

教师通过 PPT 讲解面向对象概念。

- (1) 介绍什么是对象
- (2) 介绍什么是类
- (3) 介绍什么是抽象
- (4) 介绍什么是封装
- (5) 介绍什么是继承
- (6) 介绍什么多态

### 知识点 3-类和对象的关系

教师通过 PPT 讲解类和对象的关系。

- (1) 通过汽车设计图介绍类和对象的关系

### 知识点 4-类的定义

教师通过 PPT 讲解类的定义。

- (1) 介绍如何定义类的语法格式
- (2) 通过示例代码演示如何定义类

## 三、归纳总结

教师回顾本节课所讲的内容，并通过测试题的方式引导学生解答问题并给予指导。

## 四、布置作业

教师通过高校教辅平台 (<http://tch.ityxb.com>) 布置本节课作业以及下节课的预习作业。

## 第二课时

### (对象的创建与使用、访问限制、构造方法、类方法、静态方法)

## 一、复习巩固

教师通过上节课作业的完成情况，对学生吸收不好的知识点进行再次巩固讲解。

## 二、通过直接引入的方式导入新课

在上一节中，我们介绍了面向对象概述、面向对象的基本概念、类和对象的关系、类的定义，在本节中将对象的创建与使用、访问限制、构造方法、类方法、静态方法进行介绍。

## 三、新课讲解

### 知识点 1-对象的创建与使用

教师通过 PPT 讲解对象的创建与使用。

- (1) 介绍创建对象的格式
- (2) 通过示例代码演示如何创建对象
- (3) 介绍访问对象成员的格式
- (4) 通过示例代码演示如何访问对象成员

### 知识点 2-访问限制

教师通过 PPT 讲解访问限制。

- (1) 介绍定义私有成员的格式
- (2) 通过示例代码演示如何定义私有成员
- (3) 通过示例代码演示如何使用私有成员

#### 知识点 3-构造方法

教师通过 PPT 讲解构造方法。

- (1) 介绍什么是构造方法
- (2) 通过示例代码演示如何定义无参的构造方法
- (3) 通过示例代码演示如何定义有参的构造方法

#### 知识点 4-类方法

教师通过 PPT 讲解类方法。

- (1) 介绍如何定义类方法
- (2) 通过示例代码演示如何定义类方法
- (3) 介绍如何调用类方法
- (4) 通过示例代码演示如何调用定义的类方法
- (5) 介绍如何修改和访问类属性
- (6) 通过示例代码演示如何修改和访问类属性

#### 知识点 5-静态方法

教师通过 PPT 讲解静态方法。

- (1) 介绍静态方法和实例方法的区别
- (2) 演示如何定义静态方法
- (3) 通过示例代码演示如何调用静态方法

#### 四、归纳总结

教师回顾本节课所讲的内容，并通过测试题的方式引导学生解答问题并给予指导。

#### 五、布置作业

教师通过高校教辅平台 (<http://tch.ityx.com>) 布置本节课作业以及下节课的预习作业。

### 第三、四课时

#### (实例 1: 航天器信息查询工具、单继承、多继承、方法的重写)

##### 一、复习巩固

教师通过上节课作业的完成情况，对学生吸收不好的知识点进行再次巩固讲解。

##### 二、通过直接引入的方式导入新课

在上一节中介绍了对象的创建与使用、访问限制、构造方法、类方法、静态方法，在本节中将对实例 1: 航天器信息查询工具、单继承、多继承、方法的重写进行介绍。

##### 三、新课讲解

#### 知识点 1-实例 1: 航天器信息查询工具

教师通过 PPT 结合实操的形式讲解实例 1: 航天器信息查询工具。

- (1) 通过 PPT 介绍任务案例需求。
- (2) 通过 PPT 介绍任务案例分析思路。
- (3) 在 PyCharm 中实现实例 1: 航天器信息查询工具代码。

### 知识点 2-单继承

教师通过 PPT 讲解单继承。

- (1) 介绍单继承的格式
- (2) 通过示例代码演示如何实现单继承

### 知识点 3-多继承

教师通过 PPT 讲解多继承。

- (1) 介绍多继承的格式
- (2) 通过示例代码演示如何实现多继承

### 知识点 4-方法的重写

教师通过 PPT 讲解方法的重写。

- (1) 介绍方法重写的使用场景
- (2) 介绍如何实现方法重写
- (3) 通过示例代码演示如何实现方法重写

## 四、归纳总结

教师回顾本节课所讲的内容，并通过测试题的方式引导学生解答问题并给予指导。

## 五、布置作业——实例 1：航天器信息查询工具

教师通过高校教辅平台 (<http://tch.itxyb.com>) 布置本节课作业以及下节课的预习作业。

## 第五、六课时

### (super() 函数、实例 2：人机猜拳游戏、多态)

#### 一、复习巩固

教师通过上节课作业的完成情况，对学生吸收不好的知识点进行再次巩固讲解。

#### 二、通过直接引入的方式导入新课

在上一节中介绍了实例 1：航天器信息查询工具、单继承、多继承、方法的重写，在本节中将对 super() 函数、实例 2：人机猜拳游戏、多态进行讲解。

#### 三、新课讲解

### 知识点 1-super() 函数

教师通过 PPT 讲解 super() 函数。

- (1) 介绍 super() 函数的使用场景
- (2) 通过示例代码演示如何使用 super() 函数

### 知识点 2-实例 2：人机猜拳游戏

教师通过 PPT 结合实操的形式讲解实例 2：人机猜拳游戏。

- (1) 通过 PPT 介绍任务案例需求。
- (2) 通过 PPT 介绍任务案例分析思路。
- (3) 在 PyCharm 中实现实例 2：人机猜拳游戏代码。

### 知识点 3-多态

教师通过 PPT 讲解多态。

- (1) 介绍什么是多态
- (2) 通过示例代码演示如何定义多态

## 四、归纳总结

	<p>教师回顾本节课所讲的内容，并通过测试题的方式引导学生解答问题并给予指导。</p> <p>五、布置作业——<b>实例 2：人机猜拳游戏</b></p> <p>教师通过高校教辅平台 (<a href="http://tch.ityx.com">http://tch.ityx.com</a>) 布置本节课作业以及下节课的预习作业。</p> <p style="text-align: center;"><b>操作实训</b></p> <p>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。</p> <p><b>上机一：（考察知识点为：类的定义、对象的创建与使用、访问限制）</b>  <b>形式：独立完成</b>  <b>题目：</b></p> <ol style="list-style-type: none"> <li>(1) 练习定义类</li> <li>(2) 练习创建对象以及使用创建的对象</li> <li>(3) 练习定义私有成员和使用私有成员</li> </ol> <p><b>上机二：（考察知识点为：构造方法、类方法、静态方法）</b>  <b>形式：独立完成</b>  <b>题目：</b></p> <ol style="list-style-type: none"> <li>(1) 练习定义有参构造方法和无参构造方法</li> <li>(2) 练习定义类方法和调用类方法</li> <li>(3) 练习静态方法的定义与使用</li> </ol> <p><b>上机三：（考察知识点为：实例 1：航天器信息查询工具、单继承、多继承）</b>  <b>形式：独立完成</b>  <b>题目：</b></p> <ol style="list-style-type: none"> <li>(1) 根据任务描述的要求，编写代码实现实例 1：航天器信息查询工具。</li> <li>(2) 练习单继承和多继承的使用</li> </ol> <p><b>上机四：（考察知识点为：方法的重写、super()函数、实例 2：人机猜拳游戏）</b>  <b>形式：独立完成</b>  <b>题目：</b></p> <ol style="list-style-type: none"> <li>(1) 练习方法的重写和 super()函数的使用</li> <li>(2) 根据任务描述的要求，编写代码实现实例 2：人机猜拳游戏。</li> </ol>
教学后记	

课题名称	第 8 章 模块	计划课时	6 课时
教学引入	<p>在实际开发中，我们经常会遇到各种各样的需求，例如生成随机数、处理日期和时间、与数据库交互等，这些需求往往超出了 Python 语言本身的能力范围。为了满足这些需求，Python 提供了丰富的模块，如前面学习中接触过的 os 和 random 模块。这些模块封装了各种功能和工具，能够帮助开发人员简化开发流程，提高工作效率。此外，开发人员也可以创建自己的模块，并将多个模块组织成包。本章将针对模块和包进行讲解。</p>		
教学目标	<ul style="list-style-type: none"> <li>● 了解模块的概念，能够说出什么是模块以及模块的分类</li> <li>● 掌握模块的导入方式，能够通过 import 和 from...import...语句导入模块</li> <li>● 掌握常见内置模块的使用，能够使用 random、time 模块生成随机数或处理时间</li> <li>● 熟悉自定义模块，能够在程序中定义与使用自定义模块</li> <li>● 掌握模块的导入特性，能够在程序中合理使用 __all__ 和 __name__ 变量</li> <li>● 熟悉包的结构，能够归纳包目录下 __init__.py 文件的作用</li> <li>● 掌握包的导入方式，能够通过 import 和 from...import...语句导入包</li> <li>● 掌握第三方模块的下载与安装，能够通过 pip 命令下载与安装指定的第三方模块</li> </ul>		
教学重点	<ul style="list-style-type: none"> <li>● 常见的内置模块</li> <li>● 自定义模块</li> <li>● __all__ 变量、__name__ 变量</li> <li>● 包的结构、包的导入</li> <li>● 第三方模块的下载与安装</li> </ul>		
教学难点	<p>__all__ 变量 __name__ 变量</p>		
思政要素	<ol style="list-style-type: none"> <li>1. 培养模块化、组件化的工程化编程思维</li> <li>2. 树立资源复用、协同开发的高效编程理念</li> <li>3. 锤炼规范导入、灵活运用模块的实操能力</li> <li>4. 激发结合模块工具解决实际场景问题的应用思维</li> </ol>		
教学方式	PPT 讲授演示 + 实例编写调试		
教学过程	<p style="text-align: center;"><b>第一课时</b> <b>（模块的概念、模块的导入方式、常见的内置模块）</b></p> <p>一、复习巩固 教师通过上节课作业的完成情况，对学生吸收不好的知识点进行再次巩固讲解。</p> <p>二、通过直接引入的方式导入新课 模块（Module）是一种以.py 为后缀的 Python 文件，用于组织相关的函数、变量或类，从而扩展 Python 的功能以实现更复杂的任务。程序中每个后缀名为.py 的文件都是模块，一个模块通过导入其他模块，可以使用被导入模块的内容。本节将对模块的概念、模块的导入方式、常见的内置模块进行介绍。</p> <p><b>知识点 1-模块的概念</b> 教师通过 PPT 讲解模块的概念。</p>		

- (1) 介绍模块概念
- (2) 介绍什么是内置模块
- (3) 介绍什么是第三方模块
- (4) 介绍什么是自定义模块
- (5) 介绍模块的好处

### 知识点 2-模块的导入方式

教师通过 PPT 讲解模块的导入方式。

- (1) 使用 import 语句导入模块
- (2) 通过示例代码演示 import 语句的使用
- (3) 使用 from...import...导入模块
- (4) 通过示例代码演示 from...import...语句的使用

### 知识点 3-常见的内置模块

教师通过 PPT 讲解常见的内置模块。

#### (1) random 模块

- random()函数
- randint()函数
- choice()函数

#### (2) time 模块

- time()函数
- sleep()函数
- localtime()函数
- strftime()函数
- asctime()函数
- strptime()函数
- mktime()函数

### 三、归纳总结

教师回顾本节课所讲的内容，并通过测试题的方式引导学生解答问题并给予指导。

### 四、布置作业

教师通过高校教辅平台 (<http://tch.ityx.com>) 布置本节课作业以及下节课的预习作业。

## 第二课时

(自定义模块、\_\_all\_\_变量、\_\_name\_\_变量)

### 一、复习巩固

教师通过上节课作业的完成情况，对学生吸收不好的知识点进行再次巩固讲解。

### 二、通过直接引入的方式导入新课

在上一节中，我们介绍了模块的概念、模块的导入方式、常见的内置模块，在本节中将对自定义模块、\_\_all\_\_变量、\_\_name\_\_变量进行介绍。

### 三、新课讲解

#### 知识点 1-自定义模块

教师通过 PPT 讲解自定义模块。

- (1) 介绍如何自定义模块

(2) 通过示例演示如何自定义模块

#### 知识点 2-\_\_all\_\_ 变量

教师通过 PPT 讲解\_\_all\_\_ 变量。

(1) 介绍\_\_all\_\_ 变量的作用

(2) 通过示例代码演示\_\_all\_\_ 变量的使用

#### 知识点 3-\_\_name\_\_ 变量

教师通过 PPT 讲解构造方法。

(1) 介绍\_\_name\_\_ 变量的作用

(2) 通过示例代码演示\_\_name\_\_ 变量的使用

#### 四、归纳总结

教师回顾本节课所讲的内容，并通过测试题的方式引导学生解答问题并给予指导。

#### 五、布置作业

教师通过高校教辅平台 (<http://tch.ityx.com>) 布置本节课作业以及下节课的预习作业。

### 第三、四课时

#### (包的结构、包的导入、第三方模块的下载与安装)

#### 一、复习巩固

教师通过上节课作业的完成情况，对学生吸收不好的知识点进行再次巩固讲解。

#### 二、通过直接引入的方式导入新课

在上一节中介绍了自定义模块、\_\_all\_\_ 变量、\_\_name\_\_ 变量，在本节中将对比包的结构、包的导入、第三方模块的下载与安装进行介绍。

#### 三、新课讲解

##### 知识点 1-包的结构

教师通过 PPT 讲解包的结构。

(1) 介绍包的作用

(2) 介绍示例演示包的结构

##### 知识点 2-包的导入

教师通过 PPT 讲解包的导入。

(1) 使用 import 语句导入包的模块

(2) 使用 from...import...语句导入包的模块

##### 知识点 3-第三方模块的下载与安装

教师通过 PPT 讲解方法的重写。

(1) 介绍第三方模块的下载安装命令

(2) 通过示例演示下载安装模块

(3) 介绍在 PyChrm 下载安装模块

#### 四、归纳总结

教师回顾本节课所讲的内容，并通过测试题的方式引导学生解答问题并给予指导。

#### 五、布置作业

教师通过高校教辅平台 (<http://tch.ityx.com>) 布置本节课作业以及下节课的预习作业。

## 第五、六课时

(实例 1: 验证码、实例 2: 文本进度条、实例 3: 高考倒计时器)

### 一、复习巩固

教师通过上节课作业的完成情况,对学生吸收不好的知识点进行再次巩固讲解。

### 二、通过直接引入的方式导入新课

在上一节中介绍了包的结构、包的导入、第三方模块的下载与安装,在本节中将对实例 1: 验证码、实例 2: 文本进度条、实例 3: 高考倒计时器进行讲解。

### 三、新课讲解

#### 知识点 1-实例 1: 验证码

教师通过 PPT 结合实操的形式讲解实例 1: 验证码。

- (1) 通过 PPT 介绍任务案例需求。
- (2) 通过 PPT 介绍任务案例分析思路。
- (3) 在 PyCharm 中实现实例 1: 验证码代码。

#### 知识点 2-实例 2: 文本进度条

教师通过 PPT 结合实操的形式讲解实例 2: 文本进度条。

- (1) 通过 PPT 介绍任务案例需求。
- (2) 通过 PPT 介绍任务案例分析思路。
- (3) 在 PyCharm 中实现实例 2: 文本进度条代码。

#### 知识点 3-实例 3: 高考倒计时器

教师通过 PPT 结合实操的形式讲解实例 3: 高考倒计时器。

- (1) 通过 PPT 介绍任务案例需求。
- (2) 通过 PPT 介绍任务案例分析思路。
- (3) 在 PyCharm 中实现实例 3: 高考倒计时器代码。

### 四、归纳总结

教师回顾本节课所讲的内容,并通过测试题的方式引导学生解答问题并给予指导。

### 五、布置作业

教师通过高校教辅平台 (<http://tch.ityx.com>) 布置本节课作业以及下节课的预习作业。

## 操作实训

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: (考察知识点为: 模块的导入方式、常见的内置模块、自定义模块、`__all__` 变量)

形式: 独立完成

题目:

- (1) 练习 `import` 语句和 `from...import...` 语句的导入模块
- (2) 练习 `random` 模块和 `time` 模块中常用函数的使用
- (3) 练习自定义模块

	<p>上机二：（考察知识点为：__name__变量、包的结构、包的导入、第三方模块的下载与安装）</p> <p>形式：单独完成</p> <p>题目：</p> <p>（1）练习 import 语句和 from...import...语句的导入包</p> <p>（2）练习第三方模块的下载与安装</p> <p>上机三：（考察知识点为：实例 1：验证码、实例 2：文本进度条、实例 3：高考倒计时器）</p> <p>形式：单独完成</p> <p>题目：</p> <p>根据任务描述的要求，编写代码实现实例 1：验证码、实例 2：文本进度条、实例 3：高考倒计时器。</p>
教学后记	

课题名称	第9章 文件与文件路径操作	计划课时	6课时
教学引入	<p>程序中使用变量保存运行时产生的临时数据，然而程序一旦运行结束，它所产生的数据也会随之消失。那么，有没有一种方法能够永久保存数据呢？答案是肯定的。在计算机中，我们可以使用文件永久保存程序运行时产生的数据。文件在计算机中的存放位置各不相同，因此在操作文件之前，我们需要了解目录、文件的路径，以便能够准确地找到文件。本章将主要介绍文件的操作，并简单介绍一些与目录、文件路径相关的操作。</p>		
教学目标	<ul style="list-style-type: none"> <li>● 掌握文件的打开和关闭方式，能够通过 <code>open()</code>函数和 <code>close()</code>方法打开和关闭文件</li> <li>● 掌握读取文件的方式，能够通过 <code>read()</code>、<code>readline()</code>和 <code>readlines()</code>方法从文件中读取数据</li> <li>● 掌握写入文件的方式，能够通过 <code>write()</code>、<code>writelines()</code>方法向文件中写入数据</li> <li>● 掌握文件定位读写的方法，能够通过 <code>tell()</code>和 <code>seek()</code>方法获取和设置文件读写位置</li> <li>● 掌握文件的备份和重命名操作，能够实现文件的备份和重命名操作</li> <li>● 熟悉目录的相关操作，能够在程序中熟练地创建目录、删除目录、获取目录文件列表的操作</li> <li>● 熟悉文件路径的相关操作，能够在程序中熟练获取当前路径、检测路径的有效性和拼接路径</li> </ul>		
教学重点	<ul style="list-style-type: none"> <li>● 文件打开、关闭、读取、写入、定位读/写、重命名</li> <li>● 创建目录、删除目录、获取目录的文件列表</li> <li>● 相对路径与绝对路径、获取当前路径、检测路径的有效性、路径的拼接</li> </ul>		
教学难点	<p>文件的定位读/写 路径的拼接</p>		
思政要素	<ol style="list-style-type: none"> <li>1. 培养数据持久化、规范化管理的实操思维</li> <li>2. 树立文件备份、路径规范的安全操作意识</li> <li>3. 锤炼精准操作文件、高效处理路径的实操能力</li> <li>4. 激发用文件操作解决数据存储类实际问题的应用思维</li> <li>5. 培养严谨细致、权责清晰的文件管理与编程素养</li> </ol>		
教学方式	PPT 讲授演示 + 实例编写调试		
教学过程	<p style="text-align: center;"><b>第一课时</b></p> <p style="text-align: center;"><b>（打开文件、关闭文件、读取文件、实例 1：身份证归属地查询）</b></p> <p>一、复习巩固 教师通过上节课作业的完成情况，对学生吸收不好的知识点进行再次巩固讲解。</p> <p>二、通过需求引入的方式导入新课 程序中使用变量保存运行时产生的临时数据，然而程序一旦运行结束，它所产生的数据也会随之消失。那么，有没有一种方法能够永久保存数据呢？答案是肯定的。在计算机中，我们可以使用文件永久保存程序运行时产生的数据。本节课将对打开文件、关闭文件、读取文件、实例 1：身份证归属地查询进行介绍。</p> <p style="text-align: center;"><b>知识点 1-打开文件</b></p>		

教师通过 PPT 讲解打开文件。

- (1) 介绍 open()函数的使用
- (2) 介绍文件打开模式的使用

#### 知识点 2-关闭文件

教师通过 PPT 讲解关闭文件。

- (1) 介绍 close()方法关闭文件对象
- (2) 介绍使用 with open()语句关闭文件对象

#### 知识点 3-读取文件

教师通过 PPT 讲解读取文件。

- (1) 介绍 read()方法的使用
- (2) 介绍 readline()方法的使用
- (3) 介绍 readlines()方法的使用

#### 知识点 4-实例 1：身份证归属地查询

教师通过 PPT 结合实操的形式讲解实例 1：身份证归属地查询。

- (1) 通过 PPT 介绍任务案例需求。
- (2) 通过 PPT 介绍任务案例分析思路。
- (3) 在 PyCharm 中实现实例 1：身份证归属地查询代码。

#### 三、归纳总结

教师回顾本节课所讲的内容，并通过测试题的方式引导学生解答问题并给予指导。

#### 四、布置作业——实例 1：身份证归属地查询

教师通过高校教辅平台 (<http://tch.ityx.com>) 布置本节课作业以及下节课的预习作业。

### 第二课时

#### (写入文件、实例 2：通讯录、文件的定位读/写)

#### 一、复习巩固

教师通过上节课作业的完成情况，对学生吸收不好的知识点进行再次巩固讲解。

#### 二、通过直接引入的方式导入新课

在上一节中，我们介绍了打开文件、关闭文件、读取文件、实例 1：身份证归属地查询，在本节中将对写入文件、实例 2：通讯录、文件的定位读写进行介绍。

#### 三、新课讲解

##### 知识点 1-写入文件

教师通过 PPT 讲解写入文件。

- (1) 介绍 write()方法的使用
- (2) 介绍 writelines()方法的使用

##### 知识点 2-实例 2：通讯录

教师通过 PPT 结合实操的形式讲解实例 2：通讯录。

- (1) 通过 PPT 介绍任务案例需求。
- (2) 通过 PPT 介绍任务案例分析思路。
- (3) 在 PyCharm 中实现实例 2：通讯录代码。

##### 知识点 3-文件的定位读/写

教师通过 PPT 讲解文件的定位读写。

(1) 介绍 tell()方法的使用

(2) 介绍 seek()方法的使用

#### 四、归纳总结

教师回顾本节课所讲的内容，并通过测试题的方式引导学生解答问题并给予指导。

#### 五、布置作业——实例 2：通讯录

教师通过高校教辅平台 (<http://tch.ityx.com>) 布置本节课作业以及下节课的预习作业。

### 第三、四课时

(文件的备份、文件的重命名、创建目录、删除目录、获取目录的文件列表)

#### 一、复习巩固

教师通过上节课作业的完成情况，对学生吸收不好的知识点进行再次巩固讲解。

#### 二、通过直接引入的方式导入新课

在上一节中介绍了写入文件、实例 2：通讯录、文件的定位读/写，在本节中将对文件的备份、文件的重命名、创建目录、删除目录、获取目录的文件列表进行介绍。

#### 三、新课讲解

##### 知识点 1-文件的备份

教师通过 PPT 讲解文件的备份。

(1) 介绍文件备份逻辑

(2) 通过代码演示如何实现文件备份

##### 知识点 2-文件的重命名

教师通过 PPT 讲解文件的重命名。

(1) 介绍 rename()函数

(2) 通过示例演示 rename()函数的使用

##### 知识点 3-创建目录

教师通过 PPT 讲解创建目录。

(1) 介绍 mkdir()函数

(2) 通过示例演示 mkdir()函数的使用

##### 知识点 4-删除目录

教师通过 PPT 讲解删除目录。

(1) 介绍 rmdir()函数

(2) 通过示例演示 rmdir()函数的使用

##### 知识点 5-获取目录的文件列表

教师通过 PPT 讲解获取目录的文件列表。

(1) 介绍 listdir()函数

(2) 通过示例演示 listdir()函数的使用

#### 四、归纳总结

教师回顾本节课所讲的内容，并通过测试题的方式引导学生解答问题并给予指导。

#### 五、布置作业

教师通过高校教辅平台 (<http://tch.ityxb.com>) 布置本节课作业以及下节课的预习作业。

## 第五、六课时

### (相对路径与绝对路径、获取当前路径、检测路径的有效性、路径的拼接、实例 3: 用户登录)

#### 一、复习巩固

教师通过上节课作业的完成情况, 对学生吸收不好的知识点进行再次巩固讲解。

#### 二、通过直接引入的方式导入新课

在上一节中介绍了文件的备份、文件的重命名、创建目录、删除目录、获取目录的文件列表, 在本节中将对相对路径与绝对路径、获取当前路径、检测路径的有效性、路径的拼接、实例 3: 用户登录进行讲解。

#### 三、新课讲解

##### 知识点 1-相对路径与绝对路径

教师通过 PPT 讲解相对路径与绝对路径。

- (1) 介绍 `isabs()` 函数
- (2) 通过示例代码演示 `isabs()` 函数的使用
- (3) 介绍 `abspath()` 函数
- (4) 通过示例代码演示 `abspath()` 函数的使用

##### 知识点 2-获取当前路径

教师通过 PPT 讲解获取当前路径。

- (1) 介绍 `getcwd()` 函数
- (2) 通过示例代码演示 `getcwd()` 函数的使用

##### 知识点 3-检测路径的有效性

教师通过 PPT 讲解获取当前路径。

- (1) 介绍 `exists()` 函数
- (2) 通过示例代码演示 `exists()` 函数的使用

##### 知识点 4-路径的拼接

教师通过 PPT 讲解获取当前路径。

- (1) 介绍 `join()` 函数
- (2) 通过示例代码演示 `join()` 函数的使用

##### 知识点 5-实例 3: 用户登录

教师通过 PPT 结合实操的形式讲解实例 3: 用户登录。

- (1) 通过 PPT 介绍任务案例需求。
- (2) 通过 PPT 介绍任务案例分析思路。
- (3) 在 PyCharm 中实现实例 3: 用户登录代码。

#### 四、归纳总结

教师回顾本节课所讲的内容, 并通过测试题的方式引导学生解答问题并给予指导。

#### 五、布置作业——实例 3: 用户登录

教师通过高校教辅平台 (<http://tch.ityxb.com>) 布置本节课作业以及下节课的预习作业。

### 操作实训

	<p>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。</p> <p><b>上机一：（考察知识点为：打开文件、关闭文件、读取文件、实例 1：身份证归属地查询）</b></p> <p><b>形式：单独完成</b></p> <p><b>题目：</b></p> <ol style="list-style-type: none"> <li>（1）练习使用 open()函数打开文件</li> <li>（2）练习使用 close()方法和 with open()语句关闭文件</li> <li>（3）练习使用 read()、readline()和 readlines()方法读取文件内容</li> <li>（4）根据任务描述的要求，编写代码实现实例 1：身份证归属地查询</li> </ol> <p><b>上机二：（考察知识点为：写入文件、实例 2：通讯录、文件的定位读/写、文件的重命名）</b></p> <p><b>形式：单独完成</b></p> <p><b>题目：</b></p> <ol style="list-style-type: none"> <li>（1）练习使用 write()、writelines()方法写入文件</li> <li>（2）根据任务描述的要求，编写代码实现实例 2：通讯录</li> <li>（3）练习 tell()方法和 seek()方法的使用</li> <li>（4）练习 rename()函数的使用</li> </ol> <p><b>上机三：（考察知识点为：创建目录、删除目录、获取目录的文件列表、相对路径与绝对路径）</b></p> <p><b>形式：单独完成</b></p> <p><b>题目：</b></p> <ol style="list-style-type: none"> <li>（1）练习使用 mkdir()函数创建目录</li> <li>（2）练习使用 rmdir()函数删除目录</li> <li>（3）练习使用 listdir()函数获取目录的文件列表</li> <li>（4）练习使用 isabs()函数检测路径是否有效</li> <li>（5）练习使用 abspath()函数将相对路径转换为绝对路径</li> </ol> <p><b>上机四：（考察知识点为：获取当前路径、检测路径的有效性、路径的拼接、实例 3：用户登录）</b></p> <p><b>形式：单独完成</b></p> <p><b>题目：</b></p> <ol style="list-style-type: none"> <li>（1）练习使用 getcwd()函数获取当前路径</li> <li>（2）练习使用 exists()函数检测指定路径是否有效</li> <li>（3）练习使用 join()函数对路径进行拼接</li> <li>（4）根据任务描述的要求，编写代码实现实例 3：用户登录</li> </ol>
教学后记	

课题名称	第 10 章 错误和异常	计划课时	3 课时
教学引入	<p>在现实生活中，我们经常会遇到各种突发情况，比如航班延误、火车晚点、交通堵塞等。而在程序中同样不可避免地会出现异常，比如试图打开一个不存在的文件、访问未定义的变量等，这些异常默认情况下都会导致程序崩溃而无法正常运行。为了增强程序的稳定性，Python 提供了强大的异常处理机制，它不仅可以捕获以及处理异常，防止程序崩溃，还提供详细的诊断信息，帮助开发人员快速定位和解决问题。接下来，本章将针对错误和异常的内容进行讲解。</p>		
教学目标	<ul style="list-style-type: none"> <li>● 了解错误和异常，能够说出什么是错误和异常</li> <li>● 熟悉异常的类型，能够理解常见异常类的含义</li> <li>● 掌握 try-except 语句的使用，能够在程序中通过 try-except 语句捕获处理异常</li> <li>● 掌握 try-except-else 语句的使用，能够通过 try-except-else 语句捕获异常，并在 else 子句中添加没有异常的处理代码</li> <li>● 掌握 try-except-finally 语句的使用，能够通过 try-except-finally 语句捕获异常，并在 finally 子句中添加释放资源的代码</li> <li>● 掌握 raise 语句的使用，能够在程序中通过 raise 语句抛出异常</li> <li>● 了解异常的传递，能够说出异常传递的特点</li> <li>● 掌握 assert 断言语句的使用，能够在程序中通过 assert 语句抛出异常</li> <li>● 掌握自定义异常的方法，能够在程序中自定义并处理异常</li> <li>● 熟悉 with 语句与上下文管理器，能够在程序中使用 with 语句释放资源</li> </ul>		
教学重点	<ul style="list-style-type: none"> <li>● try-except 语句</li> <li>● else 子句</li> <li>● finally 子句</li> <li>● raise 语句</li> <li>● asser 断言语句</li> <li>● 自定义异常</li> </ul>		
教学难点	<p>finally 子句 自定义异常</p>		
思政要素	<ol style="list-style-type: none"> <li>1. 培养预判风险、主动容错的程序设计思维</li> <li>2. 树立责任意识，养成资源释放、异常处理的规范编程习惯</li> <li>3. 锤炼排查问题、精准定位错误的问题解决能力</li> <li>4. 激发严谨验证、自定义规则的创新思维</li> <li>5. 培养面对异常不慌乱、积极应对的抗压素养</li> </ol>		
教学方式	PPT 讲授演示 + 实例编写调试		
教学过程	<p style="text-align: center;"><b>第一课时</b> <b>(认识错误和异常、异常的类型、try-except 语句)</b></p> <p>一、复习巩固 教师通过上节课作业的完成情况，对学生吸收不好的知识点进行再次巩固讲解。</p> <p>二、通过需求引入的方式导入新课 在现实生活中，我们经常会遇到各种突发情况，比如航班延误、火车晚点、交通堵塞等。而在程序中同样不可避免地会出现异常，比如试图打开一个不存在</p>		

的文件、访问未定义的变量等，这些异常默认情况下都会导致程序崩溃而无法正常运行。为了增强程序的稳定性，Python 提供了强大的异常处理机制，它不仅捕获以及处理异常，防止程序崩溃，还提供详细的诊断信息，帮助开发人员快速定位和解决问题。本节将对认识错误和异常、异常的类型、try-except 语句进行介绍。

#### 知识点 1-认识错误和异常

教师通过 PPT 讲解认识错误和异常。

- (1) 介绍什么是语法错误
- (2) 介绍什么逻辑错误

#### 知识点 2-异常类型

教师通过 PPT 讲解异常类型。

- (1) 介绍 Python 中异常类的继承关系
- (2) 介绍 Exception 常见子类

- SyntaxError
- FileNotFoundError
- NameError
- ZeroDivisionError
- IndexError
- KeyError
- AttributeError
- TypeError

#### 知识点 3-try-except 语句

教师通过 PPT 讲解读取文件。

- (1) 介绍 try-except 语句的语法格式
- (2) 介绍 try-except 语句的执行流程
- (3) 介绍捕获单个异常
- (4) 介绍捕获多个异常
- (5) 介绍捕获所有异常

### 三、归纳总结

教师回顾本节课所讲的内容，并通过测试题的方式引导学生解答问题并给予指导。

### 四、布置作业

教师通过高校教辅平台 (<http://tch.ityx.com>) 布置本节课作业以及下节课的预习作业。

## 第二课时

### (捕获异常信息、else 子句、finally 子句、raise 语句)

#### 一、复习巩固

教师通过上节课作业的完成情况，对学生吸收不好的知识点进行再次巩固讲解。

#### 二、通过直接引入的方式导入新课

在上一节中，我们介绍了认识错误和异常、异常的类型、try-except 语句，在本节中将对捕获异常信息、else 子句、finally 子句、raise 语句进行介绍。

#### 三、新课讲解

### 知识点 1-捕获异常信息

教师通过 PPT 讲解捕获异常信息。

- (1) 介绍如何捕获异常信息
- (2) 通过示例代码演示捕获异常信息

### 知识点 2-else 子句

教师通过 PPT 讲解 else 子句。

- (1) 介绍 try-except-else 语句的格式
- (2) 通过示例代码演示 try-except-else 语句的使用

### 知识点 3-finally 子句

教师通过 PPT 讲解 finally 子句。

- (1) 介绍 try-except-finally 语句的格式
- (2) 通过示例代码演示 try-except-finally 语句的使用

### 知识点 4-raise 语句

教师通过 PPT 讲解 raise 语句。

- (1) 介绍使用异常类引发异常
- (2) 介绍使用异常类对象引发异常
- (3) 介绍重新引发异常

## 四、归纳总结

教师回顾本节课所讲的内容，并通过测试题的方式引导学生解答问题并给予指导。

## 五、布置作业

教师通过高校教辅平台 (<http://tch.ityx.com>) 布置本节课作业以及下节课的预习作业。

## 第三课时

(异常的传递、asser 断言语句、自定义异常、with 语句、实例：身份归属地查询)

### 一、复习巩固

教师通过上节课作业的完成情况，对学生吸收不好的知识点进行再次巩固讲解。

### 二、通过直接引入的方式导入新课

在上一节中介绍了捕获异常信息、else 子句、finally 子句、raise 语句，在本节中将对异常的传递、asser 断言语句、自定义异常、with 语句、实例：身份归属地查询进行介绍。

### 三、新课讲解

#### 知识点 1-异常的传递

教师通过 PPT 讲解异常的传递。

- (1) 接收异常传递顺序
- (2) 通过示例代码介绍异常传递的顺序

#### 知识点 2-assert 断言语句

教师通过 PPT 讲解 assert 断言语句。

- (1) 介绍 assert 断言语句的格式
- (2) 通过示例代码演示 assert 的使用

#### 知识点 3-自定义异常

教师通过 PPT 讲解自定义异常。

- (1) 介绍自定义异常的步骤
- (2) 通过示例代码演示自定义异常

#### 知识点 4-with 语句

教师通过 PPT 讲解 with 语句。

- (1) 介绍 with 语句的语法格式
- (2) 介绍 with 语句如何关闭资源

#### 知识点 5-实例：身份归属地查询

教师通过 PPT 结合实操的形式讲解实例：身份证归属地查询。

- (1) 通过 PPT 介绍任务案例需求。
- (2) 通过 PPT 介绍任务案例分析思路。
- (3) 在 PyCharm 中实现实例：身份证归属地查询代码。

#### 四、归纳总结

教师回顾本节课所讲的内容，并通过测试题的方式引导学生解答问题并给予指导。

#### 五、布置作业

教师通过高校教辅平台 (<http://tch.itymb.com>) 布置本节课作业以及下节课的预习作业。

### 操作实训

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：（考察知识点为：try-except 语句、捕获异常信息、else 子句、finally 子句）

形式：独立完成

题目：

- (1) 练习使用 try-except 语句捕获异常
- (2) 练习使用 try-except-else 语句捕获异常
- (3) 练习使用 try-except-finally 语句捕获异常

上机二：（考察知识点为：raise 语句、asser 断言语句、自定义异常、实例：身份归属地查询）

形式：独立完成

题目：

- (1) 练习使用 raise 语句抛出异常
- (2) 练习使用 assert 断言判定一个表达式是否为真
- (3) 练习自定义异常
- (4) 根据任务描述的要求，编写代码实现实例：身份证归属地查询

教学后记	
------	--

课题名称	第 11 章 正则表达式	计划课时	4 课时
教学引入	<p>网站开发人员需要对用户在注册页面提交的信息进行验证，包括手机号、用户名、邮箱等。由于这些信息遵循复杂的规则，仅使用条件语句判断将增加工作量。然而，正则表达式解决了这个问题。正则表达式是一种描述字符串结构的语法规则，它在字符串的查找、匹配、替换等方面非常强大，并且被许多编程语言广泛支持，包括 Python。本章将介绍如何在 Python 中使用正则表达式。</p>		
教学目标	<ul style="list-style-type: none"> <li>● 熟悉元字符，能够归纳元字符的功能以及基本用法</li> <li>● 熟悉预定义字符集，能够归纳预定义字符集的功能</li> <li>● 了解 re 模块，能够说出 re 模块中常用函数或方法的作用</li> <li>● 掌握预编译的方式，能够通过 compile()函数将正则表达式预编译为正则对象</li> <li>● 掌握匹配与搜索的方式，能够通过 match()与 search()函数实现匹配与搜索的功能</li> <li>● 掌握匹配对象，能够通过匹配对象的方法获取匹配结果中的各项数据</li> <li>● 掌握全文匹配的方式，能够通过 findall()与 finditer()函数实现全文匹配功能</li> <li>● 掌握检索替换的方式，能够通过 sub()、subn()函数实现检索替换的功能</li> <li>● 掌握文本分割的方式，能够通过 split()函数实现文本分割的功能</li> <li>● 熟悉贪婪匹配，能够区分贪婪匹配和非贪婪匹配的不同</li> </ul>		
教学重点	<ul style="list-style-type: none"> <li>● re 模块</li> <li>● 预编译</li> <li>● 使用 match()函数进行匹配</li> <li>● 使用 search()函数进行匹配</li> <li>● 匹配对象</li> <li>● findall()函数</li> <li>● finditer()函数</li> <li>● 检索替换</li> <li>● 文本分割</li> <li>● 贪婪匹配</li> </ul>		
教学难点	<ul style="list-style-type: none"> <li>● 使用 match()函数进行匹配</li> <li>● 使用 search()函数进行匹配</li> <li>● 匹配对象</li> <li>● 贪婪匹配</li> </ul>		
思政要素	<ol style="list-style-type: none"> <li>1. 培养精准匹配、规则化处理文本的逻辑思维</li> <li>2. 树立数据验证、信息筛选的严谨编程意识</li> <li>3. 锤炼灵活运用正则语法解决复杂文本问题的实操能力</li> <li>4. 激发结合实际场景（如注册验证）设计匹配规则的创新思维</li> </ol>		
教学方式	PPT 讲授演示 + 实例编写调试		
教学过程	<p style="text-align: center;"><b>第一课时</b> <b>（元字符、预定义字符集、re 模块）</b></p> <p>一、复习巩固 教师通过上节课作业的完成情况，对学生吸收不好的知识点进行再次巩固讲解。</p> <p>二、通过直接引入的方式导入新课</p>		

正则表达式实际上是一种符号语言，用于定义一组文本模式的匹配规则。每个正则表达式也称为一个模式，通过这些模式可以匹配指定文本中与表达式模式相同的字符串。本节将对元字符、预定义字符集、re 模块进行介绍。

### 知识点 1-元字符

教师通过 PPT 讲解元字符。

- (1) 介绍元字符的作用
- (2) 介绍常见元字符的匹配规则

- 点字符 “.”
- 脱字符 “^” 和美元符 “\$”
- 连接符 “[”
- 字符组 “[ ]”
- 连字符 “-”
- 元字符 “?”
- 元字符 “\*”、“+” 以及 {n}/{m,n}
- 元字符 “()”

### 知识点 2-预定义字符集

教师通过 PPT 讲解预定义字符集。

- (1) 介绍预定义字符集的作用
- (2) 介绍预定义字符集的匹配规则

- \w: 匹配下划线或任何字母与数字
- \s: 匹配任意的空白字符
- \d: 匹配任意数字
- \b: 匹配单词的边界
- \W: 与 \w 相反，匹配特殊字符
- \S: 与 \s 相反，匹配任意非空白字符的字符
- \D: 与 \d 相反，匹配任意非数字的字符
- \B: 与 \b 相反，匹配不出现在单词边界的元素
- \A: 仅匹配字符串的开头
- \Z: 仅匹配字符串的结尾

### 知识点 3-re 模块

教师通过 PPT 讲解 re 模块。

- (1) 介绍 re 模块的作用
- (2) 介绍 re 模块常用函数的使用

- compile()
- match()
- search()
- split()
- findall()
- finditer()
- sub()
- subn()
- group()
- groups()

三、归纳总结

教师回顾本节课所讲的内容，并通过测试题的方式引导学生解答问题并给予指导。

#### 四、布置作业

教师通过高校教辅平台 (<http://tch.ityx.com>) 布置本节课作业以及下节课的预习作业。

### 第二课时

#### (预编译、使用 `match()` 函数进行匹配、使用 `search()` 函数进行匹配、实例 1：手机号运营商)

##### 一、复习巩固

教师通过上节课作业的完成情况，对学生吸收不好的知识点进行再次巩固讲解。

##### 二、通过直接引入的方式导入新课

在上一节中，我们介绍了元字符、预定义字符集、`re` 模块，在本节中将对预编译、使用 `match()` 函数进行匹配、使用 `search()` 函数进行匹配、实例 1：手机号运营商进行介绍。

##### 三、新课讲解

###### 知识点 1-预编译

教师通过 PPT 讲解预编译。

- (1) 介绍 `compile()` 函数的作用
- (2) 介绍参数 `flags` 匹配模式的说明

- `re.I`
- `re.L`
- `re.M`
- `re.S`
- `re.U`
- `re.A`
- `re.X`

- (3) 通过示例代码演示 `compile()` 函数的使用

###### 知识点 2-使用 `match()` 函数进行匹配

教师通过 PPT 讲解使用 `match()` 函数进行匹配。

- (1) 介绍 `match()` 函数的作用
- (2) 介绍 `match()` 函数参数含义
- (3) 通过示例代码演示 `match()` 函数的使用

###### 知识点 3-使用 `search()` 函数进行匹配

教师通过 PPT 讲解使用 `search()` 函数进行匹配。

- (1) 介绍 `search()` 函数的作用
- (2) 介绍 `search()` 函数的参数含义
- (3) 通过示例代码演示 `search()` 函数的使用

###### 知识点 4-实例 1：手机号运营商

教师通过 PPT 结合实操的形式讲解实例 1：手机号运营商。

- (1) 通过 PPT 介绍任务案例需求。
- (2) 通过 PPT 介绍任务案例分析思路。
- (3) 在 PyCharm 中实现实例 1：手机号运营商代码。

#### 四、归纳总结

教师回顾本节课所讲的内容，并通过测试题的方式引导学生解答问题并给予指导。

#### 五、布置作业

教师通过高校教辅平台 (<http://tch.ityx.com>) 布置本节课作业以及下节课的预习作业。

### 第三课时

#### (匹配对象、findall()函数、finditer()函数、检索替换)

##### 一、复习巩固

教师通过上节课作业的完成情况，对学生吸收不好的知识点进行再次巩固讲解。

##### 二、通过直接引入的方式导入新课

在上一节中介绍了预编译、使用 match()函数进行匹配、使用 search()函数进行匹配、实例 1: 手机号运营商，在本节中将对匹配对象、findall()函数、finditer()函数、检索替换进行介绍。

##### 三、新课讲解

###### 知识点 1-匹配对象

教师通过 PPT 讲解匹配对象。

- (1) 介绍匹配对象中包含的属性所代表的含义
- (2) 介绍匹配对象常用的方法

- group()方法
- start()方法
- end()方法
- span()方法

- (3) 通过示例代码演示匹配对象常用方法的使用

###### 知识点 2-finally()函数

教师通过 PPT 讲解 finally()函数。

- (1) 介绍 finally()函数的作用
- (2) 介绍 finally()函数参数的含义
- (3) 通过示例代码演示 finally()函数的使用

###### 知识点 3-finditer()函数

教师通过 PPT 讲解 finditer()函数。

- (1) 介绍 finditer()函数的作用
- (2) 介绍 finditer()函数参数的含义
- (3) 通过示例代码演示 finditer()函数的使用

###### 知识点 4-检索替换

教师通过 PPT 讲解检索替换。

- (1) 介绍 sub()函数和 subn()函数的作用
- (2) 介绍 sub()函数和 subn()函数参数的含义

- patten
- repl
- string
- count

- flags

(3) 通过示例代码演示 sub()函数和 subn()函数的使用

#### 四、归纳总结

教师回顾本节课所讲的内容,并通过测试题的方式引导学生解答问题并给予指导。

#### 五、布置作业

教师通过高校教辅平台 (<http://tch.ityx.com>) 布置本节课作业以及下节课的预习作业。

### 第四课时

**(实例 2: 电影信息提取、文本分割、贪婪匹配、实例 3: 用户注册验证)**

#### 一、复习巩固

教师通过上节课作业的完成情况,对学生吸收不好的知识点进行再次巩固讲解。

#### 二、通过直接引入的方式导入新课

在上一节中介绍了匹配对象、findall()函数、finditer()函数、检索替换,在本节中将对实例 2: 电影信息提取、文本分割、贪婪匹配、实例 3: 用户注册验证进行讲解。

#### 三、新课讲解

##### 知识点 1-实例 2: 电影信息提取

教师通过 PPT 结合实操的形式讲解实例 2: 电影信息提取。

- (1) 通过 PPT 介绍任务案例需求。
- (2) 通过 PPT 介绍任务案例分析思路。
- (3) 在 PyCharm 中实现实例 2: 电影信息提取代码。

##### 知识点 2-文本分隔

教师通过 PPT 讲解文本分隔。

- (1) 介绍 split()函数的作用
- (2) 介绍 split()函数参数的含义
- (3) 通过示例代码演示 split()函数的使用

##### 知识点 3-贪婪匹配

教师通过 PPT 讲解贪婪匹配。

- (1) 介绍什么是贪婪匹配
- (2) 通过示例代码演示贪婪匹配规则
- (3) 介绍什么是非贪婪匹配
- (4) 通过示例代码演示非贪婪匹配

##### 知识点 4-实例 3: 用户注册验证

教师通过 PPT 讲解获取当前路径。

- (1) 通过 PPT 介绍任务案例需求。
- (2) 通过 PPT 介绍任务案例分析思路。
- (3) 在 PyCharm 中实现实例 3: 用户注册验证代码。

#### 四、归纳总结

教师回顾本节课所讲的内容,并通过测试题的方式引导学生解答问题并给予指导。

#### 五、布置作业

	<p>教师通过高校教辅平台 (<a href="http://tch.ityxb.com">http://tch.ityxb.com</a>) 布置本节课作业以及下节课的预习作业。</p> <p style="text-align: center;"><b>操作实训</b></p> <p>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。</p> <p><b>上机一: (考察知识点为: 预编译、使用 <code>match()</code> 函数进行匹配)</b>  <b>形式: 独立完成</b>  <b>题目:</b></p> <ol style="list-style-type: none"> <li>(1) 练习使用 <code>re</code> 模块 <code>compile()</code> 函数的使用</li> <li>(2) 练习使用 <code>match()</code> 函数匹配正则表达式</li> </ol> <p><b>上机二: (考察知识点为: 使用 <code>search()</code> 函数进行匹配、实例 1: 手机号运营商、匹配对象)</b>  <b>形式: 独立完成</b>  <b>题目:</b></p> <ol style="list-style-type: none"> <li>(1) 练习使用 <code>search()</code> 函数匹配正则表达式</li> <li>(2) 根据任务描述的要求, 编写代码实现实例 1: 手机号运营商</li> <li>(3) 练习 <code>group()</code> 方法、<code>start()</code> 方法、<code>end()</code> 方法和 <code>span()</code> 方法的使用</li> </ol> <p><b>上机三: (考察知识点为: <code>findall()</code> 函数、<code>finditer()</code> 函数、检索替换)</b>  <b>形式: 独立完成</b>  <b>题目:</b></p> <ol style="list-style-type: none"> <li>(1) 练习使用 <code>findall()</code> 函数匹配正则表达式</li> <li>(2) 练习使用 <code>finditer()</code> 函数匹配正则表达式</li> <li>(3) 练习使用 <code>sub()</code>、<code>subn()</code> 函数替换目标字符串的匹配项</li> </ol> <p><b>上机四: (考察知识点为: 实例 2: 电影信息提取、文本分割、贪婪匹配、实例 3: 用户注册验证)</b>  <b>形式: 独立完成</b>  <b>题目:</b></p> <ol style="list-style-type: none"> <li>(1) 根据任务描述的要求, 编写代码实现实例 2: 电影信息提取</li> <li>(2) 练习使用 <code>split()</code> 函数分割指定文本</li> <li>(3) 练习贪婪匹配模式和非贪婪匹配模式匹配字符串</li> <li>(4) 根据任务描述的要求, 编写代码实现实例 3: 用户注册验证</li> </ol>
教学后记	