

教案三 定义变量

单元标题	定义变量	单元教学学时	4 课时
授课场所	一体化实训室	授课形式	线上线下混合模式
在课程中的位置			
学习内容	<ol style="list-style-type: none"> 1. 数字类型表示方法 2. 字符串的常见操作、字符串的格式化输出、字符串的索引与切片 3. 运算符及优先级 		
学情分析	<p>2020 级软件技术专业大二的学生思维活跃，对软件开发有浓厚的学习兴趣，但学习主动性较差，综合运用知识的能力不足。该学生大一上学期已经开设《面向对象程序设计（Java）》，有一定的编程基础。学生能够搭建 Python 开发环境、创建 Python 项目、掌握基本的输入输出语句。通过课前测试大数据分析，学生的知识迁移能力较弱，不理解先行课程 Java 中定义变量和本课程定义变量的区别与联系。</p>		
教学	思政目标	知识目标	能力目标
目标	<ol style="list-style-type: none"> 1. 理解事物的联系普遍存在的，引导学生用类比的方法进行知识的迁移 2. 树立社会主义荣辱观，从自身做起，在以积极态度促进互联网健康发展，坚持文明上网 	<ol style="list-style-type: none"> 1. 掌握数字类型表示方法 2. 掌握字符串的常见操作 3. 掌握字符串的索引切片 4. 熟练使用运算符，明确混合运算中运算符的优先级 	<ol style="list-style-type: none"> 1. 能够熟练创建数字类型、字符串类型变量 2. 能够熟练操作字符串
课程思政	融入知识点	<ol style="list-style-type: none"> 1. Java 定义变量与 Python 定义变量的区别与联系 2. 敏感词替换 	
	融入方式	通过知识点、案例融入思政	
	思政元素	<ol style="list-style-type: none"> 1. 通过 Java 定义变量与 Python 定义变量的区别与联系，让学生理解事物的联系普遍存在的，引导学生用类比的方法进行知识的迁移。 2. 通过敏感词替换案例，引导学生树立社会主义荣辱观，从自身做起，在以积极态度促进互联网健康发展，承担起应负的社会责任，提倡正确导向，反对不良网风，坚持文明上网。 	

	<p>思政资源</p> <p>文章：文明上网自律公约</p> <p>视频：文明上网，我们应该怎么做</p> <p>视频：如何利用知识迁移能力快速进入一个领域</p>										
<p>教学重点、难点</p>	<p>教学重点：字符串的格式化输出；字符串的常见操作</p> <p>教学难点：字符串的常见操作</p>										
<p>教学方法与教学手段</p>	<p>1. 教学方法：</p> <p>(1) 介绍/演示 Python 常用的数据类型</p> <p>(2) 介绍/演示数字类型的表示方法</p> <p>(3) 通过案例“模拟超市收银抹零行为”讲解数字类型</p> <p>(4) 示例说明字符串的定义、格式化输出和常见操作</p> <p>(5) 通过案例“敏感词替换”讲解字符串的用法</p> <p>(6) 分组讨论 Python 中定义变量和 Java 的区别，培养学生的知识迁移能力</p> <p>(7) 示例讲解比较运算符、赋值运算符、逻辑运算符和运算符的优先级</p> <p>(8) 提问“10<<2 的结果”</p> <p>2. 辅助手段：</p> <p>(1) 多媒体演示</p> <p>(2) 视频讲解</p> <p>(3) 在线教学平台在线测试，大数据分析测试结果</p> <p>3. 对于重点和难点，通过案例讨论讲解、师生互动、在线测试、动画演示等解决和突破。</p>										
<p>课前需掌握内容</p>	<p>1. 二进制、八进制、十进制、十六进制数</p> <p>2. 数制之间的转换</p>										
<p>教学内容设计</p>	<p>任务一 定义数字类型变量</p> <p>任务二 定义与操作字符串</p> <p>任务三 运用运算符</p>										
<p>教学资源</p>	<table border="1"> <thead> <tr> <th>资源类型</th> <th>数量</th> </tr> </thead> <tbody> <tr> <td>教学设计/教案/课件/实训指导书/练习题</td> <td>8</td> </tr> <tr> <td>微课/视频/思政视频/音频答疑</td> <td>30</td> </tr> <tr> <td>思维导图/流程图</td> <td>2</td> </tr> <tr> <td>案例源码/推荐学习内容</td> <td>9</td> </tr> </tbody> </table>	资源类型	数量	教学设计/教案/课件/实训指导书/练习题	8	微课/视频/思政视频/音频答疑	30	思维导图/流程图	2	案例源码/推荐学习内容	9
资源类型	数量										
教学设计/教案/课件/实训指导书/练习题	8										
微课/视频/思政视频/音频答疑	30										
思维导图/流程图	2										
案例源码/推荐学习内容	9										
<p>课后拓展作业</p>	<p>1. 运用“三斜求积术”输出三角形面积；</p> <p>2. 课程平台拓展作业。</p>										

教学环节设计

一、课前准备

环节 用时	内容	教师活动	学生活动	教学方法 与手段
学习 微课 15 分钟	学习《数字类型的表示方法》相关微课	1. 发布预习通知; 2. 上传微课	学习微课、 课件、教案	利用教学平台完成 课前预习
课前 测试 10 分钟	<p>1. 下列函数中, 可以将数值类型转换为字符串的是()。</p> <p>A. <code>complex()</code> B. <code>int()</code> C. <code>float()</code> D. <code>str()</code></p> <p>2. 下列关于 Python 字符串的说法中, 错误的是()</p> <p>A. 字符串是用来表示文本的数据类型</p> <p>B. Python 中可以使用单引号、双引号、三引号定义字符串</p> <p>C. 单引号定义的字符串中不能包含双引号字符</p> <p>D. 使用三引号定义的字符串可以包含换行符</p> <p>3. 已知变量 <code>name = "张昊"</code>、<code>age = 18</code>, 下列字符串格式化输出, 错误的是()</p> <p>A. <code>print(' 我叫 %s, 今年我 %d 岁了' %(age, name))</code></p> <p>B. <code>print(' 我叫 %s, 今年我 %d 岁了' %(name, age))</code></p> <p>C. <code>print(' 我叫 {}, 今年我 {} 岁了' .format(name, age))</code></p> <p>D. <code>print(f' 我叫 {name}, 今年我 {age} 岁了')</code></p>	1. 发布测试题; 2. 查看测试结果; 3. 调整教学策略	使用手机 做题	1.利用教学平台完成 课前测试; 2. 运用大数据开展 学习行为分析
课前 作业 10 分钟	<p>1. 二进制、八进制、十进制、十六进制数</p> <p>2. 数制之间的转换</p>	1. 发布课前作业; 2. 查看学生作业	各小组完成 作业内容, 将作业上传 至教学平台。	利用教学平台开展 作业分析

教学环节设计

二、课堂实施（4课时）

环节 用时	内容	教师活动	学生活动	教学方法 与手段
导入 课程 5分钟	1. Python 程序中基本的数据类型有哪些？ 2. Python 程序中运算符有哪些？	1. 提出问题； 2. 简单介绍 计算机语言的 发展与分 类。	1. 思考问题 2. 小组讨论 3. 回答问题	1. 利用多 媒体教 学； 2. 利用教 学平台开 展讨论
查看课 前预习 情况 10分钟	1. 要求学生上台讲解二进制、八进制、十进制、 十六进制数的特点以及进制之间的转换； 2. 讲解错误率高的题目	1. 提出问题； 2. 总结学生 操作情况； 3. 讲解错误 率高的题目	1. 学生代表 上台讲解； 2. 其他学生 观看并提出 错误； 3. 改正错误	1. 利用教 学平台课 前分析测 试情况； 2. 多媒体 课件；
分析 问题 5分钟	1. 数字类型变量：整型、浮点型、布尔类型、复 数； 2. 字符串变量：定义字符串、字符串切片、字 符串操作方法； 3. 运算符：算数运算符、逻辑运算符、比较运 算符、位运算符	1. 演示开发 环境； 2. 引出学习 目的及重点、 难点	1. 观看运行 效果； 2. 小组讨论 Python 开发 环境需要哪 些软件； 3. 小组代表 发言	1. 多媒体 课件； 2. 运用 PyCharm 软件运行 效果

环节 用时	内容	教师活动	学生活动	教学方法 与手段
学习 新知 25 分钟	<p style="text-align: center;">任务一 定义数字类型变量</p> <p>Python 中的数字类型及其表示方法： 表示数字或数值的数据类型称为数字类型。</p> <p>Python 内置的数字类型有整型（int）、浮点型（float）、复数类型（complex）、布尔类型（bool）。</p> <p>1. 整型：Python 中使用 4 种进制表示整型，分别为二进制、八进制、十进制和十六进制。</p> <pre>0b1010 # 二进制 0o12 # 八进制 10 # 十进制 0xA # 十六进制</pre> <p>2. 浮点数：一般以十进制形式表示，对于较大或较小的浮点数，可以使用科学计数法表示。</p> <pre>num_one = 3.14 # 十进制形式表示 num_two = 2e2 # 科学计数法表示 num_third = 2e-2 # 科学计数法表示</pre> <p>3. 复数类型：Python 中有两种创建复数的方式，一种是按照复数的一般形式直接创建，另一种是通过内置函数 complex() 创建。</p> <pre>num_one = 3 + 2j # 直接创建 num_two = complex(3, 2) # 通过内置函数创建</pre> <p>4. 布尔类型：Python 中的布尔类型（bool）只有两个取值：True 和 False。</p> <p>Python 中的任何对象都可以转换为布尔类型，若要进行转换，符合以下条件的数据都会被转换为 False：</p> <pre>None;</pre> <p>任何为 0 的数字类型，如 0、0.0、0j；</p> <p>任何空序列，如 ""、()、[]；</p> <p>任何空字典，如 {}；</p> <p>5. 类型转换函数：Python 内置了一系列可实现强制类型转换的函数。数字间进行转换的函数有 int()、float()、str()。</p>	<ol style="list-style-type: none"> 1. 讲授知识点 2. 提出问题：常用进制数有哪些，如何进行进制数的转换？ 3. 总结回答情况 	<ol style="list-style-type: none"> 1. 认真听讲 2. 思考问题 3. 小组讨论 4. 回答问题 	<ol style="list-style-type: none"> 1. 多媒体教学； 2. 分组讨论； 3. 教学平台 4. PyCharm 开发环境

环节 用时	内容	教师活动	学生活动	教学方法 与手段
<p>学习 新知</p> <p>45 分钟</p>	<p style="text-align: center;">任务二 定义与操作字符串</p> <p>1. 字符串的定义</p> <p>字符串是一种用来表示文本的数据类型，它是由符号或者数值组成的一个连续序列，Python 中的字符串是不可变的，字符串一旦创建便不可修改。Python 支持使用单引号、双引号和三引号定义字符串，其中单引号和双引号通常用于定义单行字符串，三引号通常用于定义多行字符串，具体示例分别如下。</p> <p>2. 字符串的格式化输出</p> <p>字符串格式化输出的方式有 3 种分别为：占位符%、format()方法和 f-strings 三种方式实现格式化输出。</p> <p>3. 字符串的常见操作</p> <p>在 Python 中提供多种字符串操作，常见的操作有使用“+”进行字符串拼接、使用 replace()方法替换、使用 split()方法分隔字符串、使用 strip()方法去除两侧空格。</p> <p>4. 字符串的索引与切片</p> <p>在程序的开发过程中，可能需要对一组字符串中的某些字符进行特定的操作，Python 中通过字符串的索引与切片功能可以提取字符串中的特定字符或子串。</p> <p>字符串的正向索引从 0 开始计算，反向索引从-1 开始计数。</p> <p>字符串的切片语法格式为：字符串[起始:结束:步长]。</p> <p>5. 字符串处理方法</p> <p>Python 提供了许多字符串操作方法，常用的操作字符串方法可以分为大小写转换（lower()与 upper）、判断（具体函数详见表 3-15）、填充字符串（ljust()、center()、rjust()）、字符串搜索（find()、index()）、替换（replace()）、分隔（split()）等。</p>	<p>1. 讲授知识点</p> <p>2. 提出问题：Python 字符串的定义与操作与 Java 有何区别，培养学生知识迁移能力？</p> <p>3. 总结回答情况</p> <p>4. 提问：字符串常用的处理方法有哪些？</p>	<p>1. 认真听讲</p> <p>2. 思考问题通过思考 Python 字符串的定义与操作与 Java 有何区别，提升知识迁移能力</p> <p>3. 小组讨论</p> <p>4. 回答问题</p>	<p>1. 多媒体教学；</p> <p>2. 分组讨论；</p> <p>3. 教学平台</p> <p>4. PyCharm 开发环境</p>

环节 用时	内容	教师活动	学生活动	教学方法 与手段
学习 新知 45 分钟	<p style="text-align: center;">任务三 运用运算符</p> <p>1. 算术运算符的使用 Python 中的算术运算符包括+、-、*、/、//、% 和**，这些运算符都是双目运算符，一个运算符可以和两个操作数组成一个表达式。</p> <p>2. 比较运算符的使用 Python 中的比较运算符有：==、!=、>、<、>=、<=，比较运算符同样是双目运算符，它与两个操作数构成一个表达式。</p> <p>3. 赋值运算符的使用 赋值运算符的功能是：将一个表达式或对象赋给一个左值，其中左值必须是一个可修改的值，不能为一个常量。“=”是基本的赋值运算符，此外“=”可与算术运算符组合成复合赋值运算符。Python 中的复合赋值运算符有：+=、-=、*=、/=、//=、%=、**=。</p> <p>4. 逻辑运算符的使用 Python 支持逻辑运算，但 Python 逻辑运算符的功能与其它语言有所不同。Python 中分别使用“or”、“and”、“not”这三个关键字作为逻辑运算“或”、“与”、“非”的运算符，其中 or 与 and 为双目运算符，not 为单目运算符。</p> <p>5. 位运算符的使用 程序中的所有数据在计算机内存中都以二进制形式存储，位运算即以二进制位为单位进行的运算。Python 的位运算主要包括按位左移、按位右移、按位与、按位或、按位异或、按位取反这 6 种。</p> <p style="color: red;">课程思政：通过观看视频《如何利用知识迁移能力快速进入一个领域》，让学生明白知识迁移的重要性，培养学生知识迁移的能力。</p>	<p>1. 讲授知识点</p> <p>2. 提出问题：Python3.x 中算术运算符有哪些？</p> <p>3. 总结回答情况</p> <p>4. 提问：“or”、“and”、“not”操作数与运算符结果的关系，并通过示例解答问题</p>	<p>1. 认真听讲</p> <p>2. 思考问题</p> <p>3. 小组讨论</p> <p>4. 通过示例解答“or”、“and”、“not”运算符的用法</p>	<p>1.多媒体教学；</p> <p>2.分组讨论；</p> <p>3.教学平台</p> <p>4.PyCharm 开发环境</p>

环节用时	内容	教师活动	学生活动	教学方法与手段
分组实施 35分钟	<p>案例 1: 给定三角形高和底边长, 输出三角形面积</p> <pre>h=input("请输入三角形的高:") l=input("请输入三角形的底边长:") s=eval(h)*eval(l)/2 print("三角形面积是:",s)</pre> <p>课程思政: 爱国主义教育教育 我国著名的数学家秦九韶在《数书九章》提出了“三斜求积术”, 它填补了中国数学史中的一个空白。</p> <p>案例 2: 模拟超市收银抹零行为</p> <p>教师根据教材带领学生完成实例 2: 模拟超市收银抹零行为, 讲解该实例中数字类型的使用。</p> <p>案例 3: 敏感词替换</p> <p>敏感词是指带有敏感政治倾向、暴力倾向、不健康色彩的词或不文明的词语, 大部分网站、论坛、社交软件都会使用敏感词过滤系统。</p>  <p>案例核心代码:</p> <pre>sensitive_character = '你好' # 敏感词库 test_sentence = input('请输入一段话:') for line in sensitive_character: # 遍历输入的字符是否存在敏感词库中 if line in test_sentence: # 判断是否包含敏感词 test_sentence = test_sentence.replace(line, '*') print(test_sentence)</pre> <p>课程思政: 通过敏感词替换案例, 引导学生树立社会主义荣辱观, 从自身做起, 在以积极态度促进互联网健康发展, 承担起应负的社会责任, 提倡正确导向, 反对不良网风, 坚持文明上网。</p>	<ol style="list-style-type: none"> 1. 巡回指导, 解疑答难; 2. 播放视频“三斜求积术”, 培养学生民族自豪感; 3. 容易出现的问题: TypeError: unsupported operand type(s) for /: 'str' and 'str' 解决: /运算的两个操作数要求不能是字符串, 需要将输入的字符串转换为 float 类型 	<ol style="list-style-type: none"> 1. 观看视频“三斜求积术”, 思考我国古代在数学上的成就, 分组讨论完成案例 1; 1. 分组讨论完成案例 2-3 	<ol style="list-style-type: none"> 1. 运用多媒体课件和视频; 2. 利用 API 文档; 3. PyCharm 软件、Python3.7 4. 分组完成程序
小组汇报与总结 10分钟	各小组汇报代码编写及运行调试情况 教师总结	<ol style="list-style-type: none"> 1. 点评小组作品 2. 总结内容 	各小组汇报展示、各小组互相评分	<ol style="list-style-type: none"> 1. PyCharm 软件 2. 多媒体课件

教学环节设计

三、课后拓展

环节 用时	内容	教师活动	学生活动	教学方法 与手段
完善课 上作品 15 分钟	项目组根据老师的点评，继续完善课上的案例，将作品上传到教学平台。	发布作业 批阅作业 在线辅导	完成作业 上传平台 巩固知识	1. 利用教学平台作业分析功能； 2. PyCharm 软件编写代码
学习网 络资源 10 分钟	教师推荐优秀的网络学习资源，如：国家数字化学习资源中心、慕课网等，拓宽学生视野。	推荐优秀 的网络学习 资源	根据自己需 要学习网络 资源	利用网络 资源拓宽 学生视野
拓展 作业 15 分钟	思考：计算三角形面积还有哪些方法，运用“三斜求积术”输出三角形面积；	发布作业 解答问题	查找资料 编写代码 运行调试 上传平台	1. 利用模拟实训平台提交代码； 2. 小组讨论完成作业
大 数 据 开 发 Python、 计 算 机 视 觉 1+x 证书 20 分钟	学习《大数据开发（Python）》、《计算机视觉》1+x 证书考试相关资料 项目二 数据文件整理	提供《大数据 开 发 （Python）》、 《计算机视 觉》1+x 证书 考试相关资 料、辅导常见 问题	学习提供 《大数据开 发 （Python）》 、《计算机视 觉》1+x 证 书考试相关 资料	1. 利用教 学平台； 2. 网络资 源提供学 习内容

任务一

(定义数字类型变量)

一、创设情景，引出数字类型变量

数字类型和字符串是 Python 程序中基本的数据类型，其中数字类型分为整型、浮点型、复数类型、布尔类型，可通过运算符进行各种数学运算。

1. 教师通过提出需求引出 Python 中的数字类型

教师根据数学中的数字类型引出 Python 中的数字类型。

2. 明确学习目标

- (1) 要求学生了解数字类型的表示方法
- (2) 要求学生掌握类型转换函数

二、重点知识讲解

1. Python 中的数字类型及其表示方法

表示数字或数值的数据类型称为数字类型。Python 内置的数字类型有整型 (int)、浮点型 (float)、复数类型 (complex)、布尔类型 (bool)。

(1) 整型：Python 中使用 4 种进制表示整型，分别为二进制、八进制、十进制和十六进制。

案例：

```
a=0b10101
print(type(a))
```

案例：

```
print(bin(20))    #将十进制的 20 转换为二进制数
print(oct(20))   #将十进制的 20 转换为八进制数
print(hex(20))   #将十进制的 20 转换为十六进制数
```

(2) 浮点数一般以十进制形式表示，对于较大或较小的浮点数，可以使用科学计数法表示。

```
num_one = 3.14      # 十进制形式表示
num_two = 2e2       # 科学计数法表示
num_third = 2e-2    # 科学计数法表示
```

案例：print(1.25E5)

(3) 复数类型

Python 中有两种创建复数的方式，一种是按照复数的一般形式直接创建，另一种是通过内置函数 complex() 创建。

```
num_one = 3 + 2j    # 直接创建
num_two = complex(3, 2) # 通过内置函数创建
```

案例：

```
a=2+3.5j
```

```
print(a.real)    #实数部分
```

```
print(a.imag)   #虚数部分
```

(4) Python 中的布尔类型 (bool) 只有两个取值: True 和 False。

Python 中的任何对象都可以转换为布尔类型, 若要进行转换, 符合以下条件的数据都会被转换为 False:

None;

任何为 0 的数字类型, 如 0、0.0、0j;

任何空序列, 如 ""、()、[];

任何空字典, 如 {};

2. 教师根据课件讲解实例 1: 根据身高体重计算 BMI 指数

教师根据教材带领学生完成实例 1: 根据身高体重计算 BMI 指数, 讲解该实例中数字类型的使用。

3. 教师根据课件讲解类型转化函数

Python 内置了一系列可实现强制类型转换的函数, 保证用户在有需求的情况下, 将目标数据转换为指定的类型。数字间进行转换的函数有 int()、float()、str()。

函数	说明
int()	将浮点型、布尔类型和符合数值类型规范的字符串转换为整型
float()	将整型和符合数值类型规范的字符串转换为浮点型
str()	将数值类型转换为字符串

案例:

```
a=1.2
```

```
print(int(a))          #将 a 转换为整数
```

```
b=2
```

```
print(float(b))       #将 b 转换为浮点数
```

```
print(complex(a, b))  #创建一个复数
```

4. 教师引导学生思考

Java 定义变量与 Python 定义变量的区别与联系, 让学生理解事物的联系普遍存在的, 引导学生用类比的方法进行知识的迁移。

观看视频《如何利用知识迁移能力快速进入一个领域》, 让学生明白知识迁移的重要性, 培养学生知识迁移的能力。

三、归纳总结, 布置作业

1. 回顾上课前的学习目标, 对本节课知识点进行总结。

带领学生回顾整型、浮点型、复数类型数据的定义与使用。

2. 布置随堂练习, 检查学生掌握情况。

根据随堂练习资源, 给学生布置随堂练习, 检测学生的掌握程度。

任务二

(定义字符串、字符串处理函数、字符串处理方法)

一、回顾任务一内容，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。

任务一介绍了数据类型，本次任务介绍定义字符串的方法、字符串处理函数、字符串处理方法。

3. 明确学习目标

- (1) 掌握字符串的定义与格式化
- (2) 掌握字符串处理函数
- (3) 掌握字符串处理方法
- (4) 掌握实例过滤敏感词

二、进行重点知识的讲解

1. 字符串的定义

字符串是一种用来表示文本的数据类型，它是由符号或者数值组成的一个连续序列，Python 中的字符串是不可变的，字符串一旦创建便不可修改。Python 支持使用单引号、双引号和三引号定义字符串，其中单引号和双引号通常用于定义单行字符串，三引号通常用于定义多行字符串，具体示例分别如下。

2. 使用格式符%对字符串格式化

案例

```
user_name=input("请输入用户名：")
print("用户名为%s"%user_name)
user_pass=input("请输入密码：")
print("密码为%s"%user_pass)
```

3. 字符串操作符

案例

```
a="hello"
b="python"
print(a+b)
print(a*2)
print(a==b)
print(a!=b)
print(a in b)
```

4. 使用切片截取字符串

Python 中字符串的索引从 0 开始计数，并且字符串中每个元素都有对应的索引值，索引又分为正向索引与逆向索引，正向索引为字符编号从左到右由 0 开始递增，

逆向索引为字符编号从右到做从-1 开始递减。

案例：

```
s="Alphonse Elric"  
print(s[-9])  
print(s[3])
```

切片指的是通过元素的索引值进行访问。

5. 字符串处理函数

Python 中提供了很多内置函数,使用这些内置函数可以便捷地对字符串执行一些常见的操作,例如计算字符串长度的 len() 函数、返回单字符 Unicode 编码的 ord() 函数。

案例：

```
a="hello"  
print(type(a))    #type()是python查看数据类型的函数  
print(ord('a'))
```

6. 字符串处理方法

Python 提供了许多字符串操作方法,常用的操作字符串方法可以分为大小写转换 (lower() 与 upper)、判断 (具体函数详见表 3-15)、填充字符串 (ljust()、center()、rjust())、字符串搜索 (find()、index())、替换 (replace())、分隔 (split()) 等。

(1) 大小写转换

lower(): 转换为小写

upper(): 转换为大写

(2) 判断字符串中字符的类型

isdecimal(): 如果字符串中只包含十进制数字则返回 True, 否则返回 False

isdigit(): 如果字符串中只包含数字则返回 True, 否则返回 False

isnumeric(): 如果字符串中只包含数字则返回 True, 否则返回 False

isalpha(): 如果字符串中至少有一个字符, 并且所有字符都是字母则返回 True, 否则返回 False

isalnum(): 如果字符串中至少有一个字符, 并且所有字符都是字母或数字则返回 True, 否则返回 False

(3) 填充字符串

ljust(width, fillchar=None): 使用字符 fillchar 以右对齐方式填充字符串, 使其长度变为 width;

center(width, fillchar=None): 使用字符 fillchar 以居中对齐方式填充字符串, 使其长度变为 width;

rjust(width, fillchar=None): 使用字符 fillchar 以左对齐方式填充字符串, 使其长度变为 width。

教
学
内
容

```
j="Hello Python"  
print(j.ljust(50,'#'))  
print(j.rjust(50,"*"))  
print(j.center(50))
```

(4) 从字符串中搜索子串

```
find(sub, start=None, end=None)  
index(sub, start=None, end=None)  
j="Hello Python"  
print(j.find("ell", 0, 4))  
print(j.find("ell"))  
print(j.index("ell"))  
print(j.index("ell", 0, 5))
```

(5) 替换字符串

```
replace(old, new, count=None)  
print(j.replace("Python", "Java", 1))
```

(6) 分隔字符串

```
split(sep=None, maxsplit=-1)  
f="this is string example...wow!!!"  
print(f.split())  
print(f.split('i', 1))  
print(f.split('w'))
```

运行结果:

```
['this', 'is', 'string', 'example...wow!!!']  
['th', 's is string example...wow!!!']  
['this is string example...', 'o', '!!!']
```

(8) 删除字符串中指定字符

```
lstrip([chars]) 删除字符串左边的空格或指定字符  
rstrip([chars]) 删除字符串末尾的指定字符  
strip([chars]) 删除字符串头尾的指定字符  
f=" this is string example...wow!!! "  
print(f)  
print(f.lstrip())  
print(f.rstrip())  
print(f.strip())
```

三、案例：敏感词替换

敏感词是指带有敏感政治倾向、暴力倾向、不健康色彩的词或不文明的词语，

大部分网站、论坛、社交软件都会使用敏感词过滤系统。



案例核心代码：

```
sensitive_character = '你好' # 敏感词库
test_sentence = input('请输入一段话:')
for line in sensitive_character: # 遍历输入的字符是存在敏感词库中
    if line in test_sentence: # 判断是否包含敏感词
        test_sentence = test_sentence.replace(line, '*')
print(test_sentence)
```

课程思政：

通过敏感词替换案例，引导学生树立社会主义荣辱观，从自身做起，在以积极态度促进互联网健康发展，承担起应负的社会责任，提倡正确导向，反对不良网风，坚持文明上网。

四、归纳总结，布置作业

1. 回顾上课前的学习目标，对本节课知识点进行总结。

带领学生回顾字符串的定义、切片截取字符串、字符串处理函数、字符串常用方法。

2. 布置随堂练习，检查学生掌握情况。

根据随堂练习资源，给学生布置随堂练习，检测学生的掌握程度。

任务三 (运用运算符)

一、回顾上节课内容，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。
3. 明确学习目标
 - (1) 了解数字类型转换
 - (2) 掌握 Python 中数值运算符的使用
 - (3) 掌握运算符优先级
 - (4) 掌握实例模拟商家收银抹零行为的实现

二、进行重点知识的讲解

1. 教师提问：Java 中的运算符有哪些？

课程思政：培养学生不同编程语言的融会贯通能力。

2. Python 中的数值运算符

Python 中的数值运算符可大致分为 6 类，分别为算术运算符、赋值运算符、比较运算符、逻辑运算符、成员运算符、身份运算符。

(1) 算术运算符：+、-、*、/、//、%、**

案例：

```
a=20
```

```
b=10
```

```
c=0
```

```
#加法运算
```

```
c=a+b
```

```
print("1--c 的值为",c)
```

```
#减法运算
```

```
c=a-b
```

```
print("2--c 的值为",c)
```

```
#乘法运算
```

```
c=a*b
```

```
print("3--c 的值为",c)
```

```
#除法运算
```

```
c=a/b
```

```
print("4--c 的值为",c)
```

```
#取余运算
```

```
c=a%b
```

```
print("5--c 的值为",c)
```

教
学
内
容

```
#取整除运算
c=a//b
print("6--c 的值为",c)
#修改变量 a、b、c
a=2
b=3
c=a**b    #幂的运算
print("7--c 的值为",c)
(2) 赋值运算符: +=、-=、*=、/=、//=、%=、**=
案例:
a=20
b=10
c=0
#加法赋值运算
c+=a
print("1--c 的值为",c)
#减法赋值运算
c-=a
print("2--c 的值为",c)
#乘法赋值运算
c*=3
c*=a
print("3--c 的值为",c)
#除法赋值运算
c/=a
print("4--c 的值为",c)
#取余赋值运算
c%=a
print("5--c 的值为",c)
#取整除赋值运算
c//=a
print("6--c 的值为",c)
#修改变量 a、b、c
c=2
b=3
c**=b    #幂的赋值运算
```

```
print("7--c 的值为", c)
```

(3) 比较运算符: ==、!=、>、<、>=、<=

案例:

```
a=20
```

```
b=10
```

```
c=0
```

#比较 a 和 b 的值是否相等

```
print(a==b)
```

```
if(a==b):
```

```
    print("1--a 等于 b")
```

```
else:
```

```
    print("1--a 不等于 b")
```

#比较 a 是否大于 b

```
print(a>b)
```

```
if(a>b):
```

```
    print("2--a 大于 b")
```

```
else:
```

```
    print("2--a 不大于 b")
```

(4) 逻辑运算符: “or” (或)、 “and” (与)、 “not” (非)

案例:

```
a=10
```

```
b=20
```

```
print(a and b)
```

```
print(a or b)
```

```
print(not a)
```

```
print(a-10 and b)
```

```
print(a-10 or b)
```

运行结果:

```
20 10 False 0 20
```

(5) 成员运算符: in 与 not in

案例:

```
a=10
```

```
b=5
```

```
list_demo=[1, 2, 3, 4, 5, 6, 7]
```

```
print(a in list_demo)
```

```
print(a not in list_demo)
```

```
print(b in list_demo)
print(b not in list_demo)
```

(6) 位运算符

案例:

```
a=8
b=9
print(bin(a<<2))
print(bin(b>>2))
print(bin(a&b))
print(bin(a|b))
print(bin(a^b))
print(bin(~b))
```

9 的二进制数为 00001001，正数的原码=反码=补码，真正存储的时候，存储的是 00001001，对 9 的补码进行取反得 11110110，这是补码。将补码转换为原码，符号位不变，其他位取反，然后加 1，得源码，为 10001010。

3. Python 中运算符优先级

Python 中支持使用多个不同的运算符连接简单表达式，实现相对复杂的功能，为了避免含有多个运算符的表达式出现歧义，Python 为每种运算符都设定了优先级（具体详见教材表 3-5）。

案例:

```
a=20 b=10 c=15 d=5 e=0
e=(a+b)*c/d
print("(a+b)*c/d 运算结果为", e)
e=((a+b)*c)/d
print("((a+b)*c)/d 运算结果为", e)
e=(a+b)*(c/d)
print("(a+b)*(c/d)", e)
e=a+b*c/d
print("e=a+b*c/d", e)
```

4. 案例：给定三角形高和底边长，输出三角形面积

```
h=input("请输入三角形的高：")
l=input("请输入三角形的底边长：")
s=eval(h)*eval(l)/2
print("三角形面积是：", s)
```

教 学 内 容 总 结	三、归纳总结，布置作业	
	回顾上课前的学习目标，对本节课知识点进行总结。	
	带领学生回顾算术运算符、逻辑运算符、赋值运算符、比较运算符、位运算符的使用。	
	任务	知识目标
任务一 定义数字类型变量	(1) 要求学生了解数字类型的表示方法 (2) 要求学生掌握类型转换函数	1. Java 定义变量与 Python 定义变量的区别与联系, 让学生理解事物的联系普遍存在的, 引导学生用类比的方法进行知识的迁移。 2. 观看视频《如何利用知识迁移能力快速进入一个领域》, 让学生明白知识迁移的重要性, 培养学生知识迁移的能力。
任务二 定义与操作字符串	(1) 掌握字符串的定义与格式化 (2) 掌握字符串处理函数 (3) 掌握字符串处理方法 (4) 掌握实例过滤敏感词	通过敏感词替换案例, 引导学生树立社会主义荣辱观, 从自身做起, 在以积极态度促进互联网健康发展, 承担起应负的社会责任, 提倡正确导向, 反对不良网风, 坚持文明上网。
任务三 运用运算符	(1) 了解数字类型转换 (2) 掌握 Python 中数值运算符的使用 (3) 掌握运算符优先级 (4) 掌握实例模拟商家收银抹零行为的实现	通过比较 java 中运算符的用法, 培养学生对不同编程语言的融会贯通能力。
教 学 反 思	经验	1. 多门课程联动, 深入挖掘思政元素 2. 整合教学案例, 有效融入课程思政 3. 融合多种教学手段和方法, 高效完成教学目标
	不足	1. 教师课程设计能力及课程思政能力有待进一步提高 2. 课程教学材料及思政资料需要进一步完善
拓 展 作 业	1. 运用“三斜求积术”输出三角形面积; 2. 课程平台拓展作业。	