

教案七 运用组合数据类型

单元标题	运用组合数据类型		单元教学学时	8 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. 组合数据类型的分类 2. 操作序列类型的数据 3. 操作集合数据 4. 操作元组数据 			
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。学生已经掌握了流程控制语句、模块与包的用法、函数等，能够进行简单的编程。通过课前测试大数据分析，学生对 Python 中的组合数据类型的理解不够透彻；通过问卷调查，65% 的学生有过考试作弊的行为，89% 的同学欺骗过家人朋友。</p>			
教学 目标	思政目标	知识目标	能力目标	
	<ol style="list-style-type: none"> 1. 培养学生良好的诚实守信的道德品质 2. 引导学生做人要诚信：做到学业诚信不作弊；贷款诚信不逾期；做事诚信不违约。 	<ol style="list-style-type: none"> 1. 了解组合数据类型分类 2. 掌握序列类型的特点，熟练操作列表和元组 3. 了解集合类型的特点，熟悉集合的基础操作 4. 掌握映射类型的特点，可以熟练地操作字典 	<ol style="list-style-type: none"> 1. 能够熟练操作列表和元组数据类型 2. 能够熟练操作集合和字典数据类型 	
课程思政	融入 知识点	通过学生成绩管理系统引出诚信的含义		
	融入方式	通过案例融入思政		
	思政元素	诚信是一种人们在立身处世、待人接物和生活实践中必须而且应当具有的真实无欺、实事求是的态度和信守诺言的行为品质，其基本要求是说老实话、办老实事、做老实人。		

	<p>思政资源</p> <p>视频：什么是诚信</p> <p>视频：如何做到诚信</p> <p>视频：诚信考试、杜绝作弊</p>										
教学重点、难点	<p>教学重点：列表和元组的常见操作、字典的常见操作</p> <p>教学难点：集合关系测试的操作</p>										
教学方法与教学手段	<p>1. 教学方法：</p> <p>(1) 课前让学生学习微课、互联网查找资料熟悉知识点；通过在线测试了解学生对知识点的掌握情况，调整教学策略；</p> <p>(2) 采用教学做一体的教学模式，运用翻转课堂模式，以任务驱动为载体，将公司管理机制运用到教学管理中；</p> <p>(3) 理论联系实际，引入“好友信息管理系统”、“学生成绩管理系统”、“生词本”等案例，充分调动学生的积极性和主动性，夯实学生的基础知识，培养学生探究性学习的能力；</p> <p>(4) 分组讨论“如何实现列表元素的更新操作”，小组代表汇报；</p> <p>(5) 实例讲解列表、字典、集合、元组的定义与操作</p> <p>2. 辅助手段：</p> <p>(1) 多媒体演示、原理动画展示；</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>(4) 利用编程软件 Python3.8、PyCharm 编写调试代码。</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、分析流程图等解决和突破。</p>										
课前需掌握内容	组合数据类型的分类；键值对形式的用法										
教学内容设计	<p>任务一 运用列表管理好友信息</p> <p>任务二 使用字典类型实现学生成绩管理系统</p> <p>任务三 使用集合实现生词本</p>										
教学资源	<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>65</td> </tr> <tr> <td>思维导图/流程图</td> <td>6</td> </tr> <tr> <td>案例源码/推荐学习内容</td> <td>4</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	65	思维导图/流程图	6	案例源码/推荐学习内容	4
	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	65									
	思维导图/流程图	6									
案例源码/推荐学习内容	4										
课后拓展作业	运用组合数据类型实现手机通讯录										

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《定义与访问列表》、《操作列表》相关微课	1. 将微课、课件、教案上传到教学平台； 2. 发布预习通知	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. 下列方法中，可以获取字典中所有键的是()</p> <p>A. keys() B. value()</p> <p>C. list() D. values()</p> <p>2. 下列方法中，不能删除字典中元素的是()。</p> <p>A. clear() B. remove()</p> <p>C. pop() D. popitem()</p> <p>3. 阅读下面程序： <pre>set_01={'a', 'c', 'b', 'a'}</pre> <pre>set_01.add('d')</pre> <pre>print(len(set_01))</pre> 运行程序，以下输出结果正确的是()。</p> <p>A. 5 B. 3</p> <p>C. 4 D. 2</p> <p>4. 下列方法中，可以对列表元素排列的是()。</p> <p>A. sort() B. reverse()</p> <p>C. max() D. list()</p> <p>5. 下列创建元组的语句中，正确的是()。</p> <p>A. tu_one=tuple('1', '2')</p> <p>B. tu_two=(&apos;q)</p> <p>C. tu_three=('on',)</p> <p>D. tu_four=tuple(3, 5)</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	查找资料，了解 Python 中组合数据类型的分类和各类型数据的特点。	1. 发布课前作业； 2. 查看学生作业、了解学生对知识点运用情况	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（8 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	<p>1. 教师通过提出需求引出组合数据类型</p> <p>教师提前准备一些数据，例如通讯录（包含姓名与电话）和数列，并由这些数据产生需求：通过哪种数据结构来表示这些数据？教师根据提出的需求进一步阐述解决方法：前面所讲的基本数据类型只能表示单一的数据，无法表示一组数据，因此 Python 又提供了组合数据类型。</p> <p>2. 组合数据类型有哪些好处</p> <p>组合数据类型可以同时处理一组数据，这样不仅简化了程序员的开发工作，而且大大地提高了程序的效率。</p>	<p>1. 提出问题；</p> <p>2. 启发引导学生思考；</p> <p>3. 引出学习目的及重点、难点</p>	<p>结合生活实际，积极思考踊跃回答</p>	<p>1. 多媒体课件；</p> <p>2. 教学平台；</p> <p>3. PyCharm 软件</p>
查看课 前预习 情况 10 分钟	<p>1. 小组展示课前作业，教师评价</p> <p>2. 展示课前测试情况</p>	<p>1. 教师评价；</p> <p>2. 讲解错误率高的题目</p>	<p>1. 小组展示作业情况</p> <p>2. 改正错误</p>	<p>1. 利用教学平台课前分析测试情况；</p> <p>2. 多媒体课件；</p>
分析 问题 10 分钟	<p>1. 播放视频：诚信考试</p> <p>2. Python 的组合数据类型可分成三类：序列类型、集合类型和映射类型。其中，序列类型存储一组排列有序的元素；集合同样存储一组数据，它要求其数据必须唯一，但不要求数据间有序；映射类型存储的每个元素都是一个键值对，通过键值对的键可以迅速获得对应的值，数据类型是字典。</p> <p>3. 明确学习目标</p> <p style="padding-left: 20px;">（1）要求学生熟悉组合数据类型的分类</p> <p style="padding-left: 20px;">（2）要求学生掌握列表的定义</p> <p style="padding-left: 20px;">（3）要求学生掌握列表的基本用法</p>	<p>1. 播放视频，引导学生做人要诚信；</p> <p>2. 运行学生成绩管理系统，引出组合数据类型</p>	<p>1. 观看运行效果；</p> <p>2. 小组代表发言</p>	<p>1. 多媒体课件；</p> <p>2. 运用 PyCharm 软件运行效果</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>任务一 运用列表管理好友信息</p> <p>1. 创建列表</p> <p>使用 [] 创建列表</p> <pre>list_one = [] list_two = [1, 10, 55, 20, 6] list_thr = [10, 'word', True, [6, 1]]</pre> <p>使用 list() 创建列表, 或将元组或字符串转换为列表</p> <pre>list_one = list() words = 'Python' tuple_demo = (1, 3, 5, 7, 9) print(list(tuple_demo))</pre> <p>2. 列表的循环遍历</p> <p>for 循环因子 in 列表: 语句块</p> <pre>name_list=['小明','小红','小华'] for name in name_list: print(name) while 变量<列表长度: 语句块 变量+1</pre> <pre>name_list = ['小明', '小红', '小华'] i=0 while i<len(name_list): print(name_list[i]) i+=1</pre> <p>3. 列表的常见操作</p> <table border="1"> <thead> <tr> <th>常见操作</th> <th>常见操作</th> </tr> </thead> <tbody> <tr> <td>len(s)</td> <td>list.insert()</td> </tr> <tr> <td>min(s)</td> <td>List.pop()</td> </tr> <tr> <td>max(s)</td> <td>List.remove()</td> </tr> <tr> <td>list.append()</td> <td>list.reverse()</td> </tr> <tr> <td>list.sort()</td> <td>list.clear()</td> </tr> </tbody> </table>	常见操作	常见操作	len(s)	list.insert()	min(s)	List.pop()	max(s)	List.remove()	list.append()	list.reverse()	list.sort()	list.clear()	<p>1.讲授知识 点;</p> <p>2.提出问题: 列表可以存 放的数据类 型有哪些?</p> <p>3.发起讨论 话题: 列表 常用的操作 有哪些?</p>	<p>1.认真听讲</p> <p>2.思考问题</p> <p>3.小组讨论</p> <p>4.提交讨论 结果;</p> <p>5.学习 API 中关于列 表的常用 方法</p>	<p>1.教学 平台;</p> <p>2.多媒体 课件;</p> <p>3.PyCharm 软件;</p> <p>4.Python3. 7 编辑软 件;</p> <p>5. 分组讨 论;</p>
常见操作	常见操作															
len(s)	list.insert()															
min(s)	List.pop()															
max(s)	List.remove()															
list.append()	list.reverse()															
list.sort()	list.clear()															

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;">任务二 使用字典实现学生成绩管理系统</p> <p>1.创建字典元素</p> <p>字典的创建：Python 中使用“{}”包含键值对以创建字典，字典中各个元素之间通过逗号分隔，语法格式如下：</p> <p style="text-align: center;">{键 1:值 1, 键 2:值 2,…… 键 N:值 N}</p> <div style="text-align: center;">  </div> <p>2.字典元素的添加和修改</p> <p>字典支持使用 <code>update()</code> 方法或通过指定的键添加元素或修改元素；可以通过字典中的键对元素进行修改。</p> <p>3.字典元素的删除</p> <p>删除字典中元素的方法有 <code>pop()</code>、<code>popitem()</code>、<code>clear()</code>，其中 <code>pop()</code> 方法可根据指定键值删除字典中的指定元素，若删除成功，该方法返回目标元素的值；<code>popitem()</code> 方法可以随机删除字典中的元素；<code>clear()</code> 方法用于清空字典中的元素。</p> <p>4.字典元素的查询</p> <p>字典中 <code>items()</code> 方法会返回一个 <code>dict_items</code> 对象，该对象支持迭代操作，通过 <code>for</code> 循环遍历 <code>dict_items</code> 对象中的数据并以 <code>(key, value)</code> 的形式显示；<code>keys()</code> 方法可以查看字典中所有的键；<code>values()</code> 方法返回字典中所有的值。</p> <p style="color: red;">课程思政：</p> <p style="color: red;">1. 诚信是一种人们在立身处世、待人接物和生活实践中必须而且应当具有的真实无欺、实事求是的态度和信守诺言的行为品质，其基本要求是说老实话、办老实事、做老实人。</p> <p style="color: red;">2. 引导学生做人要诚信：做到学业诚信不作弊；贷款诚信不逾期；做事诚信不违约。</p>	<ol style="list-style-type: none"> 1. 讲授知识点； 2. 提出问题：字典类型与列表类型有哪些区别； 3. 发起讨论话题：Java 中什么类型可以以键-值对的形式存储数据 	<ol style="list-style-type: none"> 1. 认真听讲 2. 思考问题 3. 小组讨论 4. 提交讨论结果 	<ol style="list-style-type: none"> 1.教学平台； 2.多媒体课件； 3.PyCharm 软件； 4.Python3.7 编辑软件； 5.分组讨论；

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 align="center">任务三 使用集合实现生词本</p> <p>1. 什么是集合</p> <p>Python 中集合可分为可变集合与不可变集合, 可变集合由 <code>set()</code> 函数创建, 集合中的元素可以动态地增加或删除; 不可变集合由 <code>frozenset()</code> 函数创建, 集合中的元素不可改变。</p> <p>2. 集合的创建方式</p> <p>列表可以存储任何元素, 当然也可以存储列表, 如果列表存储的元素也是列表, 则称为嵌套列表。</p> <p>可以使用大括号 <code>{ }</code> 或者 <code>set()</code> 函数创建集合, 注意: 创建一个空集合必须用 <code>set()</code> 而不是 <code>{ }</code>, 因为 <code>{ }</code> 是用来创建一个空字典。</p> <p>创建格式:</p> <pre>parame = {value01,value02,...}</pre> <p>或者 <code>set(value)</code></p> <p>3. 集合的内置方法</p> <p>Python 中可变集合支持添加、删除和情况元素, 可变集合的 <code>add()</code> 方法或 <code>update()</code> 方法都可以实现向集合中添加元素, 不同的是, <code>add()</code> 方法只能添加一个元素, 而 <code>update()</code> 方法可以添加多个元素; <code>remove()</code>、<code>discard()</code>、<code>pop()</code> 方法删除可变集合中的元素; <code>clear()</code> 方法可以清空集合中的元素。</p> <table border="1"> <thead> <tr> <th>常见操作</th> <th>常见操作</th> </tr> </thead> <tbody> <tr> <td><code>add()</code></td> <td><code>union()</code></td> </tr> <tr> <td><code>clear()</code></td> <td><code>update()</code></td> </tr> <tr> <td><code>copy()</code></td> <td><code>difference()</code></td> </tr> <tr> <td><code>pop()</code></td> <td><code>discard()</code></td> </tr> <tr> <td><code>remove()</code></td> <td><code>intersection()</code></td> </tr> </tbody> </table> <p>4. 集合类型的操作符</p> <p>Python 支持通过操作符 <code> </code>、<code>&</code>、<code>-</code>、<code>^</code> 对集合进行联合、取交集、差补和对称差分操作。</p>	常见操作	常见操作	<code>add()</code>	<code>union()</code>	<code>clear()</code>	<code>update()</code>	<code>copy()</code>	<code>difference()</code>	<code>pop()</code>	<code>discard()</code>	<code>remove()</code>	<code>intersection()</code>	<p>1. 讲授知识点</p> <p>2. 提出问题: 集合类型、字典类型和列表类型有哪些区别;</p> <p>3. 总结回答情况</p>	<p>1. 认真听讲</p> <p>2. 思考问题</p> <p>3. 小组讨论</p> <p>4. 提交讨论结果</p>	<p>1. 教学平台;</p> <p>2. 多媒体课件;</p> <p>3. PyCharm 软件;</p> <p>4. Python3.7 编辑软件;</p> <p>5. 分组讨论;</p>
常见操作	常见操作															
<code>add()</code>	<code>union()</code>															
<code>clear()</code>	<code>update()</code>															
<code>copy()</code>	<code>difference()</code>															
<code>pop()</code>	<code>discard()</code>															
<code>remove()</code>	<code>intersection()</code>															

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 60 分钟	<p style="text-align: center;">任务一 运用列表管理好友信息</p> <p>实施步骤:</p> <ol style="list-style-type: none"> 定义列表 name_list1 = ['小明', '小红', '小华'] 使用 append 方法向列表中添加元素 temp_name=input("请输入要添加的姓名:") name_list1.append(temp_name) 使用 extend 方法向列表中添加元素 name_list2=['小雪','小云'] name_list1.extend(name_list2) 查找元素 find_name=input("请输入要查找的姓名:") if find_name in name_list1: print("在列表中找到了相同的名字") else: print("没有找到") 修改元素的值 modify_id=int(input("请输入要修改的学生编号:")) modify_name=input("请输入要修改学生的姓名:") name_list1[modify_id]=modify_name 删除元素 #使用 del 删除指定下标的列表元素 del_id=int(input("请输入要删除学生的编号")) del name_list1[del_id] #使用 pop 删除列表中最后一个元素 name_list1.pop() #使用 remove 删除指定的元素 del_name=input("请输入要删除学生的姓名:") if del_name in name_list1: name_list1.remove(del_name) #列表的排序操作 name_list1.reverse(); name_list1.sort() 	<ol style="list-style-type: none"> 讲述要求及步骤; 巡回指导; 解疑答疑常见问题: name_list1.extend(name_list2) 	<ol style="list-style-type: none"> 思考实施步骤; 学习 API 文档中的方法; 小组为单位进行编写代码、运行测试 	<ol style="list-style-type: none"> API 文档; 网络资源; 教学平台; 多媒体课件; PyCharm 软件; Python3 .7

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 60 分钟	<p>任务二 使用字典实现学生成绩管理系统</p> <p>实施步骤:</p> <ol style="list-style-type: none"> 1. 创建字典类型 <pre>person_info = []</pre> 2. 输出主界面 3. 判断用户输入的功能键 <ol style="list-style-type: none"> (1) 添加信息 <pre>per_dict.update({'姓名': per_name, '年龄': phone_num, '班级': per_class, '科目': per_course, '成绩': per_grade },) person_info.append(per_dict)</pre> (2) 查询信息 <pre>for title, info in i.items(): print(title + ':' + info)</pre> (3) 删除信息 <pre>del_name = input('输入删除的联系人姓名: ') for i in person_info: if del_name in i.values(): person_info.remove(i)</pre> (4) 修改信息 <pre>modi_info = input('请输入要修改的联系人姓名: ') for i in person_info: if modi_info in i.values(): # 获取所在元组在列表中的索引位置 index_num = person_info.index(i) dict_cur_perinfo = person_info[index_num] for title, info in dict_cur_perinfo.items(): per_name = input('请输入学生的姓名: ') dict_cur_perinfo.update(姓名= per_name)</pre> (5) 查找信息 <pre>if query_name in i.values(): index_num = person_info.index(i)</pre> 	<ol style="list-style-type: none"> 1. 讲述要求及步骤; 2. 巡回指导; 3. 解疑答难 常见问题: <pre>person_info.remove(i), dict_cur_perinfo.update(姓名= per_name)</pre> 	<ol style="list-style-type: none"> 1. 思考实现步骤; 2. 学习 API 文档中的方法; 3. 小组为单位进行编写代码、运行测试 	<ol style="list-style-type: none"> 1. API 文档; 2. 网络资源; 3. 教学平台; 4. 多媒体课件; 5. PyCharm 软件; 6. Python3 .7

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 60 分钟	<p align="center">任务三 使用集合实现生词本</p> <p>实施步骤:</p> <ol style="list-style-type: none"> 定义集合 <pre>words_book = set() words_only_set = set()</pre> 查看生词本 <pre>print(words_book)</pre> 背单词 <pre>in_words = input("请输入" + w[0] + ' 翻译' + ': \n') if in_words == w[1].strip(): print('太棒了')</pre> 添加单词 <pre>new_words = input('请输入新单词: ') new_chinese = input('请输入单词翻译: ') word_dict.update({new_words: new_chinese}) # 转换成字符串存入 set 集合 words_book.add(dict_str)</pre> 删除单词 <pre>del_wd = input("请输入要删除的单词") words_only_set.discard(del_wd) for temp in temp_list: if del_wd in temp: ords_book.remove(temp)</pre> 清空 <pre>words_only_set.clear() words_book.clear()</pre> 	<ol style="list-style-type: none"> 讲述要求及步骤; 巡回指导; 解疑答难 <p>常见问题:</p> <pre>for temp in temp_list: if del_wd in temp: ords_book.remove(temp)</pre>	<ol style="list-style-type: none"> 思考实现步骤; 学习 API 文档中的方法; 小组为单位进行编写代码、运行测试 	<ol style="list-style-type: none"> 利用动画演示诺塔游戏过程; 网络资源; 教学平台; 多媒体课件; PyCharm 软件; Python3.7
小组 汇报 15 分钟	各小组汇报代码编写及运行调试情况	点评小组作品	各小组汇报展示、各小组互相评分	PyCharm 软件
课后 拓展 5 分钟	<ol style="list-style-type: none"> 完成课后习题 完善学生成绩管理系统 	<ol style="list-style-type: none"> 引发思考 布置任务 	课后查阅资料, 完成作业后上传课程平台	<ol style="list-style-type: none"> 教学平台 PyCharm 软件

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 作品 15 分钟	项目组根据老师的点评，继续完善作品，上传至教学平台。	批阅作业	完善作品 巩固知识	<ol style="list-style-type: none"> 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 20 分钟	<ol style="list-style-type: none"> 1. 完成课后习题 2. 完善学生成绩管理系统 3. 有余力的同学完成银行管理系统 	发布作业 解答问题	查找资料 编写代码 运行调试 上传平台	<ol style="list-style-type: none"> 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 项目六 图像标注	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	<ol style="list-style-type: none"> 1. 利用教学平台； 2. 网络资源提供学习内容

任务一

（运用列表管理好友信息）

一、 创设情境，导入组合数据类型的概念

1. 教师通过提出需求引出组合数据类型

教师提前准备一些数据，例如通讯录（包含姓名与电话）和数列，并由这些数据产生需求：通过哪种数据结构来表示这些数据？教师根据提出的需求进一步阐述解决方法：前面所讲的基本数据类型只能表示单一的数据，无法表示一组数据，因此 Python 又提供了组合数据类型。

2. 组合数据类型有哪些好处

组合数据类型可以同时处理一组数据，这样不仅简化了程序员的开发工作，而且大大地提高了程序的效率。

3. 明确学习目标

- (1) 要求学生熟悉组合数据类型的分类
- (2) 要求学生掌握列表的定义
- (3) 要求学生掌握列表的基本用法

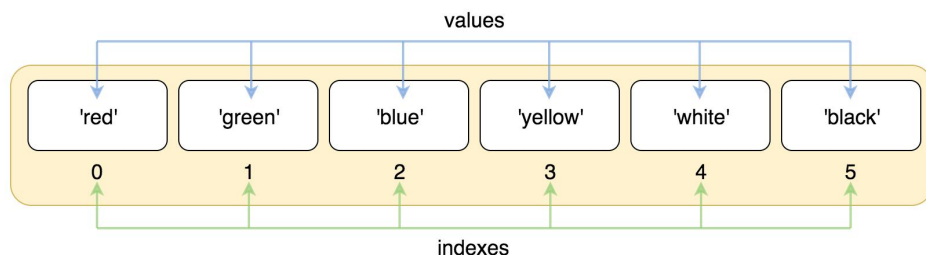
二、 进行重点知识的讲解

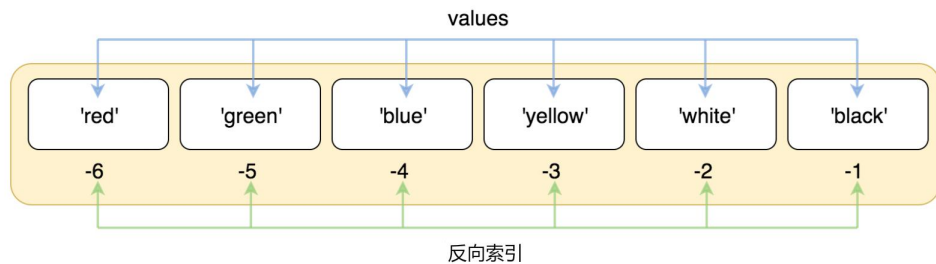
1. 组合数据类型的三个分类

Python 的组合数据类型可分成三类：序列类型、集合类型和映射类型。其中，序列类型存储一组排列有序的元素，每个元素的类型可以不同，通过索引可以锁定序列中的指定元素，数据类型有字符串、元组和列表；集合同样存储一组数据，它要求其中的数据必须唯一，但不要求数据间有序；映射类型存储的每个元素都是一个键值对，通过键值对的键可以迅速获得对应的值，数据类型是字典。

2. 数学中的数列与 Python 中的序列

数列是按一定顺序排成一系列的一组数，每个数称为这个数列的项，每项不是在其它项之前，就是在其它项之后。数列的索引从 0 开始，通过索引 i 可以访问数列中的第 $i-1$ 项。Python 中的序列支持双向索引：正向递增索引和反向递减索引，正向递增索引从左向右依次递增，第一个元素的索引为 0，第二个元素的索引为 1，以此类推；反向递减索引从右向左依次递减，最后一个元素的索引为 -1，倒数第二个元素的索引为 -2，以此类推。





3. 列表类型

(1) 创建列表

使用 [] 创建列表

```
list_one = []
list_two = [1, 10, 55, 20, 6]
list_thr = [10, 'word', True, [6, 1]]
print(list_one)
print(list_two)
print(list_thr[1])
print(list_thr)
```

使用 list() 创建列表，或将元组或字符串转换为列表

```
list_one = list()
words = 'Python'
print(list_one)
print(list(words))
print(list(words)[1])
tuple_demo = (1, 3, 5, 7, 9)
print(list(tuple_demo))
```

(2) 列表的循环遍历

for 循环因子 in 列表:

语句块

```
name_list=['小明','小红','小华']
for name in name_list:
    print(name)
while 变量<列表长度:
    语句块
    变量+1
name_list = ['小明', '小红', '小华']
i=0
while i<len(name_list):
```

```
print(name_list[i])
```

```
i+=1
```

(3) 列表的常见操作

常见操作	说明
len(s)	计算序列 s 的长度（元素个数）
min(s)	返回序列 s 中的最小元素
max(s)	返回序列 s 中的最大元素
list.append	在列表 list 的末尾添加元素 x
list.extend	在列表 list 中添加另一个列表 lx 的元素，与+=功能相同
list.insert	在列表 list 索引为 i 的元素之前插入元素 x
list.pop()	取出并删除列表 list 中索引为 i 的元素 x
List.remove	删除列表 list 中第一次出现的元素 x
list.reverse	将列表 list 的元素反转
list.clear	删除列表 list 中的所有元素
list.copy()	生成新列表，并拷贝列表 list 中的所有元素
list.sort()	将列表 list 中的元素排序

4. 运用列表类型管理好友信息

```
name_list1 = ['小明', '小红', '小华']
```

```
print("-----添加之前，列表 name_list 中的数据-----")
```

```
for temp in name_list1:
```

```
    print(temp)
```

#1. 使用 append 方法向列表中添加元素

```
temp_name=input("请输入要添加的姓名：")
```

```
name_list1.append(temp_name)
```

```
print("-----添加之后，列表 name_list 中的数据-----")
```

```
for temp in name_list1:
```

```
    print(temp)
```

#2. 使用 extend 方法向列表中添加元素

```
name_list2=['小雪', '小云']
```

```
name_list1.extend(name_list2)
```

```
print(name_list1)
```

#3. 查找元素

```
print("-----查找列表 name_list 中的数据-----")
```

```
find_name=input("请输入要查找的姓名：")
```

```
if find_name in name_list1:
```

```
    print("在列表中找到了相同的名字")
```

```

else:
    print("没有找到")
#修改元素的值
print("-----修改列表 name_list 中的数据-----")
print(name_list1)
modify_id=int(input("请输入要修改的学生编号："))
modify_name=input("请输入要修改学生的姓名：")
name_list1[modify_id]=modify_name
print(name_list1)
#删除元素
print("-----删除列表 name_list1 中的数据-----")
#使用 del 删除指定下标的列表元素
del_id=int(input("请输入要删除学生的编号"))
del name_list1[del_id]
print(name_list1)
#使用 pop 删除列表中最后一个元素
name_list1.pop()
print(name_list1)
#使用 remove 删除指定的元素
del_name=input("请输入要删除学生的姓名：")
if del_name in name_list1:
    name_list1.remove(del_name)
print(name_list1)
#列表的排序操作
print("-----列表 name_list1 中的数据排序-----")
name_list1.reverse();
print(name_list1)
name_list1.sort()
print(name_list1)

```

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
带领学生回顾列表的定义与常见的操作。
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

(使用字典实现学生成绩管理系统)

一、回顾任务一的内容，继续讲解本次任务的知识

1. 教师对学生们的疑问进行统一答疑
2. 回顾总结任务一内容，继续介绍本课时的内容

任务一我们学习了列表学习了列表的创建方式、访问列表元素、列表的遍历，本节课将带领大家学习列表的排序、添加列表元素、删除列表元素、修改列表元素。本次任务将带领大家学习字典元素的添加和修改、字典元素的删除、字典元素的查询、运用字典实现手机通讯录。

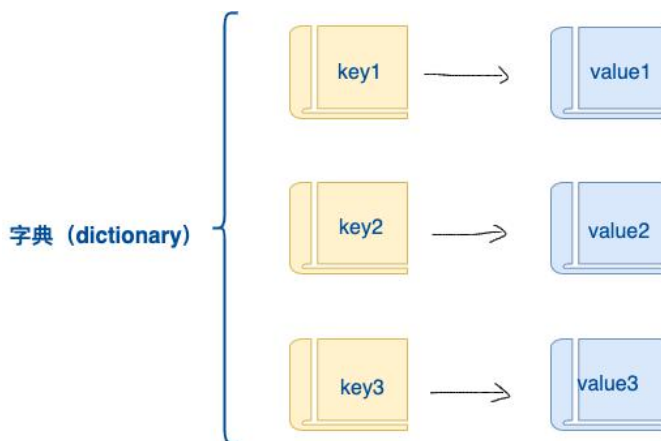
3. 明确学习目标
 - (1) 要求学生掌握字典元素的添加和修改
 - (2) 要求学生掌握字典元素的删除
 - (3) 要求学生掌握字典元素的查询
 - (4) 要求学生了解实例手机通讯录实现过程

二、进行重点知识的讲解

1.创建字典元素

字典的创建：Python 中使用“{}”包含键值对以创建字典，字典中各个元素之间通过逗号分隔，语法格式如下：

{键 1:值 1, 键 2:值 2,..... 键 N:值 N}



2.字典元素的添加和修改

字典支持使用 update()方法或通过指定的键添加元素或修改元素；可以通过字典中的键对元素进行修改。

3.字典元素的删除

删除字典中元素的方法有 pop()、popitem()、clear()，其中 pop()方法可根据指定键值删除字典中的指定元素，若删除成功，该方法返回目标元素的值；popitem()方法可以随机删除字典中的元素；clear()方法用于清空字典中的元素。

4.字典元素的查询

字典中 `items()` 方法会返回一个 `dict_items` 对象，该对象支持迭代操作，通过 `for` 循环遍历 `dict_items` 对象中的数据并以 `(key, value)` 的形式显示；`keys()` 方法可以查看字典中所有的键；`values()` 方法返回字典中所有的值。

案例：

#1. 根据键访问值

```
stu_info={'name':'张三','age':18,'class':'三班','course1':'高等数学','grade1':'90','course2':'英语','grade1':'89'}
```

```
print(stu_info['name'])
```

```
print(stu_info['class'])
```

#2. 修改字典中的元素

```
stu_name=input('请输入学生的姓名')
```

```
stu_info['name']=stu_name
```

```
print('修改后的姓名为：',stu_info['name'])
```

#3. 添加字典元素

```
stu_course=input('请输入学生的科目：')
```

```
stu_info['course3']=stu_course
```

```
print(stu_info)
```

```
stu_grade=input('请输入学生的成绩：')
```

```
stu_info['grade3']=stu_grade
```

```
print(stu_info)
```

#4. 删除字典元素

```
del stu_info['grade']
```

```
print('删除后学生的信息')
```

```
print(stu_info)
```

#5. 计算字典中键值对的个数

```
print('字典的长度',len(stu_info))
```

#6. 获取字典的键视图

```
print('字典的键视图为：')
```

```
print(stu_info.keys())
```

#7. 获取字典的值视图

```
print('字典的值视图为：')
```

```
print(stu_info.values())
```

#8. 获取字典的元素视图

```
print('字典的元素视图为：')
```

```
print(stu_info.items())
```

5.实现学生成绩管理系统

教师根据教材资源实现：学生成绩管理系统，并向学生讲解其实现过程。

```
# 实际案例中，将删除查找联系人功能（index方法没有涉及）
person_info = []
print("=" * 20)
print(' 欢迎使用学生成绩管理系统：')
print("1. 添加学生成绩信息")
print("2. 查看学生成绩信息")
print("3. 删除学生成绩信息")
print("4. 修改学生成绩信息")
print("5. 查找学生成绩信息")
print("6. 退出")
print("=" * 20)
while True:
    per_dict = {}
    fun_num = input(' 请输入功能序号：')
    if fun_num == '1':
        per_name = input(' 请输入学生的姓名：')
        per_age = input(' 请输入学生的年龄：')
        per_class = input(' 请输入学生的班级：')
        per_course = input(' 请输入学生的科目：')
        per_grade = input(' 请输入学生的成绩：')
        # 判断输入的是否为空
        if per_name.strip() == '' or per_age.strip() == '' or
per_class.strip() == '' or per_course.strip() == '':
            print(' 请输入正确信息')
            continue
        else:
            per_dict.update({'姓名': per_name,
                             '年龄': per_age,
                             '班级': per_class,
                             '科目': per_course,
                             '成绩': per_grade },)
            person_info.append(per_dict) # 保存到列表中
            print(' 保存成功')
    elif fun_num == '2':
```

教
学
内
容

```
if len(person_info) == 0:
    print('无信息')
for i in person_info:
    print('--*' * 6)
    for title, info in i.items():
        print(title + ':' + info)
    print('--*' * 6)
elif fun_num == '3': # 删除
    if len(person_info) != 0:
        del_name = input('请输入要删除的联系人姓名:')
        for i in person_info:
            if del_name in i.values():
                person_info.remove(i)
                print(person_info)
                print('删除成功')
            else:
                print('该同学不在成绩表中')
        else:
            print('无信息')
elif fun_num == '4': # 修改
    if len(person_info) != 0:
        modi_info = input('请输入要修改的联系人姓名:')
        for i in person_info:
            if modi_info in i.values():
                # 获取所在元组在列表中的索引位置
                index_num = person_info.index(i)
                dict_cur_perinfo = person_info[index_num]
                for title, info in dict_cur_perinfo.items():
                    print(title + ':' + info)
                per_name = input('请输入学生的姓名:')
                per_age = input('请输入学生的年龄:')
                per_class = input('请输入学生的班级:')
                per_course = input('请输入学生的科目:')
                per_grade = input('请输入学生的成绩:')
                dict_cur_perinfo.update(姓名= per_name)
                dict_cur_perinfo.update(年龄= per_age)
```

```

        dict_cur_perinfo.update(班级= per_course)
        dict_cur_perinfo.update(科目= modi_address)
        dict_cur_perinfo.update(成绩= per_grade)
        print(person_info)

    else:
        print('无信息')
elif fun_num == '5': # 查找
    if len(person_info) != 0:
        query_name = input('请输入要查找的学生姓名: ')
        for i in person_info:
            if query_name in i.values():
                index_num = person_info.index(i)
                for title, info in person_info[index_num].items():
                    print(title + ':' + info)
                break
            else:
                print('该学生不在成绩表中')
        else:
            print('无信息')

elif fun_num == '6': # 退出
    break

```

课程思政：

1. 诚信是一种人们在立身处世、待人接物和生活实践中必须而且应当具有的真
诚无欺、实事求是的态度和信守然诺的行为品质，其基本要求是说老实话、办老实
事、做老实人。

2. 诚信是一种社会的道德原则和规范，它要求人们以求真务实的原则指导自己
的行动，以知行合一的态度对待各项工作。

3. 引导学生做人要诚信：做到学业诚信不作弊；贷款诚信不逾期；做事诚信不
违约。

三、归纳总结

1. 回顾上课前的学习目标，对本节课知识点进行总结

带领学生回顾创建字典元素和操作字典元素。

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

(使用集合实现生词本)

一、回顾任务二内容，继续介绍本课时的内容

1. 教师对学生们的疑问进行统一答疑。
2. 教师通过提问学生问题，由任务二引出本次任务要讲解的内容。
3. 明确学习目标
 - (1) 要求学生掌握集合的创建方式
 - (2) 要求学生掌握集合元素的添加删除和清空
 - (3) 要求学生了解集合类型的操作符
 - (4) 要求学生掌握生词本的实现过程

二、进行重点知识的讲解

1. 什么是集合

Python 中集合可分为可变集合与不可变集合，可变集合由 `set()` 函数创建，集合中的元素可以动态地增加或删除；不可变集合由 `frozenset()` 函数创建，集合中的元素不可改变。

2. 集合的创建方式

列表可以存储任何元素，当然也可以存储列表，如果列表存储的元素也是列表，则称为嵌套列表。

3. 集合元素的添加删除和清空

Python 中可变集合支持添加、删除和情况元素，可变集合的 `add()` 方法或 `update()` 方法都可以实现向集合中添加元素，不同的是，`add()` 方法只能添加一个元素，而 `update()` 方法可以添加多个元素；`remove()`、`discard()`、`pop()` 方法删除可变集合中的元素；`clear()` 方法可以清空集合中的元素。

4. 集合类型的操作符

Python 支持通过操作符 `|`、`&`、`-`、`^` 对集合进行联合、取交集、差补和对称差分操作。

5. 使用集合实现生词本

教师根据教材资源实现案例：生词本，并向学生讲解其实现过程。

```
words_book = set()
words_only_set = set()
print('=' * 20)
print('欢迎使用生词本')
print('1. 查看生词本')
print('2. 背单词')
print('3. 添加新单词')
print('4. 删除单词')
```

```

print('5. 清空生词本')
print('6. 退出生词本')
print('=' * 20)
while True:
    word_dict = {}
    fun_num = input('请输入功能编号: ')
    if fun_num == '1': # 查看生词本
        if len(words_book) == 0:
            print('生词本内容为空')
        else:
            print(words_book)
    elif fun_num == '2': # 背单词
        if len(words_book) == 0:
            print('生词本内容为空')
        else:
            for random_words in words_book:
                w = random_words.split(':')
                in_words = input("请输入" + w[0] + '翻译': '\n')
                if in_words == w[1].strip():
                    print('太棒了')
                else:
                    print('再想想')
    elif fun_num == '3': # 添加新单词
        new_words = input('请输入新单词: ')
        # 检测单词是否重复
        if new_words in words_only_set:
            # 添加的单词重复
            print('此单词已存在')
        else:
            # 执行单词添加
            new_chinese = input('请输入单词翻译: ')
            word_dict.update({new_words: new_chinese})
            # 转换成字符串存入 set 集合中
            dict_str = str(word_dict).replace('{', '').replace('}',
            '').replace("'", '')
            words_book.add(dict_str)

```

```

        print('单词添加成功')
        dict_str = dict_str.replace(',',' ')
        print(dict_str)
        words_only_set.add(new_words)
elif fun_num == '4': # 删除单词
    if len(words_book) == 0:
        print('生词本为空')
    else:
        temp_list = list(words_book)
        print(temp_list)
        del_wd = input("请输入要删除的单词")
        # 如果要删除的单词不在单词集合中提示 单词不存在
        if del_wd not in words_only_set:
            print('删除的单词不存在')
        else:
            words_only_set.discard(del_wd)
            for temp in temp_list:
                if del_wd in temp:
                    words_book.remove(temp)
                    print('删除成功')
elif fun_num == '5': # 清空
    if len(words_book) == 0:
        print('生词本为空')
    else:
        words_only_set.clear()
        words_book.clear()
        print('生词本清空成功')
elif fun_num == '6': # 退出
    print('退出成功')
    break

```

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
带领学生回顾集合的创建和操作。
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。

教学 内容 总结	任务	知识目标	思政元素
	任务一 运用列表管理好友 信息	(1) 熟悉组合数据类型的 分类 (2) 掌握列表的定义 (3) 掌握列表的基本用法	1. 诚信是一种人们在立 身处世、待人接物和生活实践 中必须而且应当具有的真实 无欺、实事求是的态度和信守 然诺的行为品质,其基本要求 是说老实话、办老实事、做老 实人。 2. 诚信是一种社会的道 德原则和规范,它要求人们以 求真务实的原则指导自己的 行动,以知行合一的态度对待 各项工作。 3. 引导学生做人要诚信: 做到学业诚信不作弊;贷款诚 信不逾期;做事诚信不违约。
	任务二 使用字典类型实现 学生成绩管理系统	(1) 掌握字典元素的添加 和修改 (2) 掌握字典元素的删除 (3) 掌握字典元素的查询 (4) 了解实例手机通讯录 实现过程	
	任务三 使用集合实现生词 本	(1) 生掌握集合的创建方 式 (2) 掌握集合元素的添加 删除和清空 (3) 了解集合类型的操作 符 (4) 掌握生词本的实现过 程	
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	运用组合数据类型实现通讯录		