**Java基础**

**第三单元 容器框架及List集合**

1. **本单元教学目标：**
2. **知识目标**
3. 理解Collection集合的概念
4. 掌握List接口的常用方法
5. 了解泛型的使用
6. 掌握ArrayList和LinkedList集合类的常用方法
7. 掌握ArrayList和LinkedList集合类的区别
8. 掌握Iterator接口的常用方法
9. 了解ArrayList和Vector集合类的区别
10. **能力目标**
11. 能够熟练使用ArrayList集合类对数据进行存储
12. 能够熟练使用LinkedList集合类对数据进行存储
13. 能够熟练使用Iterator接口进行集合遍历
14. 能够熟练使用增强for循环进行集合遍历
15. **课时分配**
16. **第一课时：Collection集合概念**
17. **知识目标**

* 理解集合概念
* 掌握集合的分类
* 理解List集合概念
* 掌握List接口特点

1. **能力目标**

* 能够理解Collection集合的层次结构
* 能够掌握List集合的实现类

1. **第二课时：ArrayList集合类及Vector集合类**
2. **知识目标**

* 理解ArrayList集合类概念
* 掌握ArrayList集合类的特点
* 掌握ArrayList集合类常用方法
* 理解ArrayList集合类与Vector集合类的区别

1. **能力目标**

* 能够创建ArrayList集合类对象
* 能够熟练使用ArrayList集合类对数据进行存储

1. **第三课时：迭代器及Queue集合**
2. **知识目标**

* 理解迭代器概念
* 掌握Iterator接口的常用方法
* 理解Queue集合类概念
* 掌握Queue集合类特点

1. **能力目标**

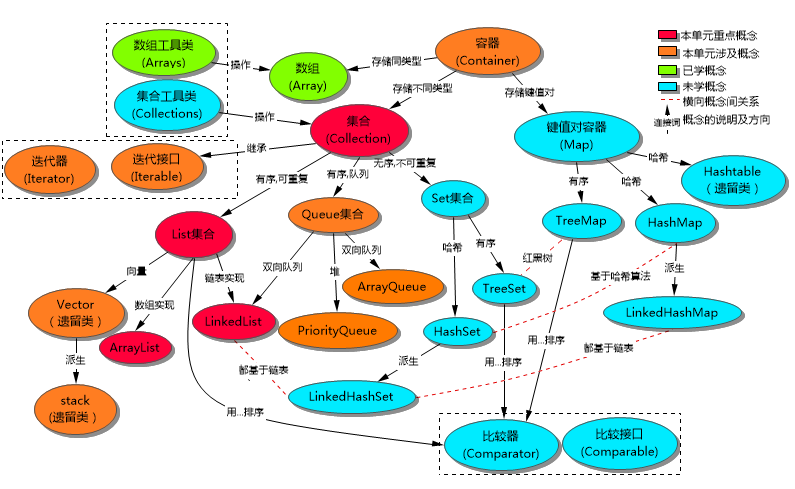
* 能够熟练使用Iteratorr迭代器遍历集合
* 能够熟练使用foreach循环遍历集合

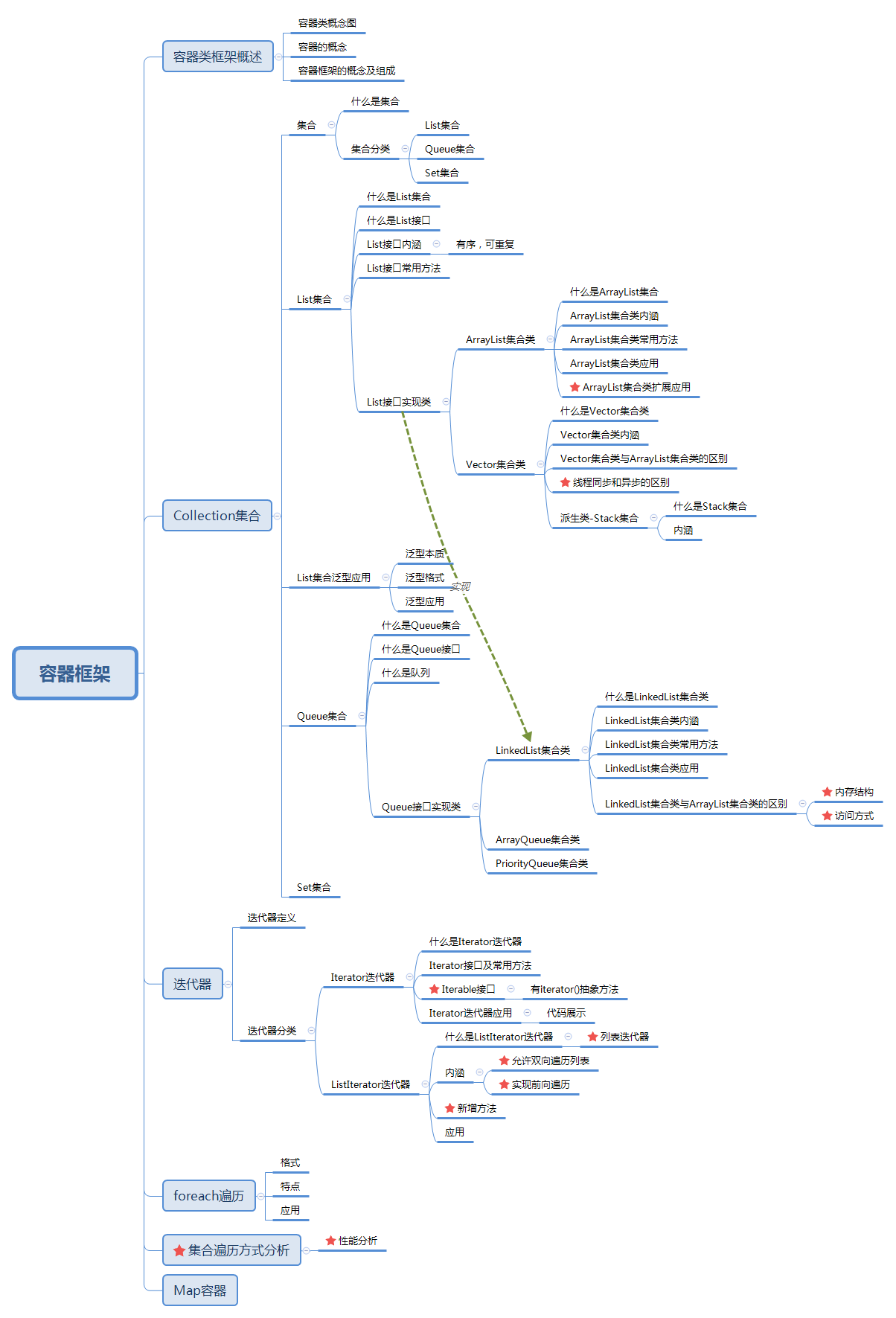
1. **第四课时：LinkedList集合类**
2. **知识目标**

* 理解LinkedList集合类概念
* 理解与ArrayList集合类的区别

1. **能力目标**

* 能够创建LinkedList集合类对象
* 能够熟练使用LinkedList集合类对数据进行存储

1. **本单元概念图**
2. **本单元知识结构图**

****

1. **教学过程脚本设计—第一课时：**
2. **45分钟脚本设计思路**

****

1. **每个脚本切片设计思路：**

|  |  |  |  |
| --- | --- | --- | --- |
| **内容** | **目标** | **设计亮点** | **时间** |
| 课堂礼仪 | 提升学生气势，精神饱满，增加团队意识 | 突出学院特色，增加学生学习的信心，坚定学习信念 | 0.23 |
| 课程回顾 | 承上，检查上节课知识点，知识体系化 | 学生互动--代码阅读 | 3.14 |
| 为什么使  用集合 | 启下，说明使用集合原因，明确本单元目标 | 现象分析--实际问题分析--发现问题--解决问题 | 3.20 |
| 什么是  容器 | 通过容器框架概念图及词源分析，理解容器概念及组成 | 词源及概念分析--逐步引导--层层深入 | 4.44 |
| 什么是集合 | 通过词源分析，理解概念，掌握集合组成 | 词源及概念分析--逐步引导 | 4.04 |
| 什么是  List集合 | 阐述List集合及List接口概念，总结List接口特点及常用方法，引出实现类 | 通过种差+属方式给概念下定义--总结法 | 4.13 |
| 什么是ArrayList集合 | 通过List接口引入，与数组作为关联点讲解，以种差+属方式下定义，使学生深入理解概念 | 对比法--种差+属方式下定义--逐步引导 | 3.04 |
| ArrayList集合  基本应用 | 通过重点语句讲解，步骤分析，使学生对ArrayList集合的应用初步理解 | 对比法--分析案例--步骤分析 | 6.06 |
| 内容小结 | 通过归纳法Collection集合，为学生梳理知识点，将知识成体系 | 对比法总结概念之间区别和联系--特点归纳法 | 2.46 |

1. **脚本切片一：[视频切](C:\\Users\\李巍\\AppData\\Roaming\\Microsoft\\Word\\1、课堂礼仪1‘.mp4)****[片1](C:\\Users\\李巍\\AppData\\Roaming\\Microsoft\\Word\\1、课堂礼仪1‘.mp4)**

|  |  |  |
| --- | --- | --- |
| **课堂模块/时长** | **教学内容** | **教学标准** |
| **课堂礼仪**  **(0.23)** | 上课铃响后，  班长喊起立—学生问好—班级口号—鼓掌—整理着装—欢迎教师上课。 | 1.上课前，提前30秒面带自信的笑容的走进教室，观察、问候学生们(等待上课)。  上课铃声响起！  2.老师检查学生们工装，整理工装速度及姿势。  IMG_256IMG_256 |
| **讲师：**上课  **学生：**老师好  **讲师：**同学们好，请坐。 | 1. 讲师严格检查学生整齐度和速度，保证上课状态，如凌乱需要重新做。  2. 学生礼仪内容：学院口号，班级口号，英文口号，鼓掌，整理工装，班长报出勤。  3. 讲师授课过程中要面带微笑（八颗牙齿）。  4. 插入合理手势，避免课堂枯燥。 |

1. **脚本切片二：[视频切片2](C:\\Users\\李巍\\AppData\\Roaming\\Microsoft\\Word\\1、课堂礼仪1‘.mp4)**

|  |  |  |
| --- | --- | --- |
| **课堂模块/时长** | **教学内容/台词** | **教学标准/素材** |
| **课程回顾**  **(3.14)** | **1、准备讲解本节课内容**  **讲师：**我们这节课讲解第三单元容器框架及List集合的内容： | 1、容器框架及List集合声音高于授课声音。 |
| **2、回顾**  **讲师：**大家回顾下上节课内容：什么是泛型？  **学生：**各抒己见。  **讲师：**好，非常好！泛型有哪些规则限制？  **讲师：**泛型的应用场合？  **学生：**各抒己见。  **讲师：**好，非常好！另一个问题泛型通配符有哪些？  **学生：**各抒己见。  **讲师：**对，非常好！看来大家对于昨天单元的内容掌握的不错，继续努力。 | 1. 以ppt文字逐步引导。 2. 互动方式。 |

1. **脚本切片三：[视频切片3](C:\\Users\\李巍\\AppData\\Roaming\\Microsoft\\Word\\1、课堂礼仪1‘.mp4)**

|  |  |  |
| --- | --- | --- |
| **课堂模块/时长** | **教学内容/台词** | **教学标准/素材** |
| **贯彻案例**  **(3.14)** | **1、贯彻案例讲解**  **讲师：**首先学习本单元的贯彻案例。  **讲师：**先看一下需求说明：  **讲师：**使用List存储当前在线人员。  **讲师：**看一下效果图：  **讲师：**在开心聊天室中当前的在线用户如：张三、李四、王五。  **讲师：**可以使用List进行存储。  **讲师：**哪如何存储呢？  **讲师：**请接下来学习本单元的内容。 | 1. 以ppt文字逐步引导。 2. 互动方式。 |

1. **脚本切片四：[视频切片5、6](C:\\Users\\李巍\\AppData\\Roaming\\Microsoft\\Word\\1、课堂礼仪1‘.mp4)**

|  |  |  |
| --- | --- | --- |
| **课堂模块/时长** | **教学内容/台词** | **教学标准/素材** |
| **知识目标**  **概念图**  **(4.44)** | 1. **容器框架讲解**   **讲师：**完成贯彻案例，需熟知本单元知识目标。  **讲师：**本单元知识是以容器框架展开的。  **讲师：**什么是容器框架呢？  **讲师：**容器框架英文Container Taxonomy。  **讲师：**是为了表示和操作容器的统一标准的体系结构。  **讲师：**下面一起学习本单元容器框架图。  **讲师：**容器框架图的顶层是容器。  **讲师：**包括存储同种类型的数组Array。  **讲师：**还包括本单元学习存储不同类型的集合Collection。  **讲师：**以及后面单元学习存储键值对的Map容器。  **讲师：**这里重点讲解集合Collection包括有序可重复的List集合和有序队列Queue。  **讲师：**以及无序不可重复的Set集合。  **讲师：**在List集合中包括向量Vector，以及派生类Stack。**讲师：**基于数组实现的ArrayList集合和基于链表实现的LinkedList集合。  **讲师：**LinkedList集合同时实现了Queue接口。  **讲师：**Queue接口还提供基于优先级堆的PriorityQueue队列和双向队列ArrayQueue。  **讲师：**在本单元将要使用之前数据结构中的内容：如向量，数组，链表，队列。 | 1、采用互动方式，逐步引导讲解。 |

1. **脚本切片五：[视频切片7](C:\\Users\\李巍\\AppData\\Roaming\\Microsoft\\Word\\1、课堂礼仪1‘.mp4)**

|  |  |  |
| --- | --- | --- |
| **课堂模块/时长** | **教学内容/台词** | **教学标准/素材** |
| **为什么使**  **用集合**  **(3.20)** | **1、生活现象分析**  **讲师：**先分析现象一：看一下QQ消息记录如何存储？  **学生：**数组。  **讲师：**分析一下数组的特点，类型相同，长度固定；  **讲师：**可是QQ消息可无限追加，不限长度并且类型不同。  **讲师：**这里可以添加文件和图片。  **讲师：**不可以使用数组存储，如何解决呢？  **讲师：**再看一个现象京东购物车，如何存储呢？  **学生：**数组。  **讲师：**可以用数组吗？一起分析一下：  **讲师：**这里商品类型不一致，商量数量不固定。  **讲师：**并且添加商品时没有固定顺序；  **讲师：**数组是不能满足需求的，这就需要用到今天讲解的集合。 | 1、在提问问题时，老师需要演示ppt上动画。  2、引导学生逐步理解使用集合意义。 |
| **2、实际问题分析**  **讲师：**再来分析一个实际问题：  **讲师：**如何将一个班级升班同学信息（学号，姓名，性别，年龄，成绩）保存起来？  **学生：**数组。  **讲师：**先看一下用数组实现的代码：  **讲师：**这里升班人数即N值如何确定呢？  **讲师：**考试成绩没出来之前，怎么知道谁升班还是降班呢？  **学生：**不知道。  **讲师：**怎么办呢？  **讲师：**如果不确定存储的对象数。  **讲师：**或需要存储复杂的引用数据类型。  **讲师：**就需要使用今天讲解的集合。 | 1、以案例引导的方式引入。  2、互动方式让学生理解使用集合的必要性。 |

1. **脚本切片六：[视频切片8](C:\\Users\\李巍\\AppData\\Roaming\\Microsoft\\Word\\1、课堂礼仪1‘.mp4)**

|  |  |  |
| --- | --- | --- |
| **课堂模块/时长** | **教学内容/台词** | **教学标准/素材** |
| **什么是集合**  **(4.04)** | **1、集合概念讲解**  **讲师：**大家现在从屏幕上能看到什么？  **讲师：**杂乱无章的书！  **讲师：**这是什么?  **学生：**盒子。  **讲师：**对!现在干什么呢?将书收集起来。  **讲师：**收集的英文是Collection，含义是收集、采集；收藏品。  **讲师：**那盒中的书可以用什么存储？  **学生：**数组**。**  **讲师：**可以，是否可以存入一些玩具呢？  **学生：**不可以。  **讲师：**类型不同。  **讲师：**那即可存放一种或多种引用类型。  **讲师：**并且还可以在上面增加书，长度是可变的。  **讲师：**用什么存储呢?  **学生：**容器。  **讲师：**对，是容器**。**那什么是容器呢？ | 1、通过词源分析，引出集合定义。 |
| **2、Collection集合定义**  **讲师：**容器是用来存储和组织其它对象的对象。  **讲师：**今天讲的集合就是容器的一种。  **讲师：**还是用种差+属的方式给集合下定义：  **讲师：**先分析临近的属是？  **学生：**容器。  **讲师：**再分析同属的其它种有哪些？  **学生：**Arrays，Map容器。  **讲师：**对，那它们的种差是什么？  **讲师：**与之前讲过的数组Arrays对比。  **讲师：**数组是存储一种数据类型，并长度固定。  **讲师：**那集合呢？结合刚才存放玩具的案例。  **讲师：**可以存储一种或多种引用类型数据，并且长度可变的。  **讲师：**所以集合的定义是在计算机中用于存储一种或多种引用类型数据，并且长度可变的容器。  **讲师：**再分析一下它的外延：  **讲师：**由List集合、Queue集合和Set集合；  **讲师：**那我们先来学习List集合。 | 1、逐步引导，使学生深入理解集合特点。  2、重点强调种差+属定义方式。 |

1. **脚本切片七：[视频切片9](C:\\Users\\李巍\\AppData\\Roaming\\Microsoft\\Word\\1、课堂礼仪1‘.mp4)**

|  |  |  |
| --- | --- | --- |
| **课堂模块/时长** | **教学内容/台词** | **教学标准/素材** |
| **什么是**  **List集合**  **(4.13)** | **1、List集合定义：**  **讲师：**什么是List集合呢？  **讲师：**屏幕上网易新闻列表，京东商品列表，QQ消息列表三张图片有什么特点？  **讲师：**首先以QQ消息列表为例：  **讲师：**先发送的信息是不是在上面？  **学生：**是。  **讲师：**所以List集合存放的信息是有序的。  **讲师：**另外经常重要的事情说三遍，也就是说可以发送相同的信息。  **讲师：**还是利用种差+属的方式给List集合下定义。  **讲师：**那它临近的属是什么？  **学生：**集合。  **讲师：**List集合没有同属的其它种。  **讲师：**那它的内涵有哪些？  **讲师：**如之前QQ消息列表存储有序，并且可以重复。  **讲师：**那List集合的定义是在计算机中用于有序存储可重复的集合。  **讲师：**再看一下它的外延：有ArrayList集合，Vector集合，及LinkedList三个集合。 | 1. 讲解引导学生根据种差+属给List集合下定义； 2. 结合生活现象讲解。 3. 把数组作为关联点层层深入引导学生给概念定义。 |
| 1. **什么是List接口**   **讲师：**List集合是通过List接口实现的。  **讲师：**那什么List接口呢?  **讲师：**从容器框架图中可以看出List接口是继承Collection接口的。  **讲师：**并且具有有序和可重复的功能。  **讲师：**那它的属是？  **学生：**接口。  **讲师：**对是公共接口。  **讲师：**List接口的定义是容器框架中继承了Collection接口，  声明有序存储对象（可重复）功能的公共接口。  **讲师：**刚刚讲解过List集合的外延类都实现了List接口。 | 1、以ppt文字方式逐步引导。  2、互动方式，学生占主导。 |
| **3、List接口常用方法**  **讲师：**接下来学习List接口的常用方法。  **讲师：**提供的主要方法有：  **讲师：**一、add()方法，在指定位置上添加一个对象。  **讲师：**以及addAll()方法，将Collection类型对象c添加到指定位置。  **讲师：**二、listIterator()方法，返回一个listIterator对象或返回指定位置的listIterator对象.  **讲师：** 三、get()方法返回指定位置的元素。  **讲师：**四、remove()方法删除指定位置元素。  **讲师：**五、set()方法用目标位置的元素取代指定位置元素，即为修改。  **讲师：** 六、indexOf()和laseindexOf()返回第一个和最后一个匹配对象的位置。 | 1、以ppt文字讲解同时打开API文档，让大家查询List接口。  2、互动方式，学生占主导。 |

1. **脚本切片八：[视频切片10](C:\\Users\\李巍\\AppData\\Roaming\\Microsoft\\Word\\1、课堂礼仪1‘.mp4)**

|  |  |  |
| --- | --- | --- |
| **课堂模块/时长** | **教学内容/台词** | **教学标准/素材** |
| **什么是**  **ArrayList集合(3.04)** | **1、什么是ArrayList集合**  **讲师：**实现List接口的类有：ArrayList集合，Vector集合，LinkedList集合。  **讲师：**先来学习第一个实现类ArrayList集合。  **讲师：**那什么是ArrayList集合？  **讲师：**还是利用种差+属的方式下定义。  **讲师：**它的属是什么？  **学生：**List集合。  **讲师：**好，非常好！再考虑同属的其它种有哪些？  **学生：**Vector和LinkedList集合。  **讲师：**对，那它们的种差是什么？即内涵。  **讲师：**在讲概念图中提到ArrayList集合是基于数组实现的，并且长度是可变的。  **讲师：**数组在内存中是连续存储的，即分配连续内存空间。  **讲师：**所以ArrayList集合的定义是基于数组实现的、长度可变的、分配连续内存空间的List集合。 | 1. 以ppt文字逐步引导。 2. 互动方式。 |

1. **脚本切片九：[视频切片11](C:\\Users\\李巍\\AppData\\Roaming\\Microsoft\\Word\\1、课堂礼仪1‘.mp4)**

|  |  |  |
| --- | --- | --- |
| **课堂模块/时长** | **教学内容/台词** | **教学标准/素材** |
| **ArrayList**  **集合基本应用**  **(6.06)** | **1、案例简介**  **讲师：**ArrayList集合如何应用呢？  **讲师：**看课前提到的案例：  **讲师：**如何将一个班级升班同学信息（学号，姓名，性别，年龄，成绩）保存起来。  **讲师：**案例需求：  **讲师：**通过List接口实现类ArrayList实现,输出学生信息。  **讲师：**如学号101，姓名小芳，性别为女等基本信息。  **讲师：**如何实现呢？ | 1、讲师声调阴阳顿挫，讲重点和难点时声音加重。  2、互动环节中讲师需关注每一个学生参与情况，发现问题所在。 |
| **2、实现步骤**  **讲师：**下面分析实现步骤：  **讲师：**首先创建集合对象。  **讲师：**在ArrayList中提供三种构造方法：分别是：  **讲师：**第一种：构造一个初始容量为十的空列表，即默认无参的构造方法。  **讲师：**第二种：构造指定初始容量的空列表初始化容量为10。  **讲师：**第三种：构造一个包含指定集合元素的列表**。**  **讲师：**其次将指定元素添加到集合，即追加到列表末尾。  **讲师：**在ArrayList中提供add（）方法，返回值为boolean类型，即是否添加成功。  **讲师：**如语句list.add("小芳");将小芳信息添加到列表末尾。  **讲师：**最后是打印输出，有两种方式：  **讲师：**第一种是System.out.println(list);输出对象本身。  **讲师：**第二种是System.out.println(list.toString());将对象转换成字符串输出。  **讲师：**两种方式是等价的**。** | 1、在提问问题时，老师需要以思路图的形式展现。 |
| **3、代码实现**  **讲师：**下面分析代码是如何实现的？  **讲师：**首先创建集合对象list调用默认无参构造方法。  **讲师**：其次在集合中添加学号，姓名，性别等基本信息。  **讲师**：多种类型即为Object类型。  **讲师**：此处添加学号101小芳同学和学号202小明同学的基本信息。  **讲师：**最后使用println()方法打印输出。  **讲师：**大家回想println()方法中输出参数类型为什么？  **学生：**String。  **讲师：**对，非常好！  **讲师：**所以这里输出list对象等价输出list.toString()方法;  **讲师：**在ArrayList集合中重写了toString()方法。  **讲师：**输出形式用[,,]分割；  **讲师：**其中每一个元素为之前学习的包装类类型和String类型；  **讲师：**在包装类Integer，Character和Double中也重写了toString()方法。  **讲师：**所以最终才转换成字符串输出。  **讲师：**输出效果为：[,,]分隔每一个学生信息。 | 1、在提问问题时，老师需要以思路图的形式展现。 |

1. **脚本切片十：[视频切片12](C:\\Users\\李巍\\AppData\\Roaming\\Microsoft\\Word\\1、课堂礼仪1‘.mp4)**

|  |  |  |
| --- | --- | --- |
| **课堂模块/时长** | **教学内容/台词** | **教学标准/素材** |
| **内容小结**  **(2.46)** | **1、演示案例：**  **讲师：**下面一起总结本节课内容：  **讲师：**首先回顾Collection集合的定义是在计算机中用于存储一种或多种引用类型数据，并且长度可变的容器。  **讲师：**其次外延由List集合，Queue集合及Set集合组成。  **讲师：**什么是List集合？  **讲师：**List集合是在计算机中用于有序存储可重复的集合。  **讲师：**List接口实现List集合。  **讲师：**具有ArrayList，Vector和LinkedList三个实现类。  **讲师：**最后什么是ArrayList集合？  **讲师：**ArrayList集合是实现了长度可变的数组，在内存中分配连续空间的集合。 | 1、总结法。  2、互动法，学生主导。  3、对比法总结概念之间的区别和联系。 |

1. **教学过程脚本设计—第二课时：**
2. **45分钟脚本设计思路：**



1. **每个脚本切片设计思路：**

|  |  |  |  |
| --- | --- | --- | --- |
| **内容** | **目标** | **设计亮点** | **时间** |
| ArrayList集合  应用 | 通过重点语句讲解，思路图分析，使学生对ArrayList集合的应用深入理解 | 对比法--分析案例--思路图 | 4.13+5.13 |
| ArrayList集合  扩展应用 | 通过重点方法讲解，代码分析，使学生能熟练应用ArrayList集合进行数据存储 | 运行结果--代码分析--演示效果 | 2.49 |
| ArrayList集合  泛型应用 | 通过代码演示，体会使用泛型的意义 | 代码分析--学生互动 | 4.50+2.19 |
| 什么是  Vector集合类 | 通过Vector本义理解，引出特点，通过动画着重讲解特点 | 对比法--动画显示--层层深入 | 3.02 |
| 什么是Stack集合及Vector和ArrayList区别 | 通过Vector和ArrayList集合类对比讲解，重点突出ArrayList集合类，通过阐述派生类Stack集合，重点强调特点 | 对比法--图形显示--层层深入 | 2.37 |
| 内容小结 | 通过贯穿案例讲解和特点总结，为学生梳理思路 | 总结法--特点归纳法 | 2.54 |

1. **脚本切片一：[视频切片1](C:\\Users\\李巍\\AppData\\Roaming\\Microsoft\\Word\\1、课堂礼仪1‘.mp4)**

|  |  |  |
| --- | --- | --- |
| **课堂模块/时长** | **教学内容/台词** | **教学标准/素材** |
| **ArrayList**  **集合应用**  **(4.13+5.32)** | 1. **案例简介**   **讲师：**本节课继续讲解ArrayList集合应用。  **讲师：**首先看一下案例简介，定义一个狗类，狗有：名字，种类等属性。狗具有吃啃骨和跑的行为。  **讲师：**需求：存储多条狗狗信息，获取狗狗总数，逐条打印出各狗狗信息。  **讲师：**通过List接口实现类ArrayList实现该需求。  **讲师：**思路：1、元素个数不确定；  **讲师：**2、要求获得元素的实际个数；  **讲师：**3、并且按照存储顺序获得并打印元素信息；  **讲师：**效果为：输出共计有4条狗狗。  **讲师：**分别是输出狗狗信息。 | 1、讲师声调阴阳顿挫，讲重点和难点时声音加重。  2、互动环节中讲师需关注每一个学生参与情况，发现问题所在。 |
| **2、实现步骤**  **讲师：**下面分析一下实现步骤：  **讲师：**第一步先要创建对象。  **讲师：**那先建什么对象？  **讲师：**因为要存狗狗对象，就要先创建狗狗对象。  **讲师：**创建狗狗对象的前提是定义狗狗类。  **讲师：**接下来创建狗类对象yayaDog和集合对象dogs。  **讲师：**第二步将狗类对象添加到集合。  **讲师：**仍然采用add()方法添加。  **讲师：**最后打印输出，可以采用两种方式输出。  **讲师：**第一输出集合对象本身相对于调用它的toString()。  **讲师：**第二用循环遍历输出。 | 1、在提问问题时，老师需要逐步引导。 |
| **3、代码演示**  **讲师：**下面分析代码实现：  **讲师：**首先定义狗类。  **讲师：**在类中声明名字和品种两个属性。  **讲师：**其次在测试类中声明亚亚，美美等狗类对象；  **讲师：**并且创建ArrayList集合对象dogs。  **讲师：**接下来将对象放入集合；  **讲师：**之后用size()方法获取狗狗数量，并输出狗类信息。  **讲师：**运行结果如何？  **讲师：**先输出共计有3条狗狗。这是正确的。  **讲师：**下面出现了完整包名+@+哈希码的形式。  **讲师：**这种形式在哪里遇见过？  **学生：**在Object类中toString()方法输出时。  **讲师：**对，非常好！  **讲师：**这说明直接输出dogs对象调用的toString()为Object类继承过来的方法。  **讲师：**大家思考这种输出形式满足子类需求吗？  **学生：**不满足。  **讲师：**那继承过来的方法不满足子类需求时，应该怎么办？  **学生：**在子类中重写继承过来的方法。  **讲师：**对，非常好！怎么重写呢？  **讲师：**需要在狗类中重写toString()方法。  **讲师：**toString()方法返回值为String，通过return语句返回名称+品种的字符串形式。  **讲师：**最后输出效果为狗狗名字和品种的基本信息。 | 1、在提问问题时，老师需要以代码及运行效果的形式展现。 |

1. **脚本切片二：[视频切片2](C:\\Users\\李巍\\AppData\\Roaming\\Microsoft\\Word\\1、课堂礼仪1‘.mp4)**

|  |  |  |
| --- | --- | --- |
| **课堂模块/时长** | **教学内容/台词** | **教学标准/素材** |
| **ArrayList**  **集合扩展应用**  **(2.49)** | 1. **案例扩展**   **讲师：**下面对之前案例进行扩展：  **讲师：**功能扩充：  **讲师：**删除指定位置的狗狗，如第一个狗狗。  **讲师：**删除指定的狗狗，如删除yayaDog对象。  **讲师：**判断集合中是否包含指定狗狗。  **讲师：**显示效果集合中狗狗数量和狗狗信息，及成功将亚亚信息删除。  **讲师：**思路提示：  **讲师：**利用List接口提供方法remove(),contains()来实现此功能。 | 1、讲解方法：ppt及API演示。 |
| 1. **代码演示**   **讲师：**下面分析代码实现：  **讲师：**上面同理创建狗狗对象和集合对象；  **讲师：**将狗狗对象放入集合。  **讲师：**下面用循环遍历集合元素：  **讲师：**在循环中首先用contains()方法判断是否包含yaya狗狗对象；  **讲师：**如果存在调用集合对象remove()方法删除;  **讲师：**用break语句结束循环。  **讲师：**最后删除亚亚狗狗信息。  **讲师：**大家考虑这里break语句是否可以省略？  **讲师：**这个问题做为同学们课后研究的任务。  **讲师：**最后效果符合题目需求，打印成功将亚亚信息删除**。** | 1、采用互动方式，逐步引导讲解。  2、讲到删除第一个时声音提高。 |

1. **脚本切片三：[视频切片3](C:\\Users\\李巍\\AppData\\Roaming\\Microsoft\\Word\\1、课堂礼仪1‘.mp4)**

|  |  |  |
| --- | --- | --- |
| **课堂模块/时长** | **教学内容/台词** | **教学标准/素材** |
| **ArrayList**  **集合泛型应用**  **(4.50+2.19)** | **1、为什么使用泛型集合**  **讲师：**下面分析这段代码有什么问题？  **讲师：**尤其是1处。  **学生：**会出错。  **讲师：**对，运行时会出现“java.lang.ClassCastException”异常。  **讲师：**什么原因呢？  **讲师：**大家考虑ArrayList集合中添加方法参数是什么类型？  **学生：**Object类型。  **讲师：**对，这里前两项添加String类型和第三项添加Integer类型都是正确的；  **讲师：**但在遍历输出时，把Object类型强制转换成什么类型了？  **学生**：String。  **讲师：**对，那么编译器遇到Integer类型时就会报“java.lang.ClassCastException”异常；  **讲师：**程序员在开发时是不是也容易出现这样的问题？  **讲师：**这就需要用到前一单元泛型的知识。 | 1、通过代码展示，语音要分明。 |
| **2、什么是泛型集合**  **讲师：**什么是泛型集合？  **讲师：** 泛型集合是使用类型形参定义的集合。  **讲师：**如List<T>就是一个泛型集合。  **讲师：**可以将所有的泛型放在该集合中来。  **讲师：**如在List集合中加入<String>类型即所谓泛型集合。  **讲师：**应用泛型集合后在编译时增加类型检查。  **讲师：**所以1处添加会报编译异常，不允许的参数类型。  **讲师：**在运行时不用类型转换。  **讲师：**在2处记住了原存储数据类型String，不会出错。  **讲师：**这就是使用泛型好处。  **讲师：**那在案例中如何应用呢？ | 1、讲解时按照代码提示，把使用泛型原因描述清楚。  2、重点强调类型转换。语音加重。 |
| **3、泛型集合应用及优势**  **讲师**：下面分析泛型集合在之前案例中的应用。  **讲师**：此处在ArrayList集合中添加String类型即为泛型集合应用。  **讲师**：格式为类名<T>的形式。  **讲师**：此处添加元素只能为String类型。  **讲师**：循环中遍历直接取出来的元素类型即为String类型，不用类型转换。  **讲师**：再看一下泛型集合有什么优势呢？  **讲师**：ArrayList集合中使用泛型机制，在使用时可以传入Object及其任意子类。  **讲师**：构成了ArrayList集合的多个版本，提高了可重用性。  **讲师：**在编译期确定对象实际类型，可以提高执行效率；  **讲师：**如果不加泛型<Integer>,会报编译异常类型不匹配。  **讲师：**那如何改进呢？  **讲师：**在dogs.get(i)前面加(Integer)强制类型转换。  **讲师：**但应用泛型后就减少类型转换，体现类型安全(减少强制类型转换)的优势。 | 1、通过ppt展示，语音要分明。 |
| **4、课程编程**  **讲师**：下面完成本单元课堂案例：  **讲师**：功能描述：使用ArrayList保存学生信息（学号，姓名，年龄，成绩），要求删除指定学生后遍历输出。  **讲师：**先分析一下思路：  **讲师：**第一步创建三个学生对象保存到集合；  **讲师：**第二步删除一个对象；  **讲师：**第三步遍历集合；  **讲师：**给大家10分钟完成此案例。  **讲师：**各组需要采用pk方式完成**。**  **讲师：**各组加油！ | 1、通过ppt展示，语音要分明。 |

1. **脚本切片四：[视频切片4](C:\\Users\\李巍\\AppData\\Roaming\\Microsoft\\Word\\1、课堂礼仪1‘.mp4)**

|  |  |  |
| --- | --- | --- |
| **课堂模块/时长** | **教学内容/台词** | **教学标准/素材** |
| **什么是Vector集合类**  **(3.02)** | **1、什么是Vector集合类**  **讲师：**实现List接口的另一个实现类是Vector。  **讲师：**那什么是Vector？  **讲师：**先看一下名词解释：  **讲师：**Vector名词是向量，矢量，动词为引导的意思。  **讲师：**在数据结构中向量就是顺序表，一般以数组实现，也可以是动态数组，并且长度可以增长。  **讲师：**从jdk1.2版本后，这个类也实现[List](mk:@MSITStore:E:\\标准化\\参考资料\\文件\\jdk%20api%201.8_google.CHM::/java/util/../../java/util/List.html" \o "java.util中的接口)接口，使其成为[Java Collections Framework](mk:@MSITStore:E:\\标准化\\参考资料\\文件\\jdk%20api%201.8_google.CHM::/java/util/../../../technotes/guides/collections/index.html)中的一员 。  **讲师：** 与新集合类实现不同， Vector是同步的。  **讲师：**如果不需要线程安全的实现，建议使用[ArrayList](mk:@MSITStore:E:\\标准化\\参考资料\\文件\\jdk%20api%201.8_google.CHM::/java/util/../../java/util/ArrayList.html" \o "java.util中的类)代替Vector 。  **讲师：**什么是Vector集合类呢？  **讲师：**Vector集合类（向量）：是基于数组实现的，线程同步的遗留集合类。  **讲师：**遗留即为早期版本中使用过，后期版本仍然保留。  **讲师：**具有什么特点？  **讲师：**Vector集合类特点：  线程同步的；  多线程安全的；  效率低；  **讲师：**线程同步及线程安全会在后面讲解。 | 1. 互动法引导学生总结Vector特点。 |
| **2、线程安全与不安全的区别**  **讲师：**刚刚提到线程安全，那什么是线程安全呢？  **讲师：**如这有一杯咖啡，线程安全即：  **讲师：** 有一群小朋友来喝咖啡；  **讲师：**必须等第一个小朋友喝完，通知第二个小朋友。  **讲师：**第二个小朋友才可以过来喝；  **讲师：** 大家思考这样小朋友之间会不会打仗？  **学生：**不会。  **讲师：**这就是所谓的线程安全。  **讲师：**与之对应即不安全？  **讲师：**先分析现象：  **讲师：**还是一群小朋友来喝，怎么喝呢？  **讲师：**这次大家一起喝。  **讲师：** 这个过程中思考是不是有的小朋友就喝不到了。  **讲师：**是不安全的。  **学生：**是。  **讲师：**但考虑这种喝法比一个一个喝的方式效率是不是会提高啊？  **学生：**会。  **讲师：**所以Vector的特点为线程同步，多线程安全和效率低。  **讲师：**因为Vector的处理方式和一个一个喝的方式类似。  **讲师：**关于线程同步和多线程安全的问题会在后面单元中详细讲解。 | 1、以ppt文字方式逐步引导。  2、互动方式，学生占主导。  3、讲师伴随动作手势提醒。 |

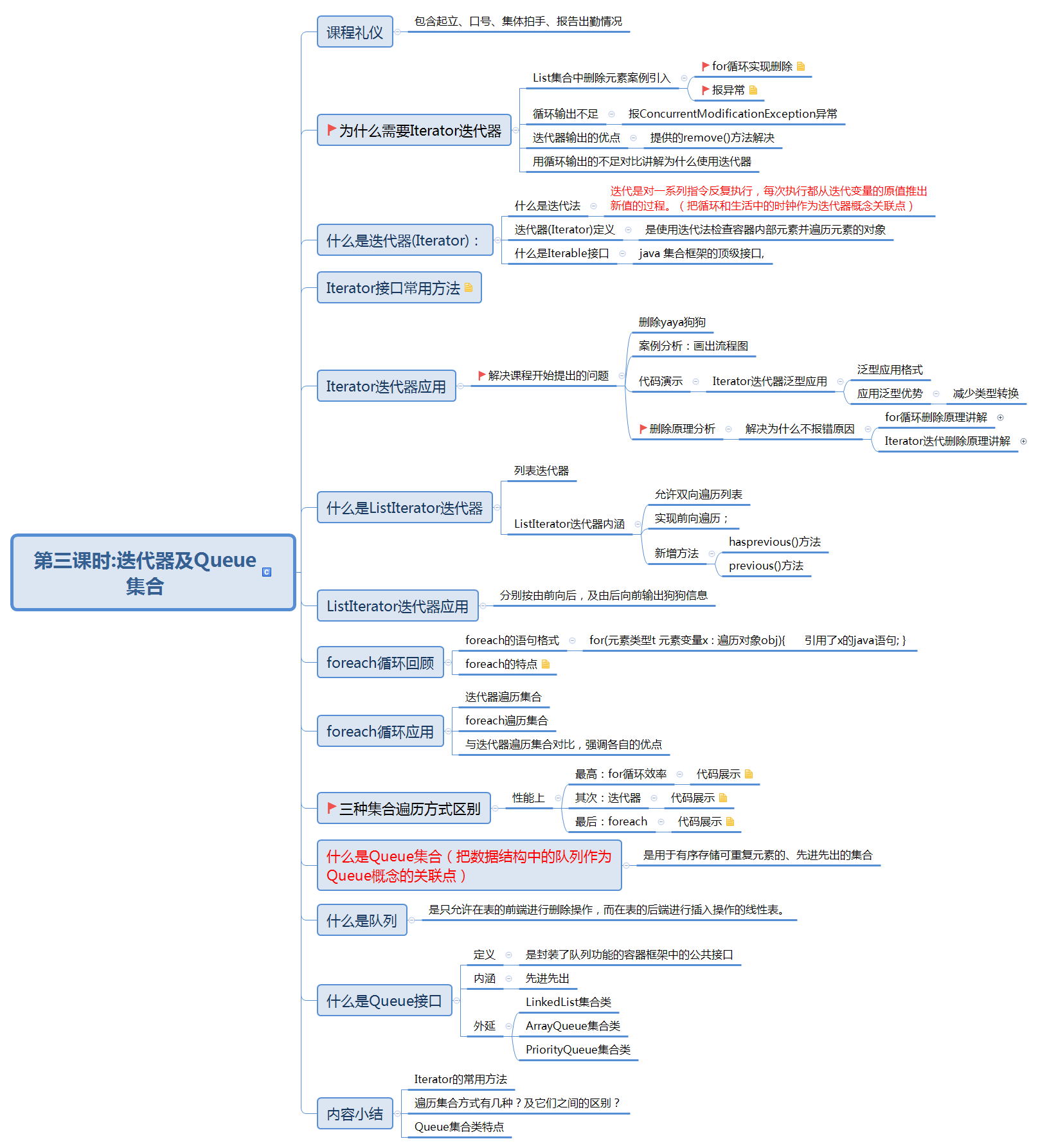
1. **脚本切片五：[视频切片5](C:\\Users\\李巍\\AppData\\Roaming\\Microsoft\\Word\\1、课堂礼仪1‘.mp4)**

|  |  |  |
| --- | --- | --- |
| **课堂模块/时长** | **教学内容/台词** | **教学标准/素材** |
| **什么是Stack集合及Vector和ArrayList集合类区别**  **(2.37)** | **1、什么是Stack集合**  **讲师：**下面学习Vector集合的派生类：Stack集合。  **讲师：**stack即数据结构中的栈。  **讲师：**什么是Stack集合？  **讲师：**是继承Vector集合的，具有后进先出特点的遗留集合类。  **讲师：**它的特点是什么?  **讲师：**后进先出。  **讲师：**栈是具有从一个指定方向，即栈顶插入和删除操作的线性表。  **讲师：**分别为入栈push操作和出栈pop操作。  **讲师**：这个内容可以作为了解内容，并不太常用。 | 1、以ppt文字方式逐步引导。  2、互动方式，学生占主导。  3、讲师伴随动作手势提醒。 |
| **2、Vector和ArrayList集合类的区别**  **讲师：**接下来讲解Vector和ArrayList集合的区别。  **讲师：**首先共同点是都具有List集合有序和可重复的特点。  **讲师：**那它们的区别是什么呢？  **讲师：**Vector是线程同步的，线程安全的。  **讲师：**线程同步打个比方如你在等一个人，一直等到人后才可以做其它事情。  **讲师：**这个过程中不会受其它事情影响，即线程安全。  **讲师：**但效率低，因为同一时间只做同一件事。  **讲师：**与之对应那什么是线程异步？  **讲师：**即等人同时可以干其它事情，如操作电脑。所以效率高。  **讲师：**但可能会受其它事情影响，即线程不安全的。  **讲师：**ArrayList集合类即是线程异步的。  **讲师：**线程不安全的，但是效率高。  **讲师：**有的同学可能会有疑问ArrayList能否实现多线程安全呢？  **讲师：** 答案是肯定的，可以。  **讲师：**ArrayList可以使用工具类Collections.synchronized()同步方法来保证多线程安全。  **讲师：**所以从jdk 1.2后在不需要多线程安全的情况下主要使用ArrayList集合替代Vector集合。 | 1、以ppt文字讲解。  2、互动方式，学生占主导。  3、讲师伴随动作手势提醒。 |

1. **脚本切片六：[视频切片6](C:\\Users\\李巍\\AppData\\Roaming\\Microsoft\\Word\\1、课堂礼仪1‘.mp4)**

|  |  |  |
| --- | --- | --- |
| **课堂模块/时长** | **教学内容/台词** | **教学标准/素材** |
| **内容小结**  **(2.54)** | **1、内容小结：**  **讲师：**大家看ArrayList集合类特点是什么？  **学生：**省略。  **讲师：**那ArrayList集合类增，删，改查方法分别为？  **学生：**省略。  **讲师**：ArrayList和Vector之间有哪些区别呢？  **学生：省略**。 | 1、总结法。  2、互动法，学生主导。 |

1. **教学过程脚本设计—第三课时：**
2. **45分钟脚本设计思路：**



1. **每个脚本切片设计思路：**

|  |  |  |  |
| --- | --- | --- | --- |
| **内容** | **目标** | **设计亮点** | **时间** |
| 课程礼仪 | 提升学生气势，精神饱满，增加团队意识 | 突出学院特色，增加学生学习的信心，坚定学习信念 | 0.23 |
| 为什么使  用迭代器 | 启下，说明使用迭代器原因，明确本单元目标 | 代码分析--发现问题--解决问题 | 1.49 |
| 什么是  迭代器 | 通过迭代器概念分析，理解迭代器对象的产生 | 词源及概念分析--逐步引导--层层深入 | 3.54 |
| 迭代器应用 | 通过常用方法阐述，完成删除功能，深入理解迭代器应用及原理 | 流程图分析--代码演示--逐步引导 | 2.35+4.58 |
| 什么是ListIterator  迭代器及应用 | 阐述ListIterator迭代器定义及特点，了解ListIterator迭代器的简单应用 | 动画显示--对比法 | 2.58 |
| foreach循环  遍历集合 | 回顾foreach循环格式，阐述foreach循环应用，对比几种遍历输出方式的区别 | 对比法--代码演示--层层深入 | 6.25 |
| 什么是  Queue集合 | 通过定义阐述，引出Queue接口定义，引出实现类LinkedList集合类 | 过程演示--强调特点 | 2.10 |
| 内容小结 | 通过提问法Iterator的常用方法，为学生梳理知识点，将知识成体系 | 互动法--以学生为主导 | 2.46 |

1. **脚本切片一：[视频切片1](C:\\Users\\李巍\\AppData\\Roaming\\Microsoft\\Word\\1、课堂礼仪1‘.mp4)**

|  |  |  |
| --- | --- | --- |
| **课堂模块/时长** | **教学内容** | **教学标准** |
| **课堂礼仪**  **(0.23)** | 上课铃响后，  班长喊起立—学生问好—班级口号—鼓掌—整理着装—欢迎教师上课。 | 1.上课前，提前30秒面带自信笑容的走进教室，观察、问候学生们(等待上课)。  上课铃声响起！  2.老师检查学生们工装，整理工装速度及姿势。  IMG_256IMG_256 |
| **讲师：**上课 ！  **学生：**老师好！  **讲师：**同学们好！请坐。 | 1. 讲师严格检查学生整齐度和速度，保证上课状态，如凌乱需要重新做。  2. 学生礼仪内容：学院口号，班级口号，英文口号，鼓掌，整理工装，班长报出勤。  3. 讲师授课过程中要面带微笑（八颗牙齿）。  4. 插入合理手势，避免课堂枯燥。 |

1. **脚本切片二：[视频切片2](C:\\Users\\李巍\\AppData\\Roaming\\Microsoft\\Word\\1、课堂礼仪1‘.mp4)**

|  |  |  |
| --- | --- | --- |
| **课堂模块/时长** | **教学内容/台词** | **教学标准/素材** |
| **为什么使**  **用迭代器**  **(1.49)** | **1、为什么需要Iterator迭代器**  **讲师：**同学们，回忆上节课利用ArrayList集合存储狗狗对象。  **讲师：**并用contains()方法查找亚亚狗狗。  **讲师：**用集合remove()方法删除亚亚狗狗。  **讲师：**当时加了break语句，如果将break语句省略，会不会出现问题？  **学生：**会**。**  **讲师：**出现了什么问题？  **讲师：**是不是在主线程中出现正在修改异常（Exception in thread"main"java.util.ConcurrentModificationException）？  **学生：**是。  **讲师：**那如何解决呢？就要利用迭代器。  **讲师：**那什么是迭代器呢？让我们一起来学习。 | 1. 以ppt文字逐步引导。 2. 互动方式。 |

1. **脚本切片三：[视频切片3](C:\\Users\\李巍\\AppData\\Roaming\\Microsoft\\Word\\1、课堂礼仪1‘.mp4)**

|  |  |  |
| --- | --- | --- |
| **课堂模块/时长** | **教学内容/台词** | **教学标准/素材** |
| **什么是迭代器**  **(3.54)** | 1. **什么是迭代法及迭代器**   **讲师：**什么是迭代器？  **讲师：**如时钟在反复不停的转动。  **讲师：**讲迭代器前，先介绍什么是迭代法？  **讲师：**迭代法是对一系列指令反复执行；  **讲师：**每次执行都从迭代变量的原值推出新值的过程。  **讲师：**那什么是迭代器(Iterator)？  **讲师：**迭代器是一种使用迭代法遍历集合的对象。 | 1、讲师声调阴阳顿挫，讲重点和难点时声音加重。  2、生活中的案例引导，使容易理解迭代法概念。 |
| **2、什么是Iterable接口**  **讲师：**Iterator迭代器是如何产生的呢？  **讲师：** 这就需要用到java 集合框架的顶级接口Iterable接口。  **讲师：**此接口中的iterator()方法可以返回迭代器对象。  **讲师：**实现此接口的集合类便可通过迭代器遍历自身元素。  **讲师：**也就是可以进行迭代。 | 1、在讲解时同时打开API，查询Iterable接口的内容。  2、同时对应ppt上文字讲解。 |

1. **脚本切片四：[视频切片4](C:\\Users\\李巍\\AppData\\Roaming\\Microsoft\\Word\\1、课堂礼仪1‘.mp4)**

|  |  |  |
| --- | --- | --- |
| **课堂模块/时长** | **教学内容/台词** | **教学标准/素材** |
| **迭代器应用**  **(2.35+4.58)** | **1、迭代器常用方法**  **讲师：**下面一起学习Iterator接口的常用方法。  **讲师：**在Iterator接口中主要有hasNext()，next()和remove()这三个方法。  **讲师：**其中hasNext()表示判断是否还有其它元素可以迭代，如果有，则返回 true；  **讲师：**next()表示获取下一个元素；  **讲师：**remove()删除最近一次调用next()方法返回的元素。 | 1、讲解方法：参考API讲解常用方法。 |
| **2、迭代器使用步骤**  **讲师：**接下来学习如何使用迭代器遍历集合。  **讲师：**步骤是：第一步调用Iterator()得到一个指向集合序列第一个元素的迭代器；  **讲师：**第二用循环调用hasNext()方法，如果有元素，返回true；  **讲师：**第三在循环中，使用next()方法获得集合的下一个元素；  **讲师：**分析一下具体代码实现：  **讲师：**这里dogs为ArrayList集合对象**。**  **讲师：**首先调用集合对象的iterator()获得迭代器对象it。  **讲师：**其次用迭代器对象的hasNext()方法判断是否有其它元素可以迭代。  **讲师：**如果有，返回为true。  **讲师：**则在循环中遍历输出此元素。 | 1、采用实现步骤，逐步引导讲解。  2、讲到各种方法时声音提高，同时演示代码。 |
| **3、流程图与代码实现**  **讲师：**下面用迭代器解决课前提出省略break语句报错的问题。  **讲师：**先看一下流程图：  **讲师：**第一步调用Iterator()方法获得迭代器对象it；  **讲师：**调用hasNext()方法判断是否有下一个元素,如果有，返回true；  **讲师：**调用it.next()获得当前对象；  **讲师：**比较当前对象是否等于yaya狗狗对象,若等于，即为true；  **讲师：**删除亚亚狗狗对象。  **讲师：**看一下具体代码实现。  **讲师：**首先创建集合对象和亚亚狗狗对象，并将亚亚对象添加到集合。  **讲师：**调用集合对象的iterator()获得迭代器对象it。  **讲师：**用hasNext()方法判断是否有下一个元素。  **讲师：**如果有用next()获取当前元素**。**  **讲师：**判断当前对象的名称与亚亚狗狗名称是否相等**。**  **讲师：**如果相等，调用迭代器的remove()删除。  **讲师：**效果如何呢？  **讲师：**成功的打印了将亚亚狗狗对象信息删除。 | 1. 采用互动方式，逐步引导讲解。 2. 采用小组PK方式画流程图。 3. 在学生回答过程中老师以ppt效果展示。 |
| **4、删除原理分析**  **讲师：**那为什么调用迭代器的remove()就不报异常了？  **讲师：**下面一起分析删除集合元素的原理：  **讲师：**第一种用循环遍历：如图所示：  **讲师：**先使用size()方法获得集合长度为6。  **讲师：**即索引值为0到size()-1即为5。  **讲师：**当删除一个元素，后面元素将依次前移；  **讲师：**但还是会遍历到最大索引值为5的位置。  **讲师：**此处元素即为不确定的值，所以会报异常。  **讲师：**那怎么使用迭代器进行遍历？  **讲师：**迭代器是通过三个指针分别指向开头元素head；  **讲师：**当前元素current和结尾元素last。  **讲师：**如何实现删除呢?  **讲师：**此处current指针一直指向当前元素，如果删除某个元素；  **讲师：**则current自动指向下一个元素；  **讲师：**而last指针永远指向最后一个元素。  **讲师：**无论中间元素如何变动，首尾指针指向不变。  **讲师：**所以不会访问到无效区域，即不会出现异常。  **讲师：**因此一般集合都使用迭代器进行删除。 | 1. 采用ppt动画方式，逐步引导讲解。 2. 采用小组PK方式画流程图。 3. 在学生回答过程中老师以ppt效果展示。 |

1. **脚本切片五：[视频切片5](C:\\Users\\李巍\\AppData\\Roaming\\Microsoft\\Word\\1、课堂礼仪1‘.mp4)**

|  |  |  |
| --- | --- | --- |
| **课堂模块/时长** | **教学内容/台词** | **教学标准/素材** |
| **什么是ListIterator**  **迭代器及应用**  **(2.58)** | **1、ListIterator迭代器概念讲解**  **讲师：**接下来学习什么是ListIterator迭代器？  **讲师：**先看一下名词解释：  **讲师：**List英文list。  **讲师：**n.目录；名单；明细表的意思，动词列为表上；  **讲师：**这里意为列表，所以ListIterator迭代器就是列表迭代器。  **讲师：**那有什么特点呢？即内涵。 | 1、通过动画展示，语音要分明。 |
| **讲师：**允许双向遍历列表；  **讲师：**实现前向遍历;  **讲师：**有没有新增的方法呢？  **讲师：**有。  **讲师：**新增hasprevious()方法用于判断是否有前一个元素可以迭代，如果有返回true。  **讲师：**previous()方法获取前一个元素；  **讲师：**那如何应用的？ | 1、讲解时按照ppt动画提示。  2、采用自问自答方式完成。 |
| **2、ListIterator迭代器应用**  **讲师：**还是刚才的案例这里使用ListIterator迭代器进行遍历。  **讲师：**通过集合对象的listIterator()获得迭代器对象iter。  **讲师：**前面使用hasNext()方法和next()实现由前向后输出**。**  **讲师：**后面使用hasPrevious()和previous()方法实现由后向前输出。  **讲师：**这里用hasPrevious()判断是否有前面的元素；  **讲师：**用previous()方法输出前面的元素。 | 1、讲解时按照ppt代码演示。  2、重点强调对象创建和方法调用。 |

1. **脚本切片六：[视频切片6](C:\\Users\\李巍\\AppData\\Roaming\\Microsoft\\Word\\1、课堂礼仪1‘.mp4)**

|  |  |  |
| --- | --- | --- |
| **课堂模块/时长** | **教学内容/台词** | **教学标准/素材** |
| **foreach循环遍历集合(6.25)** | **1、foreach循环回顾：**  **讲师：**在遍历数组时使用过foreach循环。  **讲师：**回想一下的foreach语句格式。  **讲师：**格式为for(元素类型 元素变量x：遍历对象){输出x};  **讲师：**再想想有什么特点呢？  **学生：**简单，方便。  **讲师：**对，看一下具体特点：  **讲师：**它是java5的新特征；  **讲师：**for语句的特殊简化版本；  **讲师：**遍历数组，集合方便；  **讲师：**从特点中可以看出foreach除了可以遍历数组，还有遍历集合。  **讲师：**那如何遍历集合呢？  **讲师：**先分析一下代码**。**  **讲师：**用for关键字,后面是元素类型String，定义变量s**。**  **讲师：**后面输出集合名称为a。  **讲师：**循环中打印变量s。  **讲师：**从代码分析：  **讲师：**用foreach比用迭代器遍历方式确实简单方便。 | 1. 讲解时按照ppt动画提示，声音洪亮，把每一个知识点描述清楚。 2. 用迭代器遍历和foreach遍历对比讲解 |
| **2、集合遍历方式对比**  **讲师：**到目前为止我们总共学过几种遍历集合的方式？  **学生：**三种。  **讲师：**好，非常好！现在一起分析一下它们之间有什么区别？  **讲师：**现在分别用三种方式打印0-100000的数字。  **讲师：**第一种采用基本for循环输出**。**  **讲师：**第二种采用迭代器输出。  **讲师：**第三种采用foreach循环输出。  **讲师：**那么通过运行时间可以看出**。**  **讲师：**用时最短的是基本for循环，即效率最高。  **讲师：**其次是使用迭代器输出。  **讲师：**用时最多的是foreach循环，即效率最低。  **讲师：**实质foreach循环底层也是用迭代器实现的。  **讲师：**具体使用哪种遍历方式应该根据需求而定。 | 1、以ppt代码方式逐步引导。  2、互动方式，学生占主导。  3、讲师伴随动作手势提醒。 |

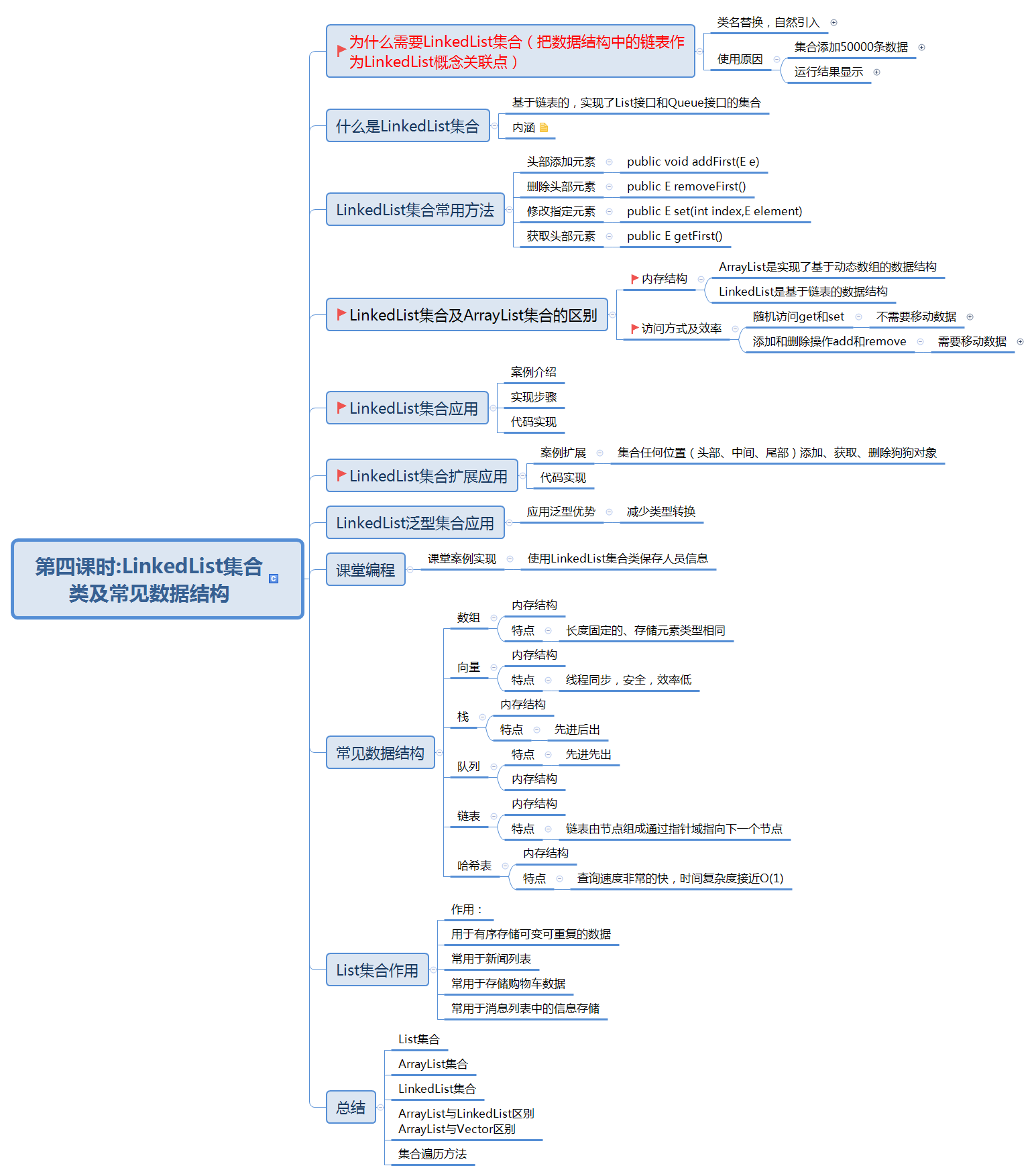
1. **脚本切片七：[视频切片7](C:\\Users\\李巍\\AppData\\Roaming\\Microsoft\\Word\\1、课堂礼仪1‘.mp4)**

|  |  |  |
| --- | --- | --- |
| **课堂模块/时长** | **教学内容/台词** | **教学标准/素材** |
| **什么是**  **Queue集合**  **(2.10)** | **1、什么是Queue集合**  **讲师：**大家回顾Collection集合下除了List集合还有什么集合？  **学生：**Queue集合。  **讲师：**好，那什么是Queue集合呢？  **讲师：**是用于有序存储可重复元素的、先进先出的集合。  **讲师：**那它是通过什么实现的呢?Queue接口。  **讲师：**那什么是Queue接口呢?  **讲师**：Queue接口是封装了队列功能的容器框架中的公共接口。  **讲师**：那什么又是队列呢？  **讲师**：队列是只允许在表的前端进行删除操作，而在表的后端进行插入操作的线性表。  **讲师**：即在队首进行出队，在队尾进行入队操作的线性表。  **讲师：**本质是一种很常见的[数据结构](http://lib.csdn.net/base/31" \o "算法与数据结构知识库" \t "https://blog.csdn.net/u010617952/article/details/_blank)。  **讲师：**这里需要强调内涵为先进先出。  **讲师**：它主要的实现类有LinkedList集合类，ArrayQueue和PriorityQueue集合类；  **讲师**：那么什么是LinkedList集合类我们下节课将给大家讲解。 | 1、讲解时用数据结构中的队列做为新知识的关联点。 |

1. **脚本切片八：[视频切片8](C:\\Users\\李巍\\AppData\\Roaming\\Microsoft\\Word\\1、课堂礼仪1‘.mp4)**

|  |  |  |
| --- | --- | --- |
| **课堂模块/时长** | **教学内容/台词** | **教学标准/素材** |
| **内容小结**  **(2.46)** | **1、内容小结：**  **讲师：**现在一起总结一下本节课内容：  **讲师：**Iterator的常用方法？  **学生：**省略。  **讲师：**遍历集合方式有几种？  **讲师：**它们有什么区别?  **学生**：省略。  **讲师：**那Queue类有哪些特点？  **学生**：省略。 | 1、总结法。  2、互动法，学生主导。 |

1. **教学过程脚本设计—第四课时：**
2. **45分钟脚本设计思路：**

****

1. **每个脚本切片设计思路：**

|  |  |  |  |
| --- | --- | --- | --- |
| **内容** | **目标** | **设计亮点** | **时间** |
| 为什么用LinkedList集合 | 通过和ArrayList集合类名互换，案例演示，明确区别，引出LinkedList集合 | 代码互换--明确区别--引出目标 | 2.31 |
| 什么是LinkedList  集合 | 通过List接口引入，以种差+属方式下定义，理解概念，掌握特点，理解常用方法 | 对比法--种差+属方式下定义--逐步引导 | 2.21 |
| ArrayList与LinkedList区别 | 通过内存结构分析区别，总结特点 | 对比法--画内存图分析--逐步引导 | 5.05 |
| LinkedList  集合应用 | 通过重点语句讲解，思路图分析，强化学生对LinkedList集合的应用 | 思路图--分析案例--案例流程 | 3.40 |
| LinkedList集合扩展应用及常见数据结构 | 通过扩展案例讲解，使学生熟练应用LinkedList集合 | 思路分析--代码演示--层层深入 | 2.16 |
| 贯穿案例及  内容小结 | 通过归纳法总结集合特点，为学生梳理思路，将知识体系化，提高学习效率 | 总结法--特点归纳法 | 4.57 |

1. **脚本切片一：[视频切片1](C:\\Users\\李巍\\AppData\\Roaming\\Microsoft\\Word\\1、课堂礼仪1‘.mp4)**

|  |  |  |
| --- | --- | --- |
| **课堂模块/时长** | **教学内容/台词** | **教学标准/素材** |
| **为什么用LinkedList集合(2.31)** | 1. **为什么需要LinkedList集合**   **讲师：**实现List接口的另一个实现类LinkedList集合类。  **讲师：**那什么是LinkedList集合呢？  **讲师：**现在将上午用ArrayList集合存储对象的代码改写成用LinkedList集合存储；  **讲师：**下面看一下运行结果有什么区别？  **学生：**没有区别。  **讲师：**对，答的非常好！输出结果相同。  **讲师：**那为什么还需要LinkedList集合呢？  **讲师：**再看一个案例在集合中填入50000条数据。  **讲师：**分别使用ArrayList和LinkedList集合进行添加和查找操作。  **讲师：**看运行结果有什么区别？  **讲师：**从运行结果可以看出ArrayList添加50000条耗时790ms，LinkedList添加50000条耗时17ms。  **讲师：**ArrayList查找50000条耗时2ms，LinkedList查找50000条耗时4ms。  **讲师：**可以看出LinkedList集合在添加操作耗时较少**。**  **讲师：**这就是学习LinkedList原因，在添加和删除操作中效率比较高。  **讲师：**那什么是LinkedList集合呢? | 1. 通过类名互换引出LinkedList集合类。 2. 互动方式，代码演示效果。 |

1. **脚本切片二：[视频切片2](C:\\Users\\李巍\\AppData\\Roaming\\Microsoft\\Word\\1、课堂礼仪1‘.mp4)**

|  |  |  |
| --- | --- | --- |
| **课堂模块/时长** | **教学内容/台词** | **教学标准/素材** |
| **什么是LinkedList**  **集合**  **(2.21)** | **1、什么是LinkedList集合**  **讲师：**什么是LinkedList集合呢？  **讲师：**还是用种差+属的方式下定义。  **讲师：**首先思考一下它临近的属应该是什么？  **学生**：集合。  **讲师：**好，非常好！再考虑同属的其它种有哪些？  **学生**：Vector，ArrayList集合。  **讲师：**好，非常好！那它们之间的区别？即种差。  **讲师：**回想ArrayList和Vector是不是都允许为空，并且可以通过索引访问，还允许重复？  **学生：**是**。**  **讲师：**因为它们同属List接口实现类**，**所以种差为首先允许为null；  **讲师：**并且可以通过索引访问；  **讲师：**可以重复的；  **讲师：**最本质的区别是LinkedList集合是基于链表实现。  **讲师：**以上即为内涵。  **讲师：**所以LinkedList集合的定义是基于链表的，实现了List接口和Queue接口的集合。 | 1. 以ppt文字逐步引导。 2. 互动方式。 |
| **2、LinkedList集合类常用方法讲解**  **讲师：**下面学习LinkedList集合的常用方法。  **讲师：**在LinkedList集合中实现了List接口的基本添加和删除方法。  **讲师：**这些方法的使用与ArrayList基本相同。  **讲师**：还增加了一些特有的方法，如：  **讲师**：addFirst（）和addLast（）分别从集合列表开头和结尾插入元素。  **讲师**：getFirst（）和getLast（）得到第一个和最后一个元素，返回这个元素。 **讲师**：removeFirst（）和removeLast（）移除第一个和最后一个元素，返回这个元素。  **讲师：**可见这些方法都是对首尾进行操作的。  **讲师：**原理是LinkedList集合底层是基于链表实现的。  **讲师：**插入和删除效率比较高。 | 1、采用互动方式，逐步引导讲解。  2、以API文档作参考，重点方法查找讲解。 |

1. **脚本切片三：[视频切片3](C:\\Users\\李巍\\AppData\\Roaming\\Microsoft\\Word\\1、课堂礼仪1‘.mp4)**

|  |  |  |
| --- | --- | --- |
| **课堂模块/时长** | **教学内容/台词** | **教学标准/素材** |
| **ArrayList与LinkedList**  **集合类区别**  **(5.05)** | **1、ArrayList集合分析**  **讲师：**那为什么插入和删除效率比较高？  **讲师：**接下来分析ArrayList集合与LinkedList集合的区别。  **讲师：**首先分析ArrayList集合，基于数组结构的集合。  **讲师：**遍历和随机访问元素的效率比较高。  **讲师：**什么原因呢？  **讲师：**如图所示：数组在内存是连续分配内存空间。  **讲师：**并且是顺序存储的，所以遍历和随机访问效率高。 | 1、讲师声调阴阳顿挫，讲遍历和随机时声音加重。  2、互动环节中讲师需关注每一个学生参与情况，发现问题所在。 |
| **2、LinkedList集合分析**  **讲师：**再来分析LinkedList集合：  **讲师：**LinkedList集合是采用链表存储方式。  **讲师：**插入，删除元素效率比较高。  **讲师：**什么原因呢？  **讲师：**如图一在单链表中插入节点c。  **讲师：**第一步将b节点的地址放入c节点的指针域。  **讲师：**使c节点指向b节点。  **讲师：**第二步将c节点地址s放入a的指针域。  **讲师：**使a节点指向c节点。  **讲师：**这样就将c节点插入到a,b节点中；  **讲师：**大家思考这里需要移动吗？  **讲师：**不需要；  **讲师：**而ArrayList集合中间插入是需要移动的，这个过程在之前分析过。  **讲师：**再看删除操作：  **讲师：**删除a,b,c节点中的b节点，直接将c的地址放入a指针域中即可。  **讲师：**这样就成功将b节点从a,b,c节点中删除了。  **讲师：**大家再思考这个过程需要移动吗？  **学生：**不需要。  **讲师：**所以ArrayList集合遍历和随机访问元素的效率比较高；  **讲师：**而插入，删除元素效率比较低。  **讲师：**LinkedList集合遍历和随机访问元素的效率比较低；  **讲师：**而插入，删除元素效率比较高。 | 1、在提问问题时，老师需要以思路图的形式展现。 |

1. **脚本切片四：[视频切片4](C:\\Users\\李巍\\AppData\\Roaming\\Microsoft\\Word\\1、课堂礼仪1‘.mp4)**

|  |  |  |
| --- | --- | --- |
| **课堂模块/时长** | **教学内容/台词** | **教学标准/素材** |
| **LinkedList**  **集合应用**  **(3.40)** | **1、案例分析**  **讲师：**接下来使用LinkedList集合实现如下案例。  **讲师：**案例介绍：  **讲师：**定义一个狗类，狗有：名字，种类等属性。狗具有吃啃骨和跑方法。  **讲师：**案例需求：  **讲师：**存储多条狗狗信息，获取第一条和最后一条，打印狗狗信息。  **讲师：**案例实现思路：  **讲师：**通过List接口实现类LinkedList实现该需求。  **讲师：**首先获得第一个元素。  **讲师：**其次获得第二个元素。  **讲师：**最后删除狗狗信息。  **讲师：**案例效果：打印第一条狗狗是欧欧。  **讲师：**打印最后一条狗狗昵称是美美。那如何实现呢？ | 1. 讲解方法：ppt及API演示。 2. 互动方式，以学生为主导。 |
| **2、实现步骤**  **讲师：**下面分析实现步骤：  **讲师：**首先定义狗类；  **讲师：**创建狗类对象yayaDog。  **讲师：**创建LinkedList集合对象dogs。  **讲师：**第二步将狗类对象添加到集合dogs中**。**  **讲师：**利用add()方法添加**。**  **讲师：**最后是打印输出。  **讲师：**分别调用getFirst()和getLast()输出第一个和最后一个元素名称。 | 1、采用互动方式，逐步引导讲解。  2、讲到删除第一个时声音提高。 |
| **3、代码演示**  **讲师：**下面分析代码实现：  **讲师：**首先创建LinkedList集合对象并把多个狗狗对象放入其中。**讲师：**如ououDog和yayaDog。  **讲师：**用getFirst()获得第一条狗狗信息。  **讲师：**用getLast()获得最后一条狗狗信息。  **讲师：**并且打印第一条和最后一条狗狗昵称。  **讲师：**最后用removeFirst()和removeLast()方法删除第一个和最后一条狗狗。  **讲师：**经代码分析，LinkedList集合在首尾操作是否比较方便？  **学生：**是。 | 1、采用互动方式，逐步引导讲解。  2、讲到删除第一个时声音提高。 |

1. **脚本切片五：[视频切片5](C:\\Users\\李巍\\AppData\\Roaming\\Microsoft\\Word\\1、课堂礼仪1‘.mp4)**

|  |  |  |
| --- | --- | --- |
| **课堂模块/时长** | **教学内容/台词** | **教学标准/素材** |
| **LinkedList集合扩展应用及常见数据结构**  **(2.16)** | **1、案例扩展**  **讲师：**下面在原有案例的基础上进行扩展。  **讲师：**实现在集合任何位置（头部、中间、尾部）添加、获取、删除狗狗对象。  **讲师：**运行效果：打印狗狗信息及成功将亚亚的信息删除。  **讲师：**案例分析：插入、删除操作频繁时，可使用LinkedList来提高效率。  **讲师：**LinkedList还额外提供头部和尾部元素进行添加和删除操作的方法。 | 1. 讲解方法在原案例上扩展，提高学生举一反三能力。 2. 学生练习，小组PK方式。 |
| **2、代码演示**  **讲师：**那如何实现呢？  **讲师：**下面分析代码：  **讲师：**首先创建LinkedList集合对象dogs。  **讲师：**在集合中应用泛型，传入类型参数Dog。  **讲师：**接下来创建狗狗对象并添加到集合。  **讲师：**下面调用集合对象的iterator()方法获得迭代器对象it。  **讲师：**用while循环遍历集合。  **讲师：**通过hasNext()方法判断是否有下一个元素。  **讲师：**并用next()获取当前元素。  **讲师：**查询是否存在亚亚狗狗对象，若存在调用迭代器的remove()方法删除。  **讲师：**大家思考这里为什么不调用集合对象的删除方法？  **讲师：**而调用迭代器删除方法呢？  **讲师：**此原理在ArrayList集合删除元素中讲解过。  **讲师：**最后输出狗狗信息及成功将亚亚的信息删除**。**  **讲师：**应用泛型后在获取当前对象时可以减少类型转换。 | 1、采用互动方式，逐步引导讲解。  2、讲到删除第一个时声音提高。 |
| **3、课程编程**  **讲师：**下面进行课程编程。  **讲师：**功能描述：  **讲师：**使用LinkedList完成保存人员信息的功能。  **讲师：**思路分析：  **讲师：**第一步创建四个人员名称；  **讲师：**第二步存入集合；  **讲师：**第三步输出显示；  **讲师：**运行效果：输出各人员信息。  **讲师**：现在给大家5分钟时间来完成。 | 1. 通过代码展示，语音要分明。 2. 学生练习，小组PK方式。 |
| **4、常见数据结构**  **讲师：**下面总结在集合中常见的数据结构。  **讲师：**数组在内存开辟一段连续的空间；  **讲师：**特点是类型固定，长度固定。  **讲师：**向量是一种动态数组，特点是线程同步，安全，效率低；  **讲师：**栈的内存中也是连续的，是一种特殊的线性表；  **讲师：**特点是先进后出；  **讲师：**只允许在一端插入和删除操作；  **讲师：**队列内存也是连续的，是一种特殊的线性表；  **讲师**：特点是先进先出；  **讲师**：在队尾插入，在队首删除的线程表；  **讲师：**链表内存是非连续，是非顺序存储的；  **讲师：**特点是由节点组成通过指针域指向下一个节点；  **讲师：**每一个节点都通过一个指针域指向下一个节点；  **讲师：**哈希表内存也是非连续，采用散列技术进行存储；  **讲师**：特点是查询速度非常快，时间复杂度接近O(1)； | 1、采用互动方式，逐步引导讲解。 |

1. **脚本切片六：[视频切片6](C:\\Users\\李巍\\AppData\\Roaming\\Microsoft\\Word\\1、课堂礼仪1‘.mp4)**

|  |  |  |
| --- | --- | --- |
| **课堂模块/时长** | **教学内容/台词** | **教学标准/素材** |
| **List集合作用(4.57)** | **1、作用：**  **讲师：**下面总结List集合的作用。  **讲师：**作用是用于存储有序可重复的数据。  **讲师：**如课上讲解案例：  网易新闻列表，京东购物车和QQ消息列表。  **讲师：**存储的数据都具有有序和可重复的特点。  **讲师：**这种数据就可以用List集合来进行存储。 | 1、互动法，学生主导。 |
| **2、使用场合 ：**  **讲师：**接下来分析使用场合：  **讲师：**如果要用队列，栈，链表，可以优先使用List集合。  **讲师：**当要求快速访问元素，则使用ArrayList 集合。  **讲师：**如要求快速插入和删除元素，则使用LinkedList集合。  **讲师：**当对多线程元素操作时，则使用Vector集合。  **讲师：**如果有要求后保存先读取数据，则使用Stack集合类。 | 1、总结法。  2、互动法，学生主导。 |

1. **脚本切片七：[视频切片7](C:\\Users\\李巍\\AppData\\Roaming\\Microsoft\\Word\\1、课堂礼仪1‘.mp4)**

|  |  |  |
| --- | --- | --- |
| **课堂模块/时长** | **教学内容/台词** | **教学标准/素材** |
| **内容小结**  **(4.57)** | **1、内容小结：**  **讲师：**List集合是通过什么实现的？  **学生：**省略。  **讲师：**那么List接口有哪几个实现类？  **学生：**省略。  **讲师：**那ArrayList集合类有什么特点？  **学生**：省略。  **讲师：**有哪些常用方法？  **学生：**省略。  **讲师：**那LinkedList集合类有什么特点？  **学生**：省略。  **讲师：**有哪些常用方法？  **学生：**省略。  **讲师：**ArrayList与LinkedList区别是什么？  **学生：**省略。  **讲师**：再回想ArrayList和Vector之间有哪些区别？  **学生：**省略。  **讲师：**最后遍历集合的方法有几种，分别是什么？ | 1、互动法，学生主导。 |
| **2、概念图总结：**  **讲师：**下面以概念图形式总结本单元内容。  **讲师：**本单元重点讲解了Collection集合。  **讲师：**及下面的List集合和Queue集合。  **讲师：**它们分别通过List接口和Queue接口实现。  **讲师**：List接口的实现类包括向量Vector和派生类Stack。  **讲师：**和基于数组实现的ArrayList集合。  **讲师：**和基于链表实现的LinkedList集合。  **讲师：**LinkedList集合同时也实现了Queue接口。  **讲师：**还简单介绍了Queue接口的另外两个实现类：PriorityQueue和ArrayQueue。  **讲师：**本单元需要重点掌握ArrayList和LinkedList集合的概念。  **讲师：**常用方法及基本操作，以及它们的区别。 | 1、互动法，学生主导。 |
| **3、课后作业：**  **讲师：**大家看一下这单元的作业是？  **讲师：**完成练习手册中简答题和编程题；  **讲师：**复习之前List集合特点。  **讲师：**ArrayList集合类的增删改查操作  **讲师：**以及使用迭代器及几种方式遍历集合。  **讲师**：还有LinkedList集合类的增删改查操作及教案中的预习内容。 | 1、总结法。  2、互动法，学生主导。 |