

**Henan University of Science and Technology**

Java程序设计基础实验

实验报告

学 院 软件学院

专业班级 软工2023

实验名称 Java高级

姓 名 闫文博

学 号 201451082337

一、实验目的

二、实验内容和结果(提交程序代码及注释)

1、实验内容1的程序代码

package 实验一01;  
  
import java.text.SimpleDateFormat;  
import java.util.GregorianCalendar;  
  
public class Car *{* String brand;  
 double price;  
 String color;  
 GregorianCalendar date = new GregorianCalendar*()*;  
 private int age;  
 SimpleDateFormat sdf = new SimpleDateFormat*(*"yyyy年MM月dd日 HH:mm:ss"*)*;  
   
 public Car*() {* super*()*;  
 *}* public Car*(*String brand, double price, String color, GregorianCalendar date*) {* super*()*;  
 this.brand = brand;  
 this.price = price;  
 this.color = color;  
 this.date = date;  
 *}* public int getAge*() {* return age;  
 *}* public void setAge*(*int age*) {* this.age = age;  
 *}* @Override  
 public String toString*() {* return "品牌：" + brand + "\n 价格：" + price + "万\n 颜色：" + color + "\n 销售日期：" + sdf.format*(*date.getTime*())* ;  
 *}  
   
   
   
  
}*

package 实验一01;  
  
import java.util.GregorianCalendar;  
  
public class CarDemo *{* public static void main*(*String*[]* args*) {* Car car1 = new Car*()*;  
 car1.brand="奥拓";  
 car1.color="土豪金";  
 car1.price = 3.5;  
 car1.date.set*(*2021,9,15,19,30,11*)*;  
   
 System.*out*.println*(*"--------------"*)*;  
 System.*out*.println*(*car1.toString*())*;  
 Car car2 = new Car*(*"奥迪",38.0,"黑色",new GregorianCalendar*())*;  
 car2.date.set*(*2021,9,15,19,30,11*)*;  
 System.*out*.println*(*car2.toString*())*;  
 *}  
}*

2、实验内容2的程序代码

package 实验一02;  
  
import java.util.ArrayList;  
import java.util.Iterator;  
  
public class ListDemo *{* public static void main*(*String*[]* args*) {* ArrayList*<*String*>* list = new ArrayList*<*String*>()*;  
 list.add*(*"hello"*)*;  
 list.add*(*"world"*)*;  
 list.add*(*"java"*)*;  
 Iterator*<*String*>* it = list.iterator*()*;  
 System.*out*.println*(*"---迭代器遍历方式"*)*;  
 while*(*it.hasNext*()) {* String s = *(*String*)* it.next*()*;  
 System.*out*.println*(*s*)*;  
 *}* System.*out*.println*(*"---增强for遍历方式"*)*;  
 for*(*String s:list*) {* System.*out*.println*(*s*)*;  
 *}* System.*out*.println*(*"---JDK8的forEach遍历方式"*)*;  
 list.forEach*(*o->*{* System.*out*.println*(*o*)*;  
 *})*;  
   
 *}  
}*

3、实验内容3的程序代码

package 实验一03;  
  
public class Student *{* String name;  
 int age;  
 public Student*(*String name, int age*) {* super*()*;  
 this.name = name;  
 this.age = age;  
 *}* @Override  
 public String toString*() {* return name + " \t"+age;  
 *}  
   
   
}*

package 实验一03;  
  
import java.util.ArrayList;  
import java.util.Iterator;  
import java.util.List;  
  
*/\*\*  
 \** ***@author*** *闫文博  
 \*/*public class StudentDemo *{* public static void main*(*String*[]* args*) {* List*<*Student*>* list = new ArrayList*<>()*;  
 list.add*(*new Student*(*"王伦",21*))*;  
 list.add*(*new Student*(*"王苑",18*))*;  
 list.add*(*new Student*(*"程宇",20*))*;  
 list.add*(*new Student*(*"毕胜",22*))*;  
 list.add*(*new Student*(*"张雪",19*))*;  
 Iterator it= list.iterator*()*;  
 while *(*it.hasNext*()){* Student s = *(*Student*)* it.next*()*;  
 System.*out*.println*(*s.toString*())*;  
 *}  
 }  
  
}*

1. 实验内容4的程序代码

package 实验一04;  
  
import java.util.LinkedList;  
import java.util.List;  
import java.util.Random;  
  
public class Random7 *{* public static void main*(*String*[]* args*) {* Random r = new Random*()*;  
 List*<*Integer*>* list = new LinkedList*()*;  
 int count = 0;  
 while*(*count <7*){* Integer a = r.nextInt*(*100*)*;  
 if*(*list.indexOf*(*a*)*==-1*){* list.add*(*a*)*;  
 *}* count++;  
 *}* for*(*int num:list*){* System.*out*.print*(*num+"\t"*)*;  
 *}  
 }  
}*

5、实验内容5的程序代码

package 实验一05;  
*/\*\*  
 \** ***@author*** *闫文博  
 \*/*public class Student *{* String name;  
 int age;  
 public Student*(*String name, int age*) {* super*()*;  
 this.name = name;  
 this.age = age;  
 *}* @Override  
 public String toString*() {* return name + "\t\t"+age;  
 *}  
  
  
}*

package 实验一05;  
  
import java.util.ArrayList;  
  
public class ArrayListDemo *{* public static void main*(*String*[]* args*) {* ArrayList*<*Student*>* smallArray = new ArrayList*<>()*;  
 smallArray.add*(*new Student*(*"貂蝉",19*))*;  
 smallArray.add*(*new Student*(*"大乔",23*))*;  
 smallArray.add*(*new Student*(*"小乔",18*))*;  
 smallArray.add*(*new Student*(*"孙尚香",26*))*;  
  
 ArrayList*<*Student*>* smallArray2 = new ArrayList*<>()*;  
 smallArray2.add*(*new Student*(*"孙二娘", 25*))*;  
 smallArray2.add*(*new Student*(*"扈三娘", 27*))*;  
 smallArray2.add*(*new Student*(*"李师师", 17*))*;  
 smallArray2.add*(*new Student*(*"阎婆惜", 36*))*;  
  
 ArrayList*<*Student*>* smallArray3 = new ArrayList*<>()*;  
 smallArray3.add*(*new Student*(*"林黛玉", 19*))*;  
 smallArray3.add*(*new Student*(*"王熙凤", 35*))*;  
 smallArray3.add*(*new Student*(*"薛宝钗", 20*))*;  
 smallArray3.add*(*new Student*(*"袭人", 21*))*;  
  
 System.*out*.println*(*"--------增强for方法-----"*)*;  
 ArrayList*<*ArrayList*<*Student*>>* bigArray = new ArrayList*<>()*;  
 bigArray.add*(*smallArray*)*;  
 bigArray.add*(*smallArray2*)*;  
 bigArray.add*(*smallArray3*)*;  
  
 for*(*ArrayList*<*Student*>* a:bigArray*){* for*(*Student s:a*){* System.*out*.println*(*s.toString*())*;  
 *}* System.*out*.println*(*"---------------"*)*;  
 *}* System.*out*.println*(*"----------forEach方法------------"*)*;  
 bigArray.forEach*(* o->*{* o.forEach*(* s->*{* System.*out*.println*(*s.toString*())*;  
 *}  
 )*;  
 System.*out*.println*(*"---------------"*)*;  
 *}  
 )*;  
  
 *}  
}*

6、实验内容6的程序代码

、package 实验一06;  
  
import java.util.Objects;  
  
*/\*\*  
 \** ***@author*** *闫文博  
 \*/*public class Student *{* String name;  
 int age;  
 public Student*(*String name, int age*) {* super*()*;  
 this.name = name;  
 this.age = age;  
 *}* @Override  
 public String toString*() {* return name + "\t\t"+age;  
 *}* @Override  
 public boolean equals*(*Object o*) {* if *(*this == o*) {* return true;  
 *}* if *(*o == null || getClass*()* != o.getClass*()) {* return false;  
 *}* Student student = *(*Student*)* o;  
 return age == student.age && Objects.*equals(*name, student.name*)*;  
 *}* @Override  
 public int hashCode*() {* return Objects.*hash(*name, age*)*;  
 *}  
}*

package 实验一06;  
  
import java.util.HashSet;  
  
public class HashSetDemo *{* public static void main*(*String*[]* args*) {* HashSet*<*Student*>* hashSet = new HashSet*<>()*;  
 hashSet.add*(*new Student*(*"林志玲", 27*))*;  
 hashSet.add*(*new Student*(*"王祖贤", 21*))*;  
 hashSet.add*(*new Student*(*"张曼玉", 25*))*;  
 hashSet.add*(*new Student*(*"关之琳", 29*))*;  
 hashSet.add*(*new Student*(*"林志玲", 27*))*;  
 hashSet.add*(*new Student*(*"林志玲", 20*))*;  
  
 for*(*Student s:hashSet*){* System.*out*.println*(*s.toString*())*;  
 *}  
  
 }  
}*

7、实验内容7的程序代码

package 实验一07;  
  
*/\*\*  
 \** ***@author*** *闫文博  
 \*/*public class Student implements Comparable*<*Student*> {* String name;  
 int age;  
 public Student*(*String name, int age*) {* super*()*;  
 this.name = name;  
 this.age = age;  
 *}* @Override  
 public String toString*() {* return name + " \t"+age;  
 *}* @Override  
 public int compareTo*(*Student o*) {* return this.age-o.age;  
 *}  
}*

package 实验一07;  
  
import java.util.TreeSet;  
  
public class HashSetDemo *{* public static void main*(*String*[]* args*) {* TreeSet*<*Student*>* students = new TreeSet*<>()*;  
 students.add*(*new Student *(*"linqingxia", 27*))*;  
 students.add*(*new Student *(*"zhangguorong", 29*))*;  
 students.add*(*new Student *(*"wanglihong", 23*))*;  
 students.add*(*new Student *(*"linqingxia", 27*))*;  
 students.add*(*new Student *(*"liushishi", 22*))*;  
 students.add*(*new Student *(*"wuqilong", 40*))*;  
 students.add*(*new Student *(*"gaoyuanyuan", 22*))*;  
  
 for*(*Student s:students*){* System.*out*.println*(*s.toString*())*;  
 *}  
 }  
}*

8、实验内容8的程序代码

package 实验一08;  
  
public class Student *{* String name;  
 int age;  
 public Student*(*String name, int age*) {* super*()*;  
 this.name = name;  
 this.age = age;  
 *}* @Override  
 public String toString*() {* return name + " \t"+age;  
 *}  
  
  
}*

package 实验一08;  
  
import java.util.Comparator;  
import java.util.TreeSet;  
  
public class HashSetDemo *{* public static void main*(*String*[]* args*) {* TreeSet*<*Student*>* students = new TreeSet*<>(*new Comparator*<*Student*>() {* @Override  
 public int compare*(*Student o1, Student o2*) {* return o1.name.length*()*-o2.name.length*()*;  
 *}  
 })*;  
  
 students.add*(*new Student *(*"linqingxia", 27*))*;  
 students.add*(*new Student *(*"zhangguorong", 29*))*;  
 students.add*(*new Student *(*"wanglihong", 23*))*;  
 students.add*(*new Student *(*"linqingxia", 27*))*;  
 students.add*(*new Student *(*"liushishi", 22*))*;  
 students.add*(*new Student *(*"wuqilong", 40*))*;  
 students.add*(*new Student *(*"gaoyuanyuan", 22*))*;  
 students.add*(*new Student*(*"linqingxia", 29*))*;  
  
 for*(*Student s:students*){* System.*out*.println*(*s.toString*())*;  
 *}  
 }  
}*

9、实验内容9的程序代码

package 实验一09;  
  
import java.util.HashSet;  
import java.util.Random;  
  
public class HashSetRandom *{* public static void main*(*String*[]* args*) {* Random r = new Random*()*;  
 HashSet*<*Integer*>* hashset = new HashSet*<>()*;  
 while *(*hashset.size*()*<7*){* hashset.add*(*r.nextInt*(*100*))*;  
 *}* for*(*int num:hashset*){* System.*out*.print*(*num+"\t"*)*;  
 *}  
 }  
}*

三、思考题。

1、什么是集合，请列举集合中常用的类和接口？

集合：存储一个元素集合的容器

常见类：ArrayList、LinkedList、HashSet、HashMap

常用接口:Collection、List、Set、Map

2、请简述TreeSet集合保证唯一性的两种方式

自然顺序:TreeSet类的add()方法中会把存入的对象提升为Comparable类型

调用对象的compareTo()方法和集合中的对象比较(当前存入的是谁,谁就会调用compareTo方法)根据compareTo()方法返回的结果进行存储

比较器顺序:创建TreeSet的时候可以制定 一个Comparator如果传入了Comparator的子类对象, 那么TreeSet就会按照比较器中的顺序排序add()方法内部会自动调用Comparator接口中compare()方法排序调用的对象(就是当前存入的对象)是compare方法的第一个参数,集合中的对象(已经添加进去的对象)是compare方法的第二个参数

3、Java语言中，List的实现类ArrayList、Vector、LinkedList特点是什么？

ArrayList:底层数据结构是数组，查询快，增删慢线程不安全，效率高

Vector:底层数据结构是数组，查询快，增删慢线程安全，效率低（使用率少）

LinkedList:底层数据是双向循环链表，查询慢，增删快线程不安全，效率高，

4、Java语言中，使用泛型有什么好处？

将类型转换报错从运行期提前到编译期，便于发现bug

四、实验总结与体会。（若无可省略）