|  |  |  |  |
| --- | --- | --- | --- |
| **学号** | 2113041111 | **姓名** | 乔嘉琛 |
| **专业** | 软件工程 | **日期** | 2024.8.14 |
| **实训任务：第五章 面向对象基础**   1. 谈谈你对面向对象和面向过程的理解。   答：  OOP（object oriented programming），面向对象编程 是一种以对象为中心的编程思想，通过借助对象实现具体的功能 将大问题拆分成小问题，然后借助不同对象分别解决，最终实现功能  POP（procedure oriented Programming），面向过程编程 是一种以过程为中心的编程思想，靠自己一步一步去实现功能，需要对每个步骤精确控制 强调按步骤实现功能，先分析解决问题所需步骤，再自定义方法实现每个步骤功能，然后依次调用 方法，最终实现功能   1. 设计一款五子棋游戏：请分别用面向过程和面向对象思想实现，请大致描述你的思路。 2. 答： 3. 面向过程:   初始化棋盘：定义一个二维字符数组来表示棋盘，初始化为空。  绘制棋盘：编写一个方法，用于在控制台输出当前棋盘的状态。  获取玩家输入：编写一个方法，用于接收玩家的输入，并 处理成棋盘上的坐标.  检查落子合法性：编写一个方法，验证玩家输入的坐标是否有效。  更新棋盘：编写一个方法，根据玩家的落子更新棋盘。  检查胜利条件：编写一个方法，检查是否有玩家获胜。  游戏主循环：编写一个主方法，包含游戏的主要逻辑，如交替落子、检查游戏结束条件等。  面向对象:   1. Board 类：负责棋盘的初始化、显示、落子和检查胜利条件。 2. Player 类：代表一个玩家，包含玩家的符号（‘X’ 或 ‘O’）和落子方法。 3. Game 类：控制游戏的流程，包括初始化游戏、轮流让玩家落子、检查游戏是否结束等。   3. 谈谈你对类、对象、引用的理解。  答：  类是对现实生活中一类具有共同属性和行为的事物的抽象 类是对象的数据类型，类是具有相同属性和行为的一组对象的集合 简单理解：类就是对现实事物的一种描述 类是引用数据类型中的一种  对象是类的实例，是实际存在的实体。在内存中，对象占用一定的空间，并具有类定义的属性和行为。通过创建对象，可以操作类的属性和方法，实现具体的功能。  引用是一个变量，用于指向（引用）内存中的对象。在Java中，引用变量存储的是对象的内存地址，而不是对象本身。通过引用，可以访问对象的属性和方法。  4. 根据下面案例，绘制对象内存图    答：    5. 需求描述：  A）定义一个汽车类 Car  属性有：颜色、品牌、价格  行为有：前进、后退、转弯和输出对象属性  B）测试类实现功能：  实例化一个Car类对象，并给元素赋值  调用这个类中方法，输出对象属性信息  调用对象前进、后退、转弯等方法  C）命名规则注意事项：  属性、方法：若只有一个单词，全小写  若多个单词构成，第一个单词全小写，后面单词首字母大写    答：      6. 谈谈你对封装的理解。  答：  Java封装是一种面向对象编程的概念，它指的是将对象的实现细节隐藏起来，仅对外暴露有限的接口以进行交互。简而言之，封装就是数据隐藏。  以下是Java封装的简述：  隐藏实现细节：通过将类的成员变量声明为私有的（private），可以阻止直接从类外部访问这些变量。  提供公共接口：通过公共（public）方法（通常称为getter和setter方法），允许外部代码安全地访问和修改对象的内部状态。  增加安全性：可以在setter方法中添加逻辑来验证输入数据的有效性，从而保护对象状态的一致性。  提高可维护性：由于内部实现细节被隐藏，因此在需要修改内部实现时，不会影响到使用该类的其他代码。  7. 列举构造方法注意事项。  答：  构造方法一般使用 public 修饰  构造方法没有返回值类型，连 void 都没有  构造方法名和类名相同（区分大小写）  构造方法可以重载  8. 构造方法编程  定义一个人类Person  属性： 姓名 （name） 、年龄 （age）；  要求：  A） 定义无参构造器  B） 定义两参构造器，参数为name和age  C） 定义一参构造器，参数为name，要求该构造器借助借助两参构造器实现功能  D） 定义show方法，输出类对象基本信息    答：    9. 简述this关键字的用法。  答：  对成员变量和局部变量进行区分 固定格式： this. 数据成员 ;  调用类中的成员方法 固定格式： this. 成员方法 ( 实际参数列表 );  调用类中的其他构造器  10. 下面代码无法通过编译，要求：  A） 修改代码使得程序能够正常运行  B） 分析其输出结果  提示：该题主要用于考核this关键字的作用      答：    11 根据要求编写时间类Duration  时间类Duration具体要求如下：  A) 包含属性hour、min、sec，都是int类型  B) 提供无参构造器  C) 提供构造器（三参），对hour、min、sec进行初始化  D) 提供构造器（一参），参数含义：总秒数int seconds，对hour、min、sec进行初始化，例如3661秒，转为a小时b分钟c秒，就是1小时1分钟1秒  E) 提供每个属性的get方法  F) 提供getTotalSeconds()方法，用于返回hour小时min分钟sec秒钟对应的总秒数seconds  G) 提供disp()方法，输出对象的属性信息和总秒数  测试类：    答：      12 扩展题：  现有一个简单的学生管理系统，系统启动后运行效果如下图：  系统启动后提示界面如下图：  image-20220907075941682  添加学生操作步骤如下：  企业微信截图_16625086835763  查看学生操作步骤如下：  image-20220907080149193  更新学生操作步骤如下：  image-20220907080356738  删除学生操作步骤如下：  image-20220907080621488  其中学生类Student，学生管理类 StudentManagement代码已经给出，但代码不完整！  请补全代码，使得程序能够实现完整上述完整功能：  **package** com.briup.chap05;  **import** java.util.Scanner; *//学生管理系统类* **public class** StudentManagement{  *//学生对象数组* **private** Student[] stus;  *//实际学生个数* **private int** counter;  *//键盘录入对象* **private** Scanner in;  *//无参构造器：默认容量100* **public** StudentManagement(){  stus = **new** Student[100];  in = **new** Scanner(System.in);  }  *//有参构造器：size表示初始容量* **public** StudentManagement(**int** size){  stus = **new** Student[size];  in = **new** Scanner(System.in);  }  *//输出 操作提示* **public void** prompt(){  System.out.println(**"\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*"**);  System.out.println(**"\*Student Management\*"**);  System.out.println(**"\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*"**);  System.out.println(**"\* 1)Add"**);  System.out.println(**"\* 2)Delete"**);  System.out.println(**"\* 3)Update"**);  System.out.println(**"\* 4)Select"**);  System.out.println(**"\* 0)Quit"**);  System.out.println(**"\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*"**);  System.out.print(**"Please choose your operate:"**);  }  *//根据用户录入的学生名称，找到学生在数组中的索引返回* **private int** findStudent(){  **int** index = -1;  String name;  System.out.print(**"Please input name:"**);  *//从键盘获取一个字符串* name = in.next();  *//请补全下面核心代码  //...* **return** index;  }  *//下面就是增删改查 四个方法* **public void** addStudent(){  String name;  **int** age;  String gender;  Student s;  *//判断 管理系统容量是否 已满* **if**(counter >= stus.length){  System.out.println(**"Add failure,too many student!"**);  *//扩容 数组拷贝，System.方法 Arrays.方法* **return**;  }  System.out.print(**"Please input name:"**);  name = in.next();  System.out.print(**"Please input age:"**);  age = in.nextInt();  System.out.print(**"Please input gender:"**);  gender = in.next();  s = **new** Student(name,age,gender);  *//关键代码* stus[counter] = s;  counter++;  System.out.println(**"Add student success!"**);  }  *//请补全下面代码* **public void** deleteStudent(){}  **public void** updateStudent(){}  **public void** selectStudent(){}  **public static void** main(String[] args){  *//实例化对象* StudentManagement sm = **new** StudentManagement();  *//用户操作标志* **int** option = -1;  Loop: **while**(**true**){  *// 打印提示信息* sm.prompt();  *//获取用户操作* option = sm.in.nextInt();  **if**(option < 0 || option > 4)  **continue**;  **switch**(option){*//byte char short int* **case** 1:  sm.addStudent();  **break**;  **case** 2:  sm.deleteStudent();  **break**;  **case** 3:  sm.updateStudent();  **break**;  **case** 4:  sm.selectStudent();  **break**;  **case** 0:  **break** Loop;  }*//end switch* }*// end while* System.out.println(**"游戏结束, byebye"**);  }*//end main* }*//end class* **class** Student {  *// 属性* **private** String name;  **private int** age;  **private** String gender;  *//默认|无参构造器  //有参构造器  //get|set方法* }  答： | | | |