1. 以下哪个选项是可以在dos命令下返回上一级目录 A
2. cd.. 返回上一级
3. dir 查看当前目录里面所有的文件和子目录
4. cd / 返回到根目录
5. cls 清屏
6. 以下哪一个是死循环 C
7. for(int i = 0 ; i<100;i++)
8. for(int i = 0 ; i >100;i--)
9. for(;;)
10. for(int i = 100; i=100;i++)

for(;;){}

while(true){}

do{}while(true)

1. 以下变量名定义正确是 A
2. osz
3. 1wer 以数字开头了
4. #es 出现了不允许出现的字符 #
5. class 关键字

标识符的命名规则：

只能是数字、下划线、美元符号、字母(区分大小写)

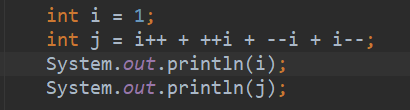
不能是关键字，不能以数字开头，小驼峰

1. 以下不能自动提升为int类型的是 D
2. byte
3. short
4. char
5. long

byte short int long float double char

字符参与+运算 会自动提升为int [byte和short也是如此] 【忘记的同学参考第二天的运算符】

1. 下面程序运行结果i和j的值分别为 。。。。。。 C



1. 1,1
2. 2.1
3. 1,8
4. 2,8
5. 以下数组定义正确的是 B
6. int arr = new int[10]; 等号左侧少了[]
7. int ar[] = new int[10];
8. int arr[] = new int[10]{1,2,3,4,5,67,8,9,0,8}; 要么去掉长度，要么去掉元素
9. int arr[] = int[10]; 等号右侧少了关键字new

数组有三种初始化方式：

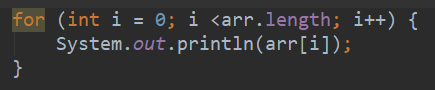
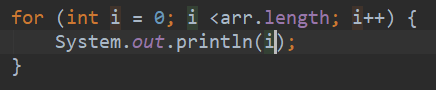
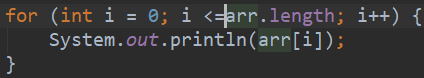
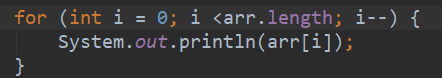
动态初始化：

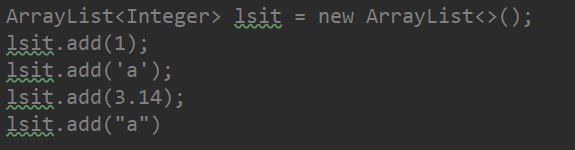
数据类型[] 变量名称 = new 数据类型[数组的长度];

数据类型 变量名称[] = new 数据类型[数组的长度];

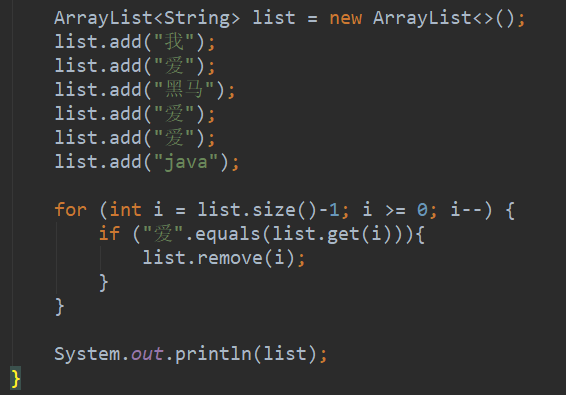
标准静态初始化：数据类型[] 变量名称 = new 数据类型[]{元素1,元素2,元素3,…};

简化静态初始化：数据类型[] 变量名称 = {元素1,元素2,元素3,…};

1. 以下关于数组String arr[] = new String[10] 说法正确的是 C
2. arr[10]=null; 最大索引值为9
3. arr[10]=0; 最大索引值为9
4. arr[0]=null;
5. arr[0]=0; 数组里面存放的元素数据的类型是String，默认值是null
6. 下列遍历数组正确的是 A
7. 
8. 遍历打印索引值
9. 条件判断表达式错误，角标越界问题
10. 条件控制语句出错！
11. 下列语句正确的是



1. 第二行
2. 第三行
3. 第四行
4. 第五行
5. 关于ArrayLsit对象方法描述不正确的是
6. 可以用过add（）方法完成元素添加
7. 可以通过remove（）方法完成元素删除
8. 可以通过set()方法完成指定索引下元素的替换
9. 可以通过length()获取数组长度
10. 下列代码的输出结果为



1. 编译错误
2. [我，黑马，爱，java]
3. [我，黑马，java]
4. [我，爱，黑马，爱，java]
5. 下列属于方法重载的是 D
6. int method(){},int method1(){} 方法名称不一致
7. int method(){},String method(){} 参数列表也相同了
8. int method(int a ,int b){},int method(int x,int y){} 参数列表也相同了
9. int method(){},int method(int a){}

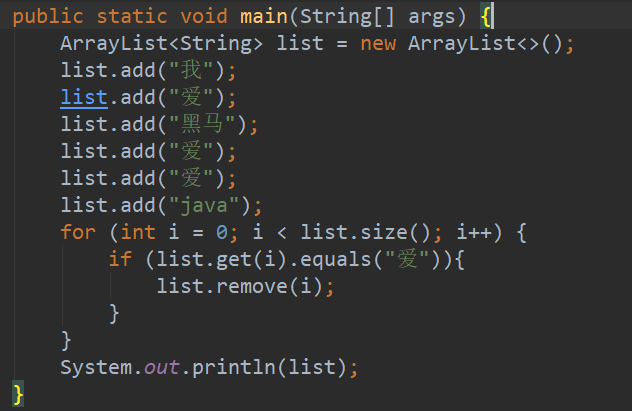
重载的要求：

多个方法在同一个类中

方法名称必须完全一致！

参数列表不一致(个数、类型)

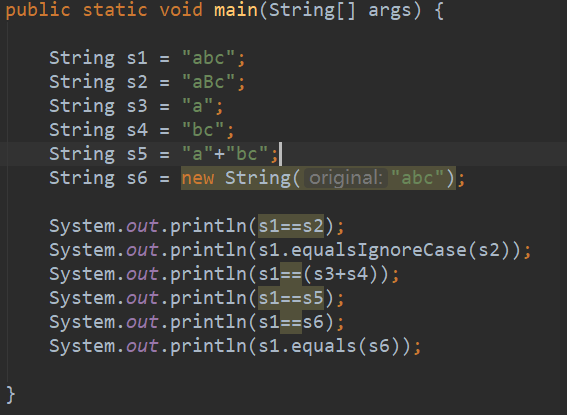
1. 以下代码的运行结果为



1. [我，爱，黑马，爱，爱，java]
2. [我，黑马，爱，java]
3. [我，黑马，java]
4. [我，爱，黑马，爱，java]
5. 构造方法何时被调用 B
6. 类定义时
7. 创建对象时
8. 调用对象方法时
9. 使用对象变量时
10. 下面关于类与对象描述错误的是 B
11. 对象是一个类的实例
12. 类是所有事物的模板，用事物属性特征和行为来描述所有事物 【具有相同属性和行为的同一类事物】
13. 类是对象的模板，对象是类的实体
14. 对象是一类事物的具体体现，必然具备该事物的属性和行为
15. 下列关键字属于权限修饰符的是 A B
16. private
17. public
18. static
19. void

权限修饰符有4个： public protected default private

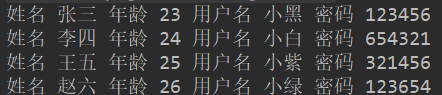
1. 下列说法错误的是 ACD
2. do while 可以一次也不执行 至少要执行一次！
3. while 可以无限循环 只要条件表达式为true
4. continue 和 break一样都是都是结束循环，执行下一条语句 【continue结束本次循环，进入下次循环】
5. for循环不可以嵌套使用 可以的！
6. 以下成员变量和局部变量，说法正确的是 ABD
7. 在类中的位置：成员变量 ：类中，方法外 局部变量：方法中或方法声明上
8. 初始化值：成员变量：有默认值，局部变量：没有默认值 必须先定义，赋值在使用
9. 内存中的位置:成员变量：栈内存，局部变量：堆内存 [说反了]
10. 作用范围：成员变量：类中 局部变量：方法中
11. 关于集合描述正确的是
12. 集合只能存储基本数据类型
13. 集合只能存储引用数据类型
14. 集合的长度不可变
15. 集合的长度可变
16. 下列代码运行结果为



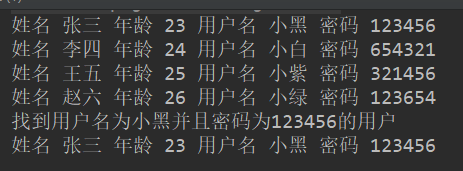
1. false true false true false true
2. true true false true false true
3. false false false true false true
4. false true false true false false
5. 定义数组 int arr [] = new int[10]需求如下
   1. 定义方法getArray()完成对数组的初始化，要求采用随机数，范围为1-100（包含1和100）
   2. 定义方法printArray遍历数组；
   3. 定义方法getIndex() 完成 通过键盘录入一个数字2-5（包含2和5，不需要做录入验证），如果数组中存在键盘录入数字的整倍数，那么打印数组中的数字
   4. 定义方法，getAv()获取数组的平均值，将结果打印在控制台

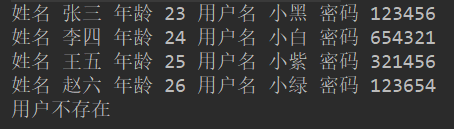
|  |
| --- |
| 1. **package** com.itheima;  **import** java.util.Random; **import** java.util.Scanner;  **public class** Test01 {  **public static void** main(String[] args) {  */\*  21. 定义数组 int arr [] = new int[10]需求如下  (1) 定义方法getArray()完成对数组的初始化，要求采用随机数，范围为1-100（包含1和100）  (2) 定义方法printArray遍历数组；  (3) 定义方法getIndex() 完成 通过键盘录入一个数字2-5（包含2和5，不需要做录入验证），如果数组中存在键盘录入数字的整倍数，那么打印数组中的数字  (4) 定义方法，getAv()获取数组的平均值，将结果打印在控制台   \*/* **int**[] array = *getArray*();  *printArray*(array);  *getIndex*(array);  **int** avg = *getAv*(array);  System.***out***.println(avg);  }    *//生成数组* **public static int**[] getArray(){  **int** arr [] = **new int**[10];  Random random = **new** Random();   **for** (**int** i = 0; i < arr.**length**; i++) {  arr[i] = random.nextInt(4) + 2;  }  **return** arr;  }   *//遍历数组* **public static void** printArray(**int**[] arr){  **for** (**int** i = 0; i < arr.**length**; i++) {  System.***out***.println(arr[i]);  }  }   **public static void** getIndex(**int**[] arr){  Scanner scanner = **new** Scanner(System.***in***);  System.***out***.println(**"请输入一个数字(2-5)"**);  **int** i = scanner.nextInt();  **for** (**int** i1 = 0; i1 < arr.**length**; i1++) {  **if** (arr[i1] % i == 0){  System.***out***.println(arr[i1]);  }  }   }   **public static int** getAv(**int**[] arr){  **int** sum = 0;  **for** (**int** i = 0; i < arr.**length**; i++) {  sum += arr[i];  }  **return** sum/arr.**length**;  } } |

1. 定义用户类 User 包含以下属性 姓名 name（String），年龄 age（int），用户名username（String），密码 password（int） 完成以下功能
   1. 定义User方法以及其属性，完成get/set方法和空参有参构造
   2. 创建测试类Test，完成以下功能
      1. 创建4个User对象，并初始化值
      2. 将4个对象存到List集合中并遍历打印，如图



* + 1. 查找集合中是否存在用户名为 “小黑”，并且密码为“123456”的人，如果有请打印对应的用户，结果如图一，如果不存在打印结果如图二；





|  |
| --- |
| **package** com.itheima;  **public class** User {  **private** String **name**;  **private int age**;  **private** String **username**;  **private int password**;   **public** User(String name, **int** age, String username, **int** password) {  **this**.**name** = name;  **this**.**age** = age;  **this**.**username** = username;  **this**.**password** = password;  }   **public** User() {  }   **public** String getName() {  **return name**;  }   **public void** setName(String name) {  **this**.**name** = name;  }   **public int** getAge() {  **return age**;  }   **public void** setAge(**int** age) {  **this**.**age** = age;  }   **public** String getUsername() {  **return username**;  }   **public void** setUsername(String username) {  **this**.**username** = username;  }   **public int** getPassword() {  **return password**;  }   **public void** setPassword(**int** password) {  **this**.**password** = password;  }   @Override  **public** String toString() {  **return "User{"** +  **"name='"** + **name** + **'\''** +  **", age="** + **age** +  **", username='"** + **username** + **'\''** +  **", password="** + **password** +  **'}'**;  } }  **package** com.itheima;  **import** java.util.ArrayList;  **public class** Test02 {  **public static void** main(String[] args) {  User user = **new** User(**"张三"**,23,**"小黑"**,123456);  User user1 = **new** User(**"李四"**,24,**"小白"**,654321);  User user2 = **new** User(**"王五"**,25,**"小绿"**,321456);  User user3 = **new** User(**"赵六"**,26,**"小紫"**,123456);   ArrayList<User> list = **new** ArrayList<>();  list.add(user);  list.add(user1);  list.add(user2);  list.add(user3);   **for** (**int** i = 0; i < list.size(); i++) {  User u = list.get(i);  System.***out***.println(**"姓名 "** + u.getName() + **" 年龄 "** + u.getAge() +  **" 用户名 "** + u.getUsername() + **" 密码 "** + u.getPassword());   }   **for** (**int** i = 0; i < list.size(); i++) {  User user4 = list.get(i);  **if**(**"小黑"**.equals(user4.getUsername()) && user4.getPassword() == 123456){  System.***out***.println(**"找到用户名为小黑并且密码为123456的用户"**);  System.***out***.println(**"姓名 "** + user4.getName() + **" 年龄 "** + user4.getAge() +  **" 用户名 "** + user4.getUsername() + **" 密码 "** + user4.getPassword());  **return**;  }  }  System.***out***.println(**"用户不存在"**);      } } |