第一阶段基础理论

**一．选择题**

1. 为了区分重载多态中同名的不同方法，要求（A）。

A） 形式参数个数或者类型不同        B） 返回值类型不同

C） 调用时用类名或对象名做前缀       D） 形式参数名称不同

1. 下面定义数组的格式中正确的是(多项)：（C E）

A．int a[10]    B。int a=new int[10]

C．int []a=new int[5]   D.int a[]

E. int []a={1,2}

1. 下面说法中不正确的是：（C）

A．类是对象的抽象，对象是类的实例 B。类是组成java程序的最小的单位

C．java语言支持多继承    D。java一个程序中只能有一个public类

1. 定义类时，不可能用到的保留字是（A）。

A） private    B） class

C） extends                     D） implements基本类型总共有多少种，并写出三到四个常用数据类型。

1. 为 AB 类的定义一个公共的构造函数，该方法头的形式为（D ）

A．void AB( )        B。public void method( )

C．public  method ( )   D。public  AB( )

1. 下面说法中不正确的是：（A ）

A．java中一个类只允许实现一个接口

B.抽象类中允许有非抽象方法的存在

C．类变量（实例变量）可以直接用类名调用

D.通过super可以调用基类的构造函数

1. 以下哪个表达式是不合法的（ B  ）

A．String   x=”Hello”;  int  y=9;   x+=y;

B．String  x=”Hello”;   int  y=9;  if(x= =y)  { }

C．String   x=”Hello”;  int  y=9;  x=x+y;

D．String   x=null;  int  y=(x!=null)&&(x.length()>0)

1. class person{

   public int addvalue(int a,int b){

       int s;

       s=a+b;

       return s;

   }

}

class child extends parent

{

}

1. 若要在child类中对addvalue方法进行重写，下面对于child类中的addvalue方法头的描述哪个是正确的：( D )

A）int addvalue(int I,int j)

B）void addvalue()

C）void addvalue(double i)

D）int addvalue(int a)

1. 下面程序在fun（）方法当出现数组下标超过界限的情况下的输出结果是：（ C ）

public void test(){

try

{

fun();

System.out.print(“情况1”);

}catch(ArrayIndexOutOfBoundsException e)

{

System.out.print(“情况2”);

}

catch(Exception e)

{

System.out.print(“情况3”);

}

finally{

System.out.print(“finally”);

}

}

A．情况1

B.情况2

C.情况2finally

D.情况3finally

**二． 填空题**

1．Java程序分为应用程序和小程序两种，其中应用程序必须包含

­­­­­­­­­­­­­­\_\_\_\_\_\_\_main\_\_\_\_方法，该方法的格式是public static void main(String[] args){}\_\_\_\_\_\_。

2．Java程序通过\_\_\_\_\_\_break、continue\_\_\_\_\_\_\_\_语句跳出本次循环(两种方式)。

3．列举一个最常见到的runtime exception（运行时异常类型）\_\_\_\_ArrayIndexOutOfBoundsException \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_。

4．抽象类通过\_\_\_\_\_\_\_abstact\_\_\_\_关键字声明，此关键字不能和\_\_\_private\_\_\_\_\_\_修饰符一起连用。

5．若想引用util包中的所有类，其格式应该为\_\_\_\_\_\_\_\_import java.util.\*\_\_\_\_\_\_\_。

6．已知定义字符串s，其格式为

String s=new String (“hello”).

若要求出s的长度，应调用String类中的\_\_\_\_length()\_\_\_\_\_\_\_方法。

1. **简答题**
2. **String、StringBuffer、StringBuilder有什么区别**

**String创建的字符串是不可变的，不可以进行修改、追加等操作，一旦进行修改，就会生成新的一个hashcode，产生一个新的对象。**

**StringBuilder创建的字符串是可变的，进行修改追加等操作后还是同一个对象，不会产生新的hashcode。**

**StringBuffer和StringBuilder相似，只是增加了**synchronized，起到了线程安全保护作用，效率要比**StringBuilder低。**

1. **抽象类与接口有什么区别(介结相同点和不同点)**

**相同点:**

**都被abstract修饰**

**内部都有抽象方法**

**都不可以创建自己的实例，都是父亲的引用指向子类的实例**

**不同点：**

**定义的关键字不同 接口：interface 2.抽象类：abstract class**

**结构体不同: 接口：常量、抽象方法、被defulat、或者public static修饰的普通方法,抽象类：常量、成员变量、普通方法、构造方法、抽象方法**

**关系：一个抽象类可以实现多个接口，通过抽象类或者类去是实现接口，且可以继承一个类,接口可以通过接口继承接口，可以多继承，可以通过类实现接口，可以同时实现多个接口**

1. **==与equalse的区别**

**==比较的是地址 equals比较的内容。且基本类型不可以使用equals方法**

1. **Arraylst,LinkedList,HashMap,TreeMap,HashSet的区别(底层实现方式,效率...)**

**ArrayList、LinkedList是List的实现类，他们都是有序、可重复的集合。**

**HashSet是Set集合的实现类，它是无序、不可重复的集合。**

**HashMap、TreeMap是Map集合，就是存储键(key)-值(value）对的集合。**

ArrayList的底层实现是一个数组，默认的无参构造，为一个初始容量为 10 的空列表，内存不够的时候默认扩展50%，其查询效率高、增删效率低。

**LinkedList底层实现是一个链表。**其查询效率高、增删效率低。

**HashSet其底层 HashMap 实例的默认初始容量是16。**

**HashMap是Map接口的实现类，它实现了Map接口中的抽象方法，HashMap底层依靠hash表(数组+链表实现)，构造一个具有默认初始容量(16)和默认加载因子(0.75)的空 HashMap。在长度达到12时进行扩充。**

**基于红黑树（Red-Black tree）的 NavigableMap 实现。该映射根据其键的自然顺序进行排序，或者根据创建映射时提供的 Comparator 进行排序，具体取决于使用的构造方法**