1. JDK 和 JRE 有什么区别？

JRE全称是java runtim enviroment，有了这个就能够跑java程序，面向的主要是程序使用者，但是还不能进行java开发。JDK是java开发的整个核心，他包含了JRE，另外还有java工具和java基础类库等，有了JDK就不仅能跑java程序，还能开发java程序了

1. == 和 equals 的区别是什么？

==主要是对比的是变量的在内存中地址，而equals如果没有重写的话对比的也是内存中的地址，一般我们会比equals进行重写，对比对象中的内容，比如String就是对equals进行了重写，对比的是字符串的内容是否相同。一般对于值类型我们使用==，因为常量在java常量池中，值类型相同地址就是相同的。而引用类型使用equals,因为引用类型每次new出来在堆中都会重新开辟一个地址出来，地址是不相同的，所以使用equals对比对象中的内容。

1. 两个对象的 hashCode()相同，则 equals()也一定为 true，对吗？

不对，两个不同的对象，经过hash函数得计算，完全有可能相同，比如hashMap中不同的对象就会发生hash碰撞，所以不为true。反之equals()为true，hashCode也一定不相等，如果重写equals后，如果不重写hashcode，则hashcode就是继承自Object的，返回内存编码，而equals相同的通常不是同一个对象，只是内容相同，这时候可能出现equals相等，而hashcode不等。不过设计有要求equals为true，hashCode一定要相同，所以一般我们会对equals和hashCode同时重写。你的对象使用集合时，就会等不到正确的结果

1. final 在 java 中有什么作用？

Final在意义上来说是最后了，就是不可变了，他的作用主要分为了3类，修饰数据，修饰方法，修饰类；

修饰数据：修饰数据又分为修饰基本数据和修饰引用数据，当修饰基本数据的时候，当前变量就相当于一个常量，当赋值之后就不能改变当前的值。当修饰对象的时候，只是限定了引用的对象不能改变，就是他的地址不能改变，不过对象的值是可以改变的。另外，修饰方法参数和修饰变量是一样的，方法参数传入后，在方法体中就不能改变了。成员变量必须在声明的时候赋值或者构造器中赋值。局部变量随时赋值，不过只能赋值一次，赋值之后就不能改变了。

修饰方法：修饰方法主要是从设计的角度考虑的，当方法加了final之后，他的子类就不能够对该方法进行重写了。

修饰类：修饰类之后，该类就不能够被继承了。

1. java 中的 Math.round(-1.5) 等于多少？

Round是四舍五入，不过他的计算方式是不论正负，加上1.5再向下取整，-1.5+1.5是-1.0，向下取整是-1.0,因此结果是-1.0

1. String 属于基础的数据类型吗？

不是String是一个类，他底层的数据结构是字符数组

1. java 中操作字符串都有哪些类？它们之间有什么区别？

String,StringBuilder和 StringBuffer，他们3个的本质都是char数组。

String 是final修饰的，是不可变的类，他的值是不可变的，每次对字符串进行操作的时候都会生成新的String对象返回，效率比较低下。一般对字符串操作我们会用StringBulider类，他是可变的字符序列，默认长度为16，默认值为字符为0的字符，就是空格。当需要扩容是使用了数组拷贝。StringBuffer就是线程安全版的StringBuilder,对所有加了sync关键字来保证线程安全。

1. String str="i"与 String str=new String(“i”)一样吗？

不一样，前者指向的是常量池中的i，后者指向的是java堆中的的i。首先他们两者都会先去看常量池中是否有i，没有就会创建这个常量，然后返回。另外，new关键词都会去堆中新开辟新的地址。

1. 抽象类必须要有抽象方法吗？

抽象类不一定有抽象方法，但是有抽象方法的一定是抽象类，并且抽象类不能直接创建，必须通过其子类进行创建。

1. 普通类和抽象类有哪些区别？

1.抽象类可以有抽象方法，普通类不能有。

2．抽象类不能被直接创建，普通类能

3．意义上，抽象类是一种约定，

4．抽象类中的抽象方法不能被private,static,final,native修饰。

5. 子类必须实现抽象类的所有方法，否则这个类也是抽象类

1. 抽象类能使用 final 修饰吗？

不能，首先final修饰的类不能够被继承，不能扩展。而抽象类的意义就是需要被子类继承，抽象方法需要被子类实现，创建也只能通过子类创建。而加了final的抽象类是没有任何意义的，因此在语法上和设计上都是不能的。

1. 类加载的顺序

父类静态变量，父类静态代码块->子类静态变量，子类静态代码块->父类的成员变量，非静态代码块，构造方法->子类成员变量，非静态代码块，构造方法

1. 静态代码块一定比构造方法先执行吗?

不一定，如果在父类的静态代码块中，new了一个子类，构造方法就会比父类先执行。

1. 接口和抽象类有什么区别？

构造方法：抽象类可以有，接口没有

普通方法：抽象类有，接口再1.8之后有默认方法。

实现：抽象类由extend,一个类只能继承一个，接口是implement,一个类可以实现多个接口。

访问修饰符：抽象类是public,protected,default，接口是public

静态方法：抽象类可以有，接口不可以

Main方法：抽象类可以有，接口不可以有

1. 抽象类可以说具有接口的全部功能，为什么还要有接口呢？

这是也是java的特性，首先java只能有一个超类，当需要更多的扩展就不行了，因此就出现了接口，可以说使对抽象类的一个扩展。从抽象的概念上来说，抽象类是对一个事物的抽象，而接口是对动作的抽象。另外，也可以一个类具有多个超类，也能实现抽象类+接口的组合，比如C++，不过这样设计会使语言变得十分复杂。Java接口的出现能够避免多重继承的复杂性和低效性。

1. java 中 IO 流分为几种？
2. BIO、NIO、AIO 有什么区别？
3. Files的常用方法都有哪些？
4. 按值传递和按引用传递的区别
5. 面向对象的三大特性的理解
6. 什么是泛型
7. dk 中线程池的类型
8. Java 反射是什么，反射的用处
9. java 怎么和数据库交互
10. AQS 的实现原理
11. c++ 和 java 的区别
12. 有哪些 jdk 调优工具
13. lambda 函数实现机理
14. java 深拷贝浅拷贝，深拷贝如何去做