

1、java中的oop:

答：oop是一种面向对象编程，即万物皆对象；它有三个特点：一是封装：即封装对象的属性与行为，封装对象的载体是类，封装可以隐藏内部程序的具体实现，对外提供访问的接口，提高程序的可靠性，特点是提高代码的可维护性；二是继承，子类继承父类的属性和行为，子类拥有父类的一切属性和行为，还可在其基础上进行扩展，拥有自己的特殊属性，特点是提高代码的复用；三是多态，多态就是表现出多种形态，使用不同的对象执行同一接口而得到不同的结果，特点是提高代码的可维护性和扩展性；

2、类与对象:

答：所谓对象就是真实世界中的实体，对象与实体是一一对应的，也就是说现实世界中每一个实体都是一个对象，它是一种具体的概念。

类是具有某些共同特征的实体的集合，它是一种抽象的概念，用程序设计的语言来说，类是一种抽象的数据类型，它是对所具有相同特征实体的抽象。

3、重写（Override）与重载（OverLoad）的区别:

答：重载是在一个类中，有多个方法名相同，但是参数列表与返回值不相同的方法，重写发生在有继承关系的子类中，在子类中，有与父类方法名相同，参数与返回值都相同的方法；重写抛出的异常不能比父类的大；重载只要参数列表相同则无法通过变异

4、java中的多态、向上转型与向下转型:

答：多态的特征是表现出多种形态，具有多种实现方式。或者说实现同一个接口，使用不同的对象而执行不同的操作；就比如有一杯水，我不知道它是烫的，温的还是冰的，只有我摸水杯才知道，我摸水杯这个动作，对于不同温度的水，就会得到不同的结果，这就是多态；多态的实现条件是：继承，重写，向上转型；

向上转型是子类引用的对象转化为父类的对象，通俗的说就是将子类对象转为父类对象，此处父类对象可以是接口；向上转型时需要注意的是，子类单独定义的属性与方法会丢失；子类应用不能指向父类对象：子类 a = （子类） new 父类（）；好处是减少代码重复性，提高系统扩展型；

向下转型是将父类对象转为子类对象；需要注意的是：向下转型的前提是父类对象转为子类对象，向下转型只能转为本类对象；

5、JDBC的使用步骤:

答：1、加载驱动

2、创建连接

3、创建statement对象

4、执行SQL

5、处理结果

6、释放资源

6、java多态的实现原理：

答：多态就是表现出多种形态，实现的原理是动态绑定，程序调用方法在运行期才动态绑定，JVM通过参数的自动转型来找到合适的办法