1. 面向对象的理解并举例？

卖东西:

面向过程：到商店-->找商户-->找商品-->买商品

面向过程：找老妈-->老妈买商品

2. 类与对象之间的关系？

类是一类具有相同特性事物的抽象集合，是对象的模版与蓝图。

3. 如何对类进行分析，如果创建自定义类对象，并如何指挥对象做事情？

类具有公有的属性及公共的方法，如果创建自定义对象，可以创建该类公共的方法

4. 对象的内存分布图？

创建对象时首先会在方法区加载该类class文件，解析该类具有的公有属性及公共行为，之后在栈内存中创建该类的引用类型变量，同时会在堆内存中创建该类的一个对象，栈内存中的引用类型变量指向堆内存中的对象。

5. 成员变量和局部变量的区别？

1.定义位置不同

成员变量：类之内，方法之外

局部变量：类之内，方法之内

2.作用范围不同

成员变量:在该类内部任何位置可以随意引用

局部变量:只在其方法定义的有效作用域内有效，一旦超出其作用域立即失效

3.生命周期不同

成员变量：随着对象的产生而产生，随着对象的消亡而消亡

局部变量：只在其方法定义的有效作用域内有效，一旦超出其作用域立即失效

4.初始值不同

成员变量：具有初始值，int 0，Boolean false

局部变量：没有初始值，必须初始化赋值才能使用

5.储存位置不同

成员变量:存储在堆内存中(对象存在的空间)

局部变量:存储在栈内存中

6. 私有的使用。

private关键字的使用：一般情况下对于一个类中的公有属性来讲，只要是实体类就应该进行private封装，并按需求创建公有的访问器及修改器。而对于该类的方法来说，如果只是给该类内部进行使用可以使用private关键字进行封装。对于一个类来说，如果只是为同一个包中使用，可以使用private进行封装。

7. 构造函数和一般函数的区别？

1.构造函数没有返回值类型

2.构造函数的函数名与该类的类名相同

3.构造函数不是手动调用的，是创建对象值JVM主动调用的，每创建一个对象就会调用一次构造函数

4.构造函数可以通过重载的方式多个出现

5.当一个类没有创建造函数是编译器会自动给其添加一个无参的构造函数(继承自object类)

6.当一个类显现创建一个构造方法时编译器不会个其添加无参的构造方法。

8. 构造函数什么时候用？

创建对象时调用该类的构造方法，对成员变量进行初始化赋值。

9. 构造代码块的作用？

对该类的对象进行统一的初始化赋值

10. this关键字的特点和使用以及应用场景？

this关键字指向本类对象

1.this关键字引用本类的构造方法时必须出现在第一句。

2.使用this关键字引用本类构造方法时要避免各自调用

3.this关键字代表本类对象，只能在非静态成员中使用

4.如果在一个函数中访问一个成员变量，而且没有存在同名的局部变量 时，那么 java编译器会默认在变量的前面加上 this关键字的。