第一周：面向对象

1.概念：面向对象OOP是基于面向过程而言，面向对象简单来说就是将功能封装到对象（数据和操作结合）里，我们面向对象，让对象去完成这些功能。

万物皆对象。（在我们程序员眼中，一切皆对象）

三大特性：封装，继承，多态

封装：减少了大量的冗余代码，将复杂的功能封装起来，对外开放一个接口，简单调用即可。

继承：单根性，传递性，减少了类的冗余代码，让类与类之间产生关系，为多态打下基础；一个新类继承了原有类的属性和方法，并增加了自己的新属性和新方法，称之为派生类。

多态：一个接口，多个方法。通过继承实现的不同对象调用相同的方法，进而有不同的行为

2.类与对象

类：物以类聚，把具有相似特性的对象归类到一个类中，类定义了这些相似对象拥有的相同的属性和方法，类的属性和方法统称为类成员。

对象（类的实例化）：通过类定义去创建一个类的对象，类只有一个，可以实例化出多个对象（一对多），类的定义里面属性值都是空的，而对象的属性都有具体的值。

3.继承与多态

（1）单继承原则：一个子类只能继承一个父类

（2）访问权限：

public:公有的类成员，可以在任何地方被访问（定义该成员的类即自身，该类的子类，其他类）

protected:受保护的类成员，可以被其自身及其子类访问

private:私有的类成员，只能被自身访问

（3）final关键词：

对于不想被任何类继承的类可以在class之前添加关键字final

对于不想被子类重写（修改）的方法，可以在方法定义的前面加final

（4）接口

接口就是把不同类的共同行为进行了定义，然后在不同的类里面实现不同的功能。定义接口:interface，继承接口：implements

接口里面的方法不需要有方法的实现。

implements用于表示类实现某个接口。

实现了某个接口之后，必须提供接口中定义的方法的具体实现。

接口不能被实例化。

可以用instanceof关键词判断某个对象是否实现了某个接口。

可以用extends让接口继承父接口。

（5）重写和重载

重写是子类对父类的允许访问的方法的实现过程进行重新编写, 返回值和形参都不能改变。即外壳不变，核心重写！

重载是在一个类里面，方法名字相同，而参数不同。返回类型可以相同也可以不同。每个重载的方法（或者构造函数）都必须有一个独一无二的参数类型列表。

（6）多态

相同的一行代码，对于传入接口不同的实现对象的时候，表现是不同的。

消除类型之间的耦合关系。

三个必要条件：继承，重写，父类引用指向子类对象

（7）抽象

在抽象方法前面加abstract关键字，可以表明这个方法是抽象方法，不需要具体实现。

在抽象类中可以包含有普通方法，有方法的具体实现。

继承抽象类的子类中需要实现抽象类中定义的抽象方法。