

Java是严格意思上面面向对象的语言

一. 面向对象编程:oop Object Oriented Programming, 迄今为止最接近人类思想的编程语言

面向对象的核心:

1. 类:具有相同特征和行为特征事物的抽象
2. 对象:万事万物皆对象

类(class)与对象(object)的区别:

看能不能具体到某个事物

关系:

1. 类是抽象的, 对象是具体的
2. 类是对象的类型(数据类型)
3. 对象是类的实例(变量)

创建类

```
/*访问修饰符 class 类名{  
    特征一  
    特征二  
    ....  
    行为1  
    行为2  
    ...  
}
```

注:类名使用大驼峰法

*/

特征->属性(字段)

格式:

访问修饰符 数据类型 属性名;

注:属性名使用小驼峰法

行为->方法

格式:

```
/*访问修饰符 返回值数据类型 方法名(参数列表) {  
    语句  
}
```

注;方法名使用小驼峰命名法*/

创建类 相当于 生成一个新的数据类型

定义整形变量a

```
1 int a=10;
```

定义Car类型的变量car

一般把变量car叫对象

```
1 Car car=new Car();
```

创建对象的格式:

类名 对象名=new 类名();

对象名使用小驼峰法

对属性赋值(设置器set, 访问器get)

对象. 属性名=值

```
1 car.wheelcount=4;  
2 car.brand="兰博基尼";  
3 car.color="土豪金";
```

获取属性的值

```
1 System.out.println(car.brand);  
2 System.out.println(car.wheelcount);  
3 System.out.println(car.color);
```

调用方法

格式：

对象.方法()

```
1 car.run();  
2 car.stop();
```

```
1 Girl girl1=new Girl();  
2 girl1.name="刘亦菲";  
3 girl1.age=31;  
4 girl1.playing();  
5 girl1.cooking();
```

二. 面向过程编程解决的思路：

1. 分析解决问题的步骤
2. 把每个步骤形成代码块
3. 依次执行代码块

三. 面向对象编程解决的思路：

1. 分析问题中出现的对象
2. 分析对象的特征和行为, 把具有相同特征和行为归成一类
3. 通过类生成对象
4. 对象调用方法