

- 1 静态字段
- 2 静态常量
- 3 静态方法
- 4 工厂方法

## 1 静态字段

---

用 `static` 关键字定义静态字段。非静态字段属于对象，每个对象都有自己的一个副本；静态字段属于类，类的所有对象共享同一个静态字段。

静态字段既可以通过对象访问，也可以直接用类名访问。

## 2 静态常量

---

静态变量使用得比较少，而静态常量很常用。例如我们常用的 `System.out` 是 `System` 类的一个静态常量字段，`Math.PI` 是 `Math` 类的一个静态常量字段。上面列出的形式就是通过类名访问的。

## 3 静态方法

---

静态方法是不在对象上执行的方法，可以认为静态方法没有 `this` 参数。例如，`Math` 类的 `pow` 方法就是一个静态方法，调用这个方法时并不使用任何 `Math` 对象，自然也不需要 `this` 参数。

静态方法不能访问非静态字段，因为静态方法没有 `this` 参数，而访问非静态字段时需要 `this` 参数来指明对象，二者是矛盾的。

与静态字段类似，静态方法既可以通过类名调用，也可以通过对象来调用。不过静态方法并不作用于对象，如果通过对象调用静态方法容易造成混乱。因此最好使用类名来调用静态方法。静态字段同理。

在下面两种情况下可以使用静态方法：

- 方法不需要访问对象状态，它需要的参数都通过显式参数提供。
- 方法只需要访问类的静态字段。

## 4 工厂方法

---

静态工厂方法用于构造对象。前面出现过的 `LocalDate.now` 和 `LocalDate.of` 就是工厂方法。

在下面的情况下可以使用工厂方法而不是构造器来构造对象：

- 如果希望构造器有不同的名字，表示不同的情况，就要使用工厂方法。
- 如果希望构造对象的类型与本类不同，就要使用工厂方法。