

this关键字

- Java中提供一个关键字this，用来表示当前对象本身，或者当前的一个实例，用于在方法中访问对象的其他成员
 - 通过this区分同名变量
 - 成员变量与方法内部变量重名时，在方法内部调用成员变量，使用this解决与局部变量名称冲突问题
 - 实例：

```
public class Test4{
    public String name;//定义成员变量name
    public int age;//定义成员变量age

    //定义构造方法Test4
    public Test4(String name,int age){
        this.name = name;
        this.age = age;
    }
    //定义方法say
    public void say(){
        System.out.println("大家好，我叫"+ name +", 我今年"+ age + "岁! ");
    }

    public static void main(String [] args){
        //创建对象obj
        Test4 obj = new Test4("黎明",23);
        obj.say();
    }
}
```

调试结果：

```
F:\java\第3章 面向对象\Test3>javac Test4.java

F:\java\第3章 面向对象\Test3>java Test4
大家好，我叫黎明，我今年23岁！
```

- ◦ 通过this关键字调用成员变量
 - this关键字用来表示当前对象本身或者当前的一个实例，通过this关键字调用本对象的所有方法和属性。
- 实例

```
public class Test5{
    public int x = 360;//定义常量x
    public int y = 720;//定义常量y

    //定义方法求和sum
    public void sum(){
        int z = this.x + this.y;
        System.out.println("x+y = " + z);
    }

    public static void main(String[] args){
        //创建对象
        Test5 p1 = new Test5();
        //调用方法sum
        p1.sum();
    }
}
```

```
}  
  
}
```

调式结果:

```
F:\java\第3章 面向对象\Test3>javac Test5.java
```

```
F:\java\第3章 面向对象\Test3>java Test5
```

```
x+y = 1080
```

- 作为方法名来初始化对象
 - 相当于调用本类的其他构造方法，它必须作为构造方法的第一句。
- 实例

```
public class Test6{  
    //定义变量  
    public String name;  
    public int age;  
    //定义构造方法  
    public Test6(){  
  
        this("黎明",26);  
    }  
  
    //定义构造方法  
    public Test6(String name,int age){  
        this.name = name;  
        this.age = age;  
    }  
    //定义方法  
    public void say(){  
        System.out.println("大家好，我叫"+ name +", 今年我"+ age +"岁");  
    }  
    public static void main(String[] args){  
        //实例化对象p1  
        Test6 p1 = new Test6();  
        //调用方法  
        p1.say();  
    }  
}
```

调试结果:

```
F:\java\第3章 面向对象\Test3>javac Test6.java
```

```
F:\java\第3章 面向对象\Test3>java Test6
```

```
大家好，我叫黎明，今年我26岁
```

- 值得注意的是:
 - 在构造方法中调用另一个构造方法，调用动作必须置于最起始的位置。
 - 不能在构造方法以外的任何方法内调用构造方法。
 - 在一个构造方法内只能调用一个构造方法。
- -作为参数传递
 - 需要在完全分离的类中调用一个方法，并且将当前对象的一个引用作为参数传递
- 实例

```
public class Test7{  
    public static void main(String[] args){  
        B b = new B(new A());  
    }  
}
```

```
}  
class A{  
  
    public A(){  
        new B(this).print();//匿名对象  
    }  
  
    public void print(){  
        System.out.println("Hello from A!");  
    }  
}  
  
class B{  
    A a;  
    public B(A a){  
        this.a=a;  
    }  
  
    public void print(){  
        a.print();  
        System.out.println("Hello from B!");  
    }  
}
```

• 调试结果

```
F:\java\第3章 面向对象\Test3>javac Test7.java
```

```
F:\java\第3章 面向对象\Test3>java Test7  
Hello from A!  
Hello from B!
```

- 匿名对象就是没有名字的对象。如果对象只使用一次，就可以作为匿名对象，代码中 `new B(this).print();` 等价于 `(new B(this)).print();`，先通过 `new B(this)` 创建一个没有名字的对象，再调用它的方法。