



2、具体内容

在程序之中使用“{}”定义的结构就称为代码块，而后根据代码块出现的位置以及定义的关键字的不同，代码块可以分为：普通代码块、构造块、静态块、同步代码块，其中对于同步代码块是在多线程的时候才会讲解的。

■普通代码块

普通代码块的主要特点是定义在一个方法之中的代码块。

范例：观察一个程序

```
public class JavaDemo{
    public static void main(String[] args) {
        if(true){//条件一定满足
            int x = 10;    //局部变量
            System.out.println("x=" + x);
        }
        int x = 100;    //全局变量
        System.out.println("x=" + x);
    }
}
```

按照Java程序的开发标准规定相同名称的变量是不能够在同一个方法之中存在的，但是由于此时有不同的分界描述。而如果说要定义普通代码块，那么只需要将当前的if语句取消即可。

```
public class JavaDemo{
    public static void main(String[] args) {
        //普通代码块
        int x = 10;    //局部变量
        System.out.println("x=" + x);
    }
    int x = 100;    //全局变量
    System.out.println("x=" + x);
}
```

```
}
```

可以在一个方法之中进行一些结构的拆分，以防止相同变量名称所带来的相互影响。

■构造代码块

构造块是定义在一个类之中的。类对象实例化

范例：观察构造块

```
class Person
{
    public Person(){
        System.out.println("【构造方法】Person类构造方法执行");
    }
    {
        System.out.println("【构造块】Person构造块执行");
    }
}

public class JavaDemo{
    public static void main(String[] args) {
        new Person();
        new Person();
        new Person();
    }
}
```

构造块会优先于构造方法执行，并且每一次实例化新对象的时候都会调用构造块中的代码。

■静态代码块

静态代码块主要指的是使用static关键字定义的代码块，静态块的定义需要考虑到两种情况：主类中定义静态块、非主类中定义静态块。

范例：在非主类中进行静态块的定义

```
class Person
{
    public Person(){
        System.out.println("【构造方法】Person类构造方法执行");
    }
    static{
        System.out.println("【静态块】静态块执行");
    }
    {
        System.out.println("【构造块】Person构造块执行");
    }
}
```

```

public class JavaDemo{
    public static void main(String[] args) {
        new Person();
        new Person();
        new Person();
    }
}

```

此时可以发现静态代码块会优先于构造块执行，并且不管有多少个实例化对象出现静态代码块只会执行一次，静态代码块的主要目的是为了类中的静态属性初始化。

范例：观察静态属性初始化

```

class Message
{
    public static String getCountry(){
        //该消息的内容可能来自于网络或其它服务器
        return "中华人民共和国";
    }
}
class Person
{
    private static String country;
    static{
        country = Message.getCountry();//编写一部分的代码
        System.out.println(country);
    }
}

public class JavaDemo{
    public static void main(String[] args) {
        new Person();
    }
}

```

对于静态代码块还必须考虑另一种情况，在主类中定义的形式。

范例：在主类中进行静态代码块的定义

```

public class JavaDemo{
    static{
        System.out.println("*****程序初始化*****");
    }
    public static void main(String[] args) {

        System.out.println("*****main*****");
    }
}

```

静态代码块优先于主方法先执行。

