以下是SDK中的叙述：

Instantiates a layout XML file into its corresponding View objects. It is never used directly. Instead, use getLayoutInflater() or getSystemService(String) to retrieve a standard LayoutInflater instance that is already hooked up to the current context and correctly configured for the device you are running on. For example:

|  |  |
| --- | --- |
| 1 | LayoutInflater inflater = (LayoutInflater)context.getSystemService(Context.LAYOUT\_INFLATER\_SERVICE); |

To create a new LayoutInflater with an additional LayoutInflater.Factory for your own views, you can use cloneInContext(Context) to clone an existing ViewFactory, and then call setFactory(LayoutInflater.Factory) on it to include your Factory.

For performance reasons, view inflation relies heavily on pre-processing of XML files that is done at build time. Therefore, it is not currently possible to use LayoutInflater with an XmlPullParser over a plain XML file at runtime; it only works with an XmlPullParser returned from a compiled resource (R.something file.)

翻译和阐述：

LayoutInflater类的作用是，将xml布局文件实例化为它对应的View对象。这个类不能直接使用，也就是不能直接调用其中的成员。一般，我们通过getLayoutInflater()方法或者 getSystemService(String)

方法来获得该类的实例，通过以上两个方法获得的LayoutInflater类实例，已经和当前的上下文关联起来，并且已经正确配置在当前程序运行的设备上。我们顺便说一下这两个获得实例的方法：getLayoutInflater()方法，

并不是上下文的方法，Activity类有这个方法，不需要传入参数，在Activity中直接调用即可。Fragment类也有这个方法，但是需要传入一个Bundle对象作为参数。可以看到，通过该方法获得的LayoutInflater类对象

和上下文环境相配合。getSystemService(String)方法是Context的方法，需要传入Context的成员变量作为参数，获得相应的对象，要获得LayoutInflater对象，需要传入Context.LAYOUT\_INFLATER\_SERVICE

LayoutInflater类的inflate方法是所有布局填充方法的基石

public View inflate (int resource, ViewGroup root, boolean attachToRoot)

Inflate a new view hierarchy from the specified xml resource. Throws InflateException if there is an error.

从指定的xml文件生成新的view视图关系。出现错误时，抛出InflateException异常。

参数分析

第一个参数，就是我们要填充的xml文件

第二个参数，这个要和第三个参数有关系，大家慢慢看。若是第三个参数为true，那么第二个参数的意义是，从第一个参数填充成的view对象的父控件；若是第三个参数为false，那么第二个参数的意义是，

可以为第一个参数生成的view对象的根布局提供一系LayoutParams参数的控件。

第三个参数，从第一个参数填充成的view对象是否要附着到第二个参数指定的空间上作为子控件。

View inflate方法和LayoutInflater inflate方法的区别详解

首先我们要清楚两者大致的区别，之后我们再来慢慢看两者具体的不同之处

LayoutInflater类的inflate方法适用于所有需要进行布局填充的场景，是Android中专门进行布局填充的方法，Android中其他需要

使用布局填充的地方，都会调用本方法，而不是View类中的inflate方法。该方法不是静态方法，需要先创建LayoutInflater类的对象

才能调用。

View类中的inflate方法内部包裹了LayoutInflater类的inflate方法，这个方法是一个静态方法，不需要创建View类的对象，直接使用

View类名调用，相比上一种方法是一种简便方法。但很明显，这个方法不如上一个方法功能强大。

关于View类的inflate方法

public static View inflate (Context context, int resource, ViewGroup root)

Inflate a view from an XML resource. This convenience method wraps the LayoutInflater class, which provides a full range of options for view inflation.

将一个xml资源填充成一个view对象。这个简便的方法包裹了LayoutInflater类，后者提供了view对象填充的所有方法。

参数分析：

第一个参数，上下文

第二个参数，要填充的xml资源

第三个参数，填充成的view对象的根布局

说明，从SDK解释中的“convenience”一词中，我们就可以看到View类中inflate的主要特点，就是简便。它将LayoutInflater类封装，且是一个静态方法，便于调用。

以下为源码：

public static View inflate(Context context, @LayoutRes int resource, ViewGroup root) {

LayoutInflater factory = LayoutInflater.from(context);

return factory.inflate(resource, root);

}