**蓝牙的打开方式1：**

会弹出提示窗口

Intent enableIntent = new Intent(BluetoothAdapter.ACTION\_REQUEST\_ENABLE)；

startActivityForResult(enableIntent,1);

**蓝牙的打开方式2：**

<uses-permission android:name=”android.permission.BLUETOOTH”/>

<uses-permission android:name=”android.permission.BLUETOOTH\_ADMIN”

BluetoothAdapter adapter = BluetoothAdapter.getDefaultAdapter();

Intent turnOn = new Intent(BluetoothAdapter.ACTION\_REQUEST\_ENABLE);

startActivityForResult(turnOn, 0);

|  |  |
| --- | --- |
| **Sr.No** | **常数说明** |
| 1 | **ACTION\_REQUEST\_DISCOVERABLE** 此常数用于开启蓝牙的发现 |
| 2 | **ACTION\_STATE\_CHANGED** 此常量将通知蓝牙状态已经改变 |
| 3 | **ACTION\_FOUND** 此常数用于接收关于所发现的每个设备的信息 |

启用了蓝牙功能之后，可以通过调用 getBondedDevices()方法来获取配对设备列表。它返回一组的蓝牙设备。其语法如下：

private Set<BluetoothDevice>pairedDevices;

pairedDevices = BA.getBondedDevices();

除了配对的设备，还有API，让更多蓝牙控制权等方法。它们在下面列出。

|  |  |
| --- | --- |
| **Sr.No** | **方法及说明** |
| 1 | **enable()** 这种方法使适配器，如果未启用 |
| 2 | **isEnabled()** 如果适配器已启用此方法返回true |
| 3 | **disable()** 该方法禁用适配器 |
| 4 | **getName()** 此方法返回的蓝牙适配器的名称 |
| 5 | **setName(String name)** 此方法更改蓝牙名称 |
| 6 | **getState()** 此方法返回蓝牙适配器的当前状态 |
| 7 | **startDiscovery()** 此方法开始蓝牙120秒的发现过程。 |

public void visible(View view){

Intent getVisible = new Intent(BluetoothAdapter.

ACTION\_REQUEST\_DISCOVERABLE);

startActivityForResult(getVisible, 0);