创建一个继承了Application类的子类，用于保存全局资源，如保存当前用户与服务器建立好的连接对象、当前登录成功的用户的用户账号、好友列表

**public** **class** ImApp **extends** Application {

**private** QQConnection conn;// 用户连接

**private** **long** account; // 用户账号

**private** String buddyListJson; // 用户好友列表的json格式信息

**private** CoreService coreService ;

/\*\*

\* 保存QQ消息的数据库操作DAO

\*/

**private** QQMessageDao msgDao ;

@Override

**public** **void** onCreate() {

**super**.onCreate();

System.***out***.println("ImApp成功创建了");

// 创建数据库

SQLiteDatabase db = **new** DaoMaster.DevOpenHelper(getBaseContext(), "im.db", **null**).getWritableDatabase();

DaoMaster master = **new** DaoMaster(db);

DaoSession session = master.newSession();

msgDao = session.getQQMessageDao();

}

**public** QQMessageDao getMsgDao() {

**return** msgDao;

. . . . . . . .

. . . . . . . . .

此处省略所有的get、set方法

在其他类中获取到该Application类

1. 在Activity和Sservice服务中直接获取

imApp = (ImApp) getApplication() ;

2、如果Context是Activity传过来的，则通过context上下文获取到

ImApp imApp = (ImApp)((Activity)context).getApplication();

Context一定要是Activity传过来的，如果是getBaseContext()或getApplicationContext()则会抛出类型转换错误的异常

需要在清单文件中的application节点中增加一个name属性，属性值为上面创建的Application子类的全类名

<application

android:name=*"com.jgsu.imessage.utils.ImApp"*

android:allowBackup=*"true"*