1、安卓四大组件以及作用：Activity、Service、BroadcastReceiver、ContentProvider

**简述Android四大组件及其作用。**

答：Android四大组件为：Activity、Service、BroadcastReceiver、ContentProvider

1）Activity : 为用户提供可视化界面及操作。一个应用程序通常包含多个Activity，每个Activity负责管理一个用户界面。

2）Service ：长期运行在后台的用户组件，没有用户界面。用于在后台处理耗时操作，不受活动生命周期的影响。

3） BroadcastReceiver：监听来自系统或应用程序的广播消息，并实现在不同组件之间的通信。

4）ContentProvider：在不同程序之间实现数据共享

2、Activity生命周期

**请思考Activity生命周期中包含有哪几种状态。**

答：Activity的生命周期分为五种状态，分别是启动状态、运行状态、暂停状态、停止状态和销毁状态，其中启动状态和销毁状态是过渡状态，Activity不会在这两个状态停留。

1）启动状态

Activity的启动状态很短暂，一般情况下，当Activity启动之后便会进入运行状态。

2）运行状态

Activity在此状态时处于屏幕最前端，可以与用户进行交互，如点击、双击、长按事件等。

3）暂停状态

当前Activity上覆盖了一个透明或者非全屏的Acitvity时，被覆盖的Activity就处于暂停状态。

4）停止状态

当Activity完全不可见时，它就处于停止状态，但仍然保留着当前状态和成员信息。如果系统内存不足，那么这种状态下的Activity很容易被销毁。

5）销毁状态

当Activity处于销毁状态时，将被清理出内存。

3、Service两种启动方式、区别，生命周期

**请思考服务有几种开启方式，每种开启方式的特点和生命周期。**

答：启动服务有两种方式，分别是通过startService()方式启动服务、bindService()方式启动服务。通过startService()方式开启的服务会长期的运行在后台，并且服务的状态与开启者的状态没有关系。通过bindService()方式开启的服务与开启者的状态有关，当调用者销毁了，服务也会被销毁。

startService()启动方式下的生命周期：

通过startService()方法启动服务时，执行生命周期的方法为onCreate()、onStartCommand()，然后服务处于运行状态，直到自身调用stopSelf()方法或者其他组件调用stopService()方法时服务停止，最终被系统销毁。

bindService()启动方式下的生命周期：

通过bindService()方法启动服务时，执行的生命周期的方法为onCreate()、onBind()，然后服务处于运行状态，直到调用unBindService()方法时，服务被解绑调用onUnbind()方法，最终被销毁。

4、Android 5种常见布局

**请思考Android中有几种布局，并说明每种布局的特点。**

答：Android中的布局分为五种，分别是相对布局、线性布局、表格布局、帧布局、绝对布局。

1）相对布局

在Android程序创建时，默认采用的就是相对布局。相对布局通常有两种形式，一种是相对于容器而言的，一种是相对于控件而言的，为了能准确定位布局中的控件。

2）线性布局

线性布局是Android中较为常用的布局方式，它使用<LinearLayout>标签表示。线性布局主要有两种形式，一种是水平线性布局，一种是垂直线性布局。

3）表格布局

表格布局就是让控件以表格的形式来排列控件的，只要将控件或信息放在单元格中，控件就可以整齐的排列。

4）帧布局

帧布局是Android布局中最简单的一种，帧布局为每个加入其中的控件创建一个空白区域（称为一帧）。采用帧布局方式设计界面时，所有控件都默认显示在屏幕左上角，并按照先后放入的顺序重叠摆放。

5）绝对布局

绝对布局需要通过指定x、y坐标来控制每一个控件的位置，放入该布局的控件需要通过android:layout\_x和android:layout\_y两个属性指定其准确的坐标值，并显示在屏幕上。

5、Android常用的存储方式

**请思考Android中有几种数据存储方式以及各自特点。**

答：Android系统中的五种数据存储方式。

1）文件存储：以IO流形式把数据存入手机内存或者SD卡，可以存储大数据，如音乐、图片或者视频等。

2）SharedPreferences：它本质上是一个XML文件，以Map<Object,Object>形式存入手机内存中。常用于存储较简单的参数设置，如账号密码的存储。

3）SQLite数据库：SQLite是Android自带的一个轻量级的数据库，是项目中经常采用的一种数据存储方式，通常用于存储用户信息等。

4）ContentProvider：又称为内容提供者，是Android四大组件之一，主要用于应用程序之间的数据交换，它可以将自己的数据共享给其他的应用程序使用。

5）网络存储：将数据存储到服务器上，通过网络提供的存储空间来存储/获取数据信息。

6、如何创建数据库，在数据库中创建表，数据库版本变化时删除表

**编程题：创建数据库**

7、SharedPreferences保存数据

**请思考在Android中如何使用SharedPreferences类。**

答：使用SharedPreferences存储数据时，需要先获取SharedPreferences对象，通过该对象获取到Editor对象，然后通过Editor对象的相关方法存储数据，具体代码如下：

|  |
| --- |
| //获取sp对象，参数data表示文件名，MODE\_PRIVATE表示文件操作模式  SharedPreferences sp = getSharedPreferences("data",MODE\_PRIVATE);  SharedPreferences.Editor editor = sp.edit(); // 获取编辑器  editor.putString("name", "张三"); // 存入String类型数据  editor.putInt("age", 18); // 存入int类型数据  editor.commit(); // 提交修改 |

SharedPreferences获取数据时比较简单，只需要创建SharedPreferences对象，然后使用该对象获取相应key的值即可，具体代码如下：

|  |
| --- |
| SharedPreferences sp = context.getSharedPreferences();  String data = sp.getString("name",""); // 获取用户名 |

SharedPreferences删除数据时与存储数据类似，同样需要先获取到Editor对象，然后通过该对象删除数据，最后提交，具体代码如下：

|  |
| --- |
| SharedPreferences sp = context.getSharedPreferences ();  Editor editor = sp.edit();  editor.remove("name"); // 删除一条数据  editor.clear(); // 删除所有数据  editor.commit(); // 提交修改 |

8、广播接受者、内容提供者

**请思考Android中广播接收者的作用。**

答：在Android系统中，广播是一种运用在应用程序之间传递消息的机制。要过滤并接收广播中的消息需要使用BroadcastReceiver（广播接收者），广播接收者是Android四大组件之一，通过广播接收者可以监听系统中的广播消息，并实现在不同组件之间的通信。

**请思考广播有几种类型，以及不同类型的区别。**

答：在Android系统中，根据广播的执行顺序不同，可将其分为有序广播和无序广播。

1）无序广播

无序广播是完全异步执行的，发送广播时，所有监听这个广播的广播接收者都会接收到此广播消息，但接收的顺序不确定。无序广播的效率比较高，但无法被拦截。

2）有序广播

有序广播是一种同步执行的广播，按照接收者的优先级被依次接收。发送广播时，只有一个广播接收者能接收消息，当这个广播接收器中的逻辑执行完毕后，广播才会继续传递。有序广播的效率较低，但此类型是有先后顺序的，并可被拦截。

**请思考在程序中如何使用内容提供者操作数据。**

答：ContentProvider（内容提供者）是Android系统四大组件之一，其功能是在不同程序之间实现数据共享。在Android系统中，应用程序之间是相互独立的，若应用程序之间需要共享数据，就要用到ContentProvider。ContentProvider是不同应用程序之间进行数据交换的标准API，它以Uri的形式对外提供数据。其他应用则使用ContentResolver，并根据ContentProvider提供的Uri操作指定数据。

**请思考什么是内容观察者，内容观察者如何应用。**

答：内容观察者（ContentObserver）是用来观察指定Uri所代表的数据。当ContentObserver观察到指定Uri代表的数据发生变化时，就会触发ContentObserver的onChange()方法。此时在onChange()方法里使用ContentResovler可以查询到变化的数据。

9、Handler

编程题：倒计时