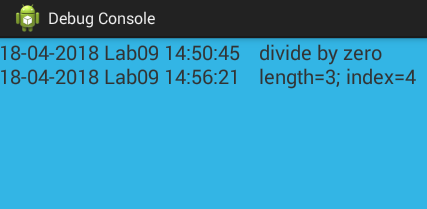
**Task#1:** Create an application named DebugConsole having the following UI:



* Main purpose of this application is to display exceptions occurred in other user applications through broadcast receiver.
* This application must register a custom broadcast receiver having DEBUG\_INFO action. On receiving that broadcast, it must obtain timestamp, application name from the associated intent.
* Other applications must send their app name and a message while timestamp must be managed by this application.
* Note that the UI contains only one TextView component having light blue background color with black text.
* The text must be scrollable by user. You may use setMovementMethod(new ScrollingMovementMethod) to set the scrolling of TextView.

**Solution:**

**XML:**

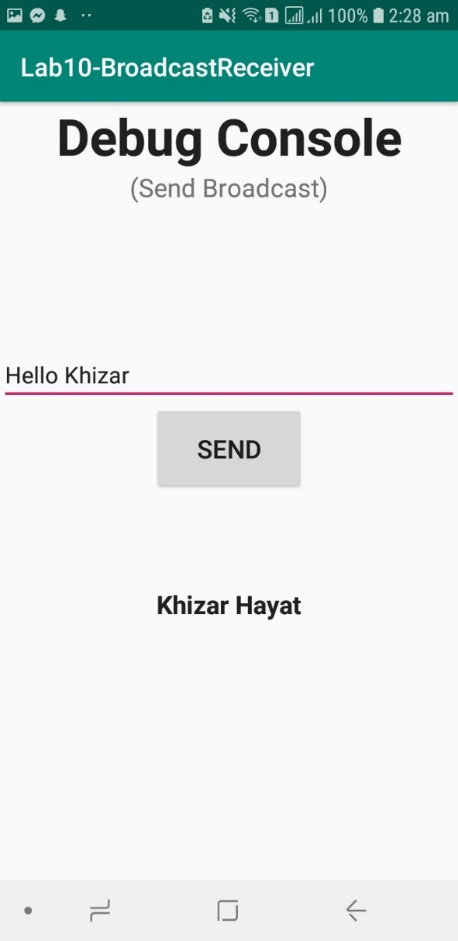
<?xml version="1.0" encoding="utf-8"?>  
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"  
  
 xmlns:tools="http://schemas.android.com/tools"  
 android:layout\_width="match\_parent"  
 android:layout\_height="match\_parent"  
 tools:context=".MainActivity">  
  
 <LinearLayout  
 android:layout\_width="match\_parent"  
 android:layout\_height="match\_parent"  
 android:weightSum="10"  
 android:orientation="vertical">  
 <TextView  
 android:id="@+id/title\_text\_view"  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:text="Debug Console"  
 android:textSize="40dp"  
 android:textAppearance="?android:textAppearanceLarge"  
 android:textStyle="bold"  
 android:textAlignment="center"/>  
 <TextView  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:layout\_weight="0.5"  
 android:text="(Send Broadcast)"  
 android:textAlignment="center"  
 android:textSize="20dp"/>  
 <TextView  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:layout\_weight="2"/>  
 <EditText  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:hint="Enter you Debug message"  
 android:id="@+id/text" />  
 <Button  
 android:id="@+id/btnSend"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:textAlignment="center"  
 android:layout\_gravity="center"  
 android:textSize="20dp"  
 android:height="70dp"  
 android:width="120dp"  
 android:text=" Send "  
 android:onClick="sendOutBroadcast"/>  
 <TextView  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:text="Khizar Hayat"  
 android:textSize="20dp"  
 android:textAppearance="?android:textAppearanceLarge"  
 android:textStyle="bold"  
 android:gravity="center"  
 android:textAlignment="center"  
 android:layout\_weight="4"/>  
 </LinearLayout>  
  
</RelativeLayout>

**JAVA:**

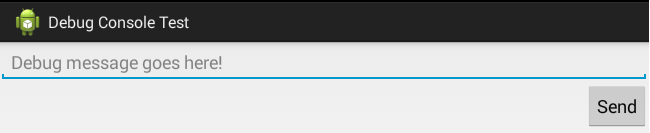
**(SenderBroadcast)**

public class MainActivity extends AppCompatActivity {  
  
 @Override  
 protected void onCreate(Bundle savedInstanceState) {  
 super.onCreate(savedInstanceState);  
 setContentView(R.layout.*activity\_main*);  
 }  
  
 public void sendOutBroadcast(View view){  
 Intent in = new Intent();  
 in.setAction("com.example.lab10\_broadcastreceiver");  
 in.addFlags(Intent.*FLAG\_INCLUDE\_STOPPED\_PACKAGES*);  
 sendBroadcast(in);  
 }  
}

**Output:**



**Task#2:** Create an application which can send broadcast to the DebugConsole (above application). You may use the following UI as reference:



* On pressing send button a broadcast with action DEBUG\_INFO must be sent to the system. That broadcast must contain application name as well as message, given by user, to be displayed on DebugConsole.

**Solution:**

**XML:**

<?xml version="1.0" encoding="utf-8"?>  
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"  
  
 xmlns:tools="http://schemas.android.com/tools"  
 android:layout\_width="match\_parent"  
 android:layout\_height="match\_parent"  
 tools:context=".MainActivity">  
  
 <LinearLayout  
 android:layout\_width="match\_parent"  
 android:layout\_height="match\_parent"  
 android:weightSum="10"  
 android:orientation="vertical">  
 <TextView  
 android:id="@+id/title\_text\_view"  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:text="Debug Console"  
 android:textSize="40dp"  
 android:textAppearance="?android:textAppearanceLarge"  
 android:textStyle="bold"  
 android:textAlignment="center"/>  
 <TextView  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:layout\_weight="0.5"  
 android:text="(Receive Broadcast)"  
 android:textAlignment="center"  
 android:textSize="20dp"/>  
 <TextView  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content" />  
 <TextView  
 android:id="@+id/console"  
 android:layout\_width="match\_parent"  
 android:layout\_height="match\_parent"  
 android:textSize="20dp"  
 android:textAppearance="?android:textAppearanceLarge"  
 android:textStyle="bold"  
 android:background="@color/colorBlue"  
 android:textColor="@color/colorBlack" />  
 </LinearLayout>  
  
</RelativeLayout>

**JAVA:**

**(ReceiverBroadcast)**

public class ReceiverBroadcast extends BroadcastReceiver {  
  
 public ReceiverBroadcast(){  
  
 }  
  
 @Override  
 public void onReceive(Context context, Intent intent) {  
 Toast.*makeText*(context,"Broadcast Received!",Toast.*LENGTH\_SHORT*).show();  
 }  
}

**Output:**

