```xml
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="vertical"
    tools:context=".MainActivity" >

    <EditText
        android:id="@+id/edit1"
        android:layout_marginTop="50dp"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:inputType="text"
        android:hint="전달할 정보 입력" />

    <Button
        android:id="@+id/button1"
        android:layout_marginTop="50dp"
        android:layout_gravity="center"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="보내기" />

    <TextView
        android:id="@+id/textView1"
        android:layout_marginTop="50dp"
        android:textSize="20sp"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:text="받은 정보 : " />
</LinearLayout>
```



```java
public class MainActivity extends AppCompatActivity {
    EditText edit1;
    Button button1;
    TextView textView1;

    // 동적 리시버 : 등록과 해제가 자유롭다.(메니페스트에 등록하지 않는다.)
    // 리시버를 내부클래스로 만든다.
    MyReceiver receiver ;
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
        edit1 = findViewById(R.id.edit1);
        button1 = findViewById(R.id.button1);
        textView1 = findViewById(R.id.textView1);

        button1.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View view) {
                String input = edit1.getText().toString();
                Intent intent = new Intent(getApplicationContext(), MyService.class);
                intent.putExtra("msg", input);

                startService(intent);
            }
        });
        receiver = new MyReceiver();
        IntentFilter filter = new IntentFilter();
        filter.addAction("TEST");
        registerReceiver(receiver,filter);
    }

    class MyReceiver extends BroadcastReceiver{
        @Override
        public void onReceive(Context context, Intent intent) {
            String msg = intent.getStringExtra("msg");
            textView1.append(msg);
        }
    }
}
```



```java
public class MyService extends Service {
    @Override
    public void onCreate() {
        super.onCreate();
    }

    @Override
    public int onStartCommand(Intent intent, int flags, int startId) {
        String msg = intent.getStringExtra("msg");

        // 리시버에 넘길 인텐트 만들어서 처리 하기
        Intent go_intent = new Intent();
        go_intent.setAction("TEST");
        go_intent.putExtra("msg", msg);

        // 리시버 호출
        sendBroadcast(go_intent);

        return super.onStartCommand(intent, flags, startId);
    }

    @Nullable
    @Override
    public IBinder onBind(Intent intent) {
        return null;
    }
}
```

```xml
<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
package="com.ict.ex70_receiver2">

<application
android:allowBackup="true"
android:icon="@mipmap/ic_launcher"
android:label="@string/app_name"
android:roundIcon="@mipmap/ic_launcher_round"
android:supportsRtl="true"
android:theme="@style/AppTheme">
<activity android:name=".MainActivity">
<intent-filter>
<action android:name="android.intent.action.MAIN" />

<category android:name="android.intent.category.LAUNCHER" />
</intent-filter>
</activity>

<service android:name=".MyService"
android:enabled="true" android:exported="true"/>
</application>

</manifest>
```