

02

## 리스트 뷰(ListView)



## 리스트 뷰(ListView)란?

- ◆ 리스트 뷰는 어댑터 뷰의 대표 위젯으로서, 복수 개의 항목을 수직으로 표시
- ◆ 리스트 뷰 설정 절차

1 XML 레이아웃에 ListView 정의

2 어댑터 생성

3 ListView 객체에 어댑터 연결

- `void setAdapter(T adapter)`





## 1. XML 레이아웃에 ListView 정의

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

    <ListView
        android:id="@+id/listView"
        android:layout_width="match_parent"
        android:layout_height="match_parent"
    />
</LinearLayout>
```

## 2 리스트 뷰(ListView)



### 2. 어댑터 생성

- ◆ 데이터 원본이 배열인 경우에 **ArrayAdapter** 객체 사용

**ArrayAdapter(Context context, int textViewResourceId, T[] objects)**



리소스 ID	설명
android.R.layout.simple_list_item_1	하나의 텍스트 뷰로 구성된 레이아웃
android.R.layout.simple_list_item_2	두 개의 텍스트 뷰로 구성된 레이아웃
android.R.layout.simple_list_item_checked	오른쪽에 체크 표시가 나타남
android.R.layout.simple_list_item_single_choice	오른쪽에 라디오 버튼이 나타남
android.R.layout.simple_lsit_item_multiple_choice	오른쪽에 체크 버튼이 나타남



## 배열과 ArrayAdapter 객체를 사용한 리스트 뷰 예제

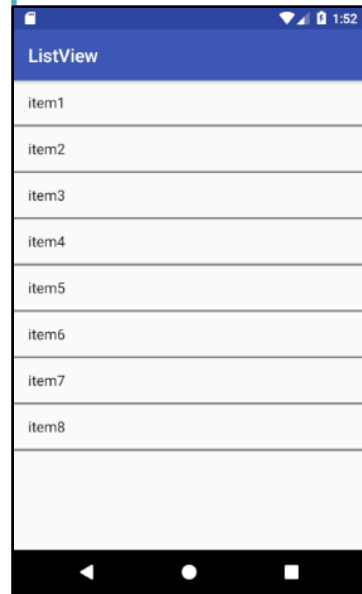
```
public class MainActivity extends AppCompatActivity {  
    protected void onCreate(Bundle savedInstanceState) {  
        super.onCreate(savedInstanceState);  
        setContentView(R.layout.activity_main);  
  
        //어댑터 생성  
        ListAdapter adapt = createAdapter();  
  
        //어댑터 연결  
        ListView list = (ListView) findViewById(R.id.listView);  
        list.setAdapter(adapt);  
  
        //ListView 기타 설정  
        list.setDivider(new ColorDrawable(Color.GRAY));  
        list.setDividerHeight(5);  
    }  
}
```

## 2 리스트 뷰(ListView)



### 배열과 ArrayAdapter 객체를 사용한 리스트 뷰 예제

```
private ListAdapter createAdapter() {  
  
    // 데이터 원본 준비  
    String[] items = {"item1", "item2", "item3", "item4", "item5", "item6",  
                      "item7", "item8"};  
  
    //어댑터 준비 (배열 객체 이용, simple_list_item_1 리소스 사용)  
    ArrayAdapter<String> adapt  
        = new ArrayAdapter<String>(  
            this,  
            android.R.layout.simple_list_item_1,  
            items);  
  
    return adapt;  
}
```



## 2 리스트 뷰(ListView)



### 문자열 배열 리소스 사용법

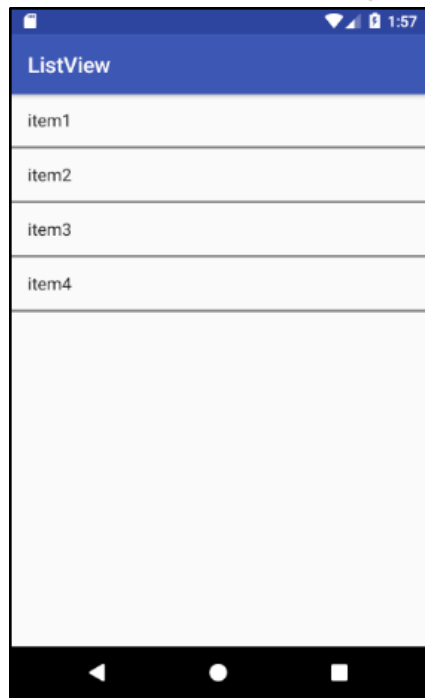
- ◆ 고정적인 문자열의 경우, 리소스에 정의해 놓고 사용

[vaules/arrays.xml](#)

```
<?xml version="1.0" encoding="utf-8"?>
<resources>
    <string-array name="items">
        <item> item1</item>
        <item> item2</item>
        <item> item3</item>
        <item> item4</item>
    </string-array>
</resources>
```

- ◆ 리소스로부터 어댑터 생성

```
ArrayAdapter<CharSequence> adapt = ArrayAdapter.createFromResource(this,
R.array.items, android.R.layout.simple_list_item_1);
```



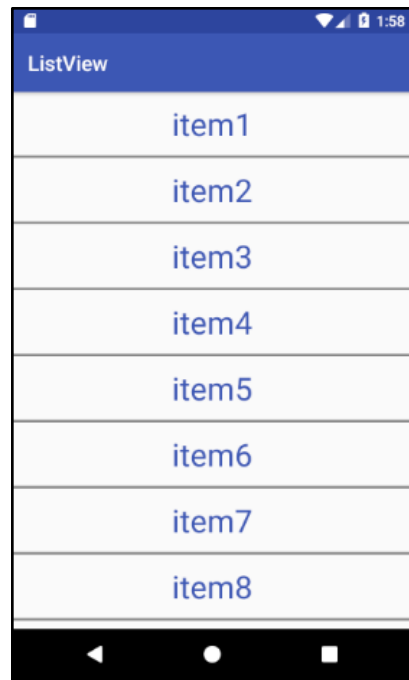


## 사용자 정의 ListView 템플릿 레이아웃

- ◆ ListView 항목의 템플릿 역할을 담당할 /res/layout/item.xml 생성

### item.xml

```
<TextView
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:textSize="@dimen/menu_item_size"
    android:textColor="@color/colorPrimary"
    android:gravity="center"
    android:padding="10dp" />
```



- ◆ 정의된 ListView 템플릿 레이아웃을 이용하여 ArrayAdapter 준비

```
ArrayAdapter<String> adapt = new ArrayAdapter<String>(this, R.layout.item, items);
```