

RecyclerView

Bağımlılıkları Dahil Etme

```
dependencies {  
    implementation 'com.android.support:recyclerview-v7:28.0.0'  
}
```

XML Dosyasında Tanımlama

```
<?xml version="1.0" encoding="utf-8"?>  
<!-- A RecyclerView with some commonly used attributes -->  
<android.support.v7.widget.RecyclerView  
    android:id="@+id/my_recycler_view"  
    android:scrollbars="vertical"  
    android:layout_width="match_parent"  
    android:layout_height="match_parent"/>
```

Activity Üzerinde Tanımlama

```
public class MyActivity extends Activity {  
    private RecyclerView recyclerView;  
    private RecyclerView.Adapter mAdapter;  
    private RecyclerView.LayoutManager layoutManager;  
  
    @Override  
    protected void onCreate(Bundle savedInstanceState) {  
        super.onCreate(savedInstanceState);  
        setContentView(R.layout.my_activity);  
        recyclerView = (RecyclerView) findViewById(R.id.my_recycler_view);  
  
        // use this setting to improve performance if you know that changes  
        // in content do not change the layout size of the RecyclerView  
        recyclerView.setHasFixedSize(true);  
  
        // use a linear layout manager  
        layoutManager = new LinearLayoutManager(this);  
        recyclerView.setLayoutManager(layoutManager);  
  
        // specify an adapter (see also next example)  
        mAdapter = new MyAdapter(myDataset);  
        recyclerView.setAdapter(mAdapter);  
    }  
}
```

```
// ...  
}
```

📁 Adapter Sınıfı

```
public class MyAdapter extends RecyclerView.Adapter<MyAdapter.MyViewHolder> {  
    private String[] mDataset;  
  
    // Provide a reference to the views for each data item  
    // Complex data items may need more than one view per item, and  
    // you provide access to all the views for a data item in a view holder  
    public static class MyViewHolder extends RecyclerView.ViewHolder {  
        // each data item is just a string in this case  
        public TextView textView;  
        public MyViewHolder(TextView v) {  
            super(v);  
            textView = v;  
        }  
    }  
  
    // Provide a suitable constructor (depends on the kind of dataset)  
    public MyAdapter(String[] myDataset) {  
        mDataset = myDataset;  
    }  
  
    // Create new views (invoked by the layout manager)  
    @Override  
    public MyAdapter.MyViewHolder onCreateViewHolder(ViewGroup parent,  
                                                    int viewType) {  
        // create a new view  
        TextView v = (TextView) LayoutInflater.from(parent.getContext())  
            .inflate(R.layout.my_text_view, parent, false);  
        ...  
        MyViewHolder vh = new MyViewHolder(v);  
        return vh;  
    }  
  
    // Replace the contents of a view (invoked by the layout manager)  
    @Override  
    public void onBindViewHolder(MyViewHolder holder, int position) {  
        // - get element from your dataset at this position  
        // - replace the contents of the view with that element  
        holder.textView.setText(mDataset[position]);  
    }  
  
    // Return the size of your dataset (invoked by the layout manager)  
    @Override  
    public int getItemCount() {  
        return mDataset.length;  
    }  
}
```

```
}  
}
```

Faydalı Bağlantılar

{% embed url="https://developer.android.com/guide/topics/ui/layout/recyclerview" %}