

WordListAdapter.java

```
package crispyid.co.id.recyclerview;

import android.content.Context;
import android.support.v7.widget.RecyclerView;
import android.view.LayoutInflater;
import android.view.View;
import android.view.ViewGroup;
import android.widget.TextView;

import java.util.LinkedList;

/**
 * Shows how to implement a simple Adapter for a RecyclerView.
 * Demonstrates how to add a click handler for each item in the ViewHolder.
 */
public class WordListAdapter extends
RecyclerView.Adapter<WordListAdapter.WordViewHolder> {

    private final LinkedList<String> mWordList;
    private final LayoutInflater mInflater;

    class WordViewHolder extends RecyclerView.ViewHolder implements
View.OnClickListener {

        public final TextView wordItemView;
        final WordListAdapter mAdapter;

        /**
         * Creates a new custom view holder to hold the view to display in the
         RecyclerView.
         *
         * @param itemView The view in which to display the data.
         * @param adapter The adapter that manages the the data and views for the
         RecyclerView.
         */
        public WordViewHolder(View itemView, WordListAdapter adapter) {
            super(itemView);
            wordItemView = (TextView) itemView.findViewById(R.id.word);
            this.mAdapter = adapter;
            itemView.setOnClickListener(this);
        }

        @Override
        public void onClick(View v) {
            // All we do here is prepend "Clicked! " to the text in the view, to
verify that
            // the correct item was clicked. The underlying data does not change.
            wordItemView.setText ("Clicked! " + wordItemView.getText());
        }
    }

    public WordListAdapter(Context context, LinkedList<String> wordList) {
        mInflater = LayoutInflater.from(context);
        this.mWordList = wordList;
    }

    /**
     * Inflates an item view and returns a new view holder that contains it.
     * Called when the RecyclerView needs a new view holder to represent an item.
     *
     * @param parent The view group that holds the item views.
     * @param viewType Used to distinguish views, if more than one type of item view
     is used.
     */
}
```

WordListAdapter.java

```
    * @return a view holder.
    */
    @Override
    public WordViewHolder onCreateViewHolder(ViewGroup parent, int viewType) {
        // Inflate an item view.
        View mItemView = mInflater.inflate(R.layout.wordlist_item, parent, false);
        return new WordViewHolder(mItemView, this);
    }

    /**
     * Sets the contents of an item at a given position in the RecyclerView.
     * Called by RecyclerView to display the data at a specified position.
     *
     * @param holder The view holder for that position in the RecyclerView.
     * @param position The position of the item in the RecyclerView.
     */
    @Override
    public void onBindViewHolder(WordViewHolder holder, int position) {
        // Retrieve the data for that position.
        String mCurrent = mWordList.get(position);
        // Add the data to the view holder.
        holder.wordItemView.setText(mCurrent);
    }

    /**
     * Returns the size of the container that holds the data.
     *
     * @return Size of the list of data.
     */
    @Override
    public int getItemCount() {
        return mWordList.size();
    }
}
```