

모바일 프로그래밍 실습 보고서

학번	201411203	이름	박소영
순서	8	날짜	2018.04.05 목요일
강의자료	04 - RecyclerView & CardView		
주제	RecyclerView, CardView, Spinner		

- RecyclerView : 큰 틀은 그대로 유지하면서 데이터만 바뀌는, item을 반복적으로 보여주는 뷰.
 1. 필요한 라이브러리 추가하기

```
implementation 'com.android.support:recyclerview-v7:26.1.0'
implementation 'com.android.support:cardview-v7:26.1.0'
```

2. 아이템이 가지고 있어야 하는 데이터 class 만들기
3. 아이템을 보여줄 xml 레이아웃 만들기
4. adapter 생성하기

```
public class RecyclerViewAdapter extends
RecyclerView.Adapter<RecyclerView.ViewHolder> {
    ArrayList<RecyclerViewItem> mItems;

    public RecyclerViewAdapter(ArrayList<RecyclerViewItem> items){
        mItems = items;
    }
    @Override // 새로운 뷰 홀더 생성
    public ViewHolder onCreateViewHolder(ViewGroup parent, int viewType) {
        View view = LayoutInflater.from(parent.getContext()).inflate(
            R.layout.item_recycler_view,parent,false);
        return new ViewHolder(view);
    }
    @Override // View 의 내용을 해당 포지션의 데이터로 바꿉니다.
    public void onBindViewHolder(ViewHolder holder, int position) {
        holder.mNameTv.setText(mItems.get(position).getName());
    }
    @Override // 데이터 셋의 크기를 리턴해줍니다.
    public int getItemCount() {
        return mItems.size();
    }
    // 커스텀 뷰홀더 : item layout 에 존재하는 위젯들을 바인딩합니다.
    class ViewHolder extends RecyclerView.ViewHolder{
        private TextView mNameTv;
        public ViewHolder(View itemView) {
            super(itemView);
            mNameTv = (TextView) itemView.findViewById(R.id.itemNameTv);
        }
    }
}
```

5. MainActivity에 java 코드 작성하기

```
public class MainActivity extends AppCompatActivity {
    private String[] names = {"Charlie", "Andrew", "Han"};
    private static final int LAYOUT = R.layout.activity_main;
    private ActivityMainBinding mainBinding;
    private RecyclerView.Adapter adapter;

    private ArrayList<RecyclerView.Item> mItems = new ArrayList<>();
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        mainBinding = DataBindingUtil.setContentView(this, LAYOUT);

        setRecyclerView();
    }

    private void setRecyclerView(){
        // 각 Item 들이 RecyclerView 의 전체 크기를 변경하지 않는다면
        // setHasFixedSize() 함수를 사용해서 성능을 개선할 수 있습니다.
        // 변경될 가능성이 있다면 false 로 , 없다면 true를 설정해주세요.
        mainBinding.recyclerView.setHasFixedSize(true);

        // RecyclerView에 Adapter를 설정해줍니다.
        adapter = new RecyclerView.Adapter(mItems);
        mainBinding.recyclerView.setAdapter(adapter);

        // 다양한 LayoutManager 가 있습니다. 원하시는 방법을 선택해주세요.
        // 지그재그형의 그리드 형식
        //mainBinding.recyclerView.setLayoutManager(new
        //    StaggeredGridLayoutManager(2, StaggeredGridLayoutManager.VERTICAL));
        // 그리드 형식
        //mainBinding.recyclerView.setLayoutManager(new
        GridLayoutManager(this,4));
        // 가로 또는 세로 스크롤 목록 형식
        mainBinding.recyclerView.setLayoutManager(new LinearLayoutManager(this));

        setData();
    }

    private void setData(){
        mItems.clear();
        // RecyclerView 에 들어갈 데이터를 추가합니다.
        for(String name : names){
            mItems.add(new RecyclerView.Item(name));
            mItems.add(new RecyclerView.Item(name));
        }
        // 데이터 추가가 완료되었으면 notifyDataSetChanged() 메서드를 호출해
        // 데이터 변경 체크를 실행합니다.
        adapter.notifyDataSetChanged();
    }
}
```

java 소스코드 : MyData.java

```
public class MyData {
    String m_name;
    String m_phone;
    boolean m_wifi;
    public MyData(String name, String phone, boolean wifi){
        m_name = name;
        m_phone = phone;
        m_wifi = wifi;
    }
    public String getName(){
        return m_name;
    }
    public String getPhone(){
        return m_phone;
    }
    public boolean getWifi(){
        return m_wifi;
    }
}
```

java 소스코드 : MainActivity.java

```
public class MainActivity extends AppCompatActivity {
    List<MyData> m_data;
    EditText m_name, m_phone;
    Spinner spinner;
    boolean m_wifi;
    RecyclerView recyclerView;
    RecyclerView.LayoutManager layoutManager;
    MyAdapter adapter;
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
        init();
    }

    public void init(){
        m_name = (EditText)findViewById(R.id.name);
        m_phone = (EditText)findViewById(R.id.phoneNum);
        spinner = (Spinner)findViewById(R.id.spinner_wifi);
        spinner.setOnItemClickListener(new
```

```

        AdapterView.OnItemClickListener() {
            @Override
            public void onItemClick(AdapterView<?> parent, View view,
                int position, long id) {
                if(position==0) m_wifi =true
                else m_wifi = false;
            }

            @Override
            public void onNothingSelected(AdapterView<?> parent) {}
        });
        m_data = new ArrayList<>();
        recyclerView = (RecyclerView)findViewById(R.id.listView);
        layoutManager = new LinearLayoutManager(this,
            LinearLayoutManager.VERTICAL, false);
        recyclerView.setLayoutManager(layoutManager);
        adapter = new MyAdapter(m_data);
        recyclerView.setAdapter(adapter);
    }

    public void onSave(View view) {
        String name = m_name.getText().toString();
        String phone = m_phone.getText().toString();
        if(name.length() > 0 && phone.length() > 0){
            m_data.add(new MyData(name, phone, m_wifi));
            adapter.notifyDataSetChanged();
        }
    }
}

```

java 소스코드 : MyAdapter.java

```

public class MyAdapter extends RecyclerView.Adapter<MyAdapter.ViewHolder>{
    List<MyData> m_data;
    public MyAdapter(List<MyData> data){
        m_data = data;
    }
    @Override
    public ViewHolder onCreateViewHolder(ViewGroup parent, int viewType) {
        View v = LayoutInflater.from(parent.getContext())
            .inflate(R.layout.card_layout, parent, false);
        ViewHolder vh = new ViewHolder(v);
        return vh;
    }
}

```

```

@Override
public void onBindViewHolder(ViewHolder holder, int position) {
    holder.r_phone.setText(m_data.get(position).getPhone());
    holder.r_name.setText(m_data.get(position).getName());
    if(m_data.get(position).getWifi())
        holder.r_imgview.setImageResource(R.drawable.coffee_wifi);
    else
        holder.r_imgview.setImageResource(R.drawable.coffee_nowifi);
    holder.index = position;
}

@Override
public int getItemCount() { return m_data.size(); }

class ViewHolder extends RecyclerView.ViewHolder implements
View.OnClickListener{
    ImageView r_imgview;
    TextView r_name, r_phone;
    int index;

    public ViewHolder(View itemView){
        super(itemView);
        r_imgview = itemView.findViewById(R.id.listicon);
        r_name = itemView.findViewById(R.id.c_name);
        r_phone = itemView.findViewById(R.id.c_phone);
        itemView.setOnClickListener(this);
    }

    @Override
    public void onClick(View v) {
        Toast.makeText(v.getContext(), m_data.get(index).getName(),
            Toast.LENGTH_SHORT).show();
    }
}
}

```

Layout : activity_main.xml

```

<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:id="@+id/activity_main"
    android:orientation="vertical"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:padding="10dp">

```

```

<Spinner
    android:id="@+id/spinner_wifi"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:spinnerMode="dialog"
    android:prompt="@string/wifi_prompt"
    android:entries="@array/wifi_array"/>
<Button
    android:id="@+id/save"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:onClick="onSave"
    android:text="저장하기"/>
<android.support.v7.widget.RecyclerView
    android:id="@+id/listView"
    android:layout_width="match_parent"
    android:layout_height="match_parent"/>
</LinearLayout>

```

Values : strings.xml

```

<resources>
    <string name="app_name">2018_04_05</string>
    <string name="spin_prompt">선택</string>
    <string name="wifi_prompt">선택하세요</string>
    <string-array name="wifi_array">
        <item>wifi</item>
        <item>nowifi</item>
    </string-array>
</resources>

```

Layout : card_layout.xml

```

<?xml version="1.0" encoding="utf-8"?>
<android.support.v7.widget.CardView
    xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    app:cardCornerRadius="25dp"
    app:cardElevation="15dp"
    android:layout_marginBottom="10dp"
    android:layout_width="match_parent"
    android:layout_height="wrap_content">

    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:orientation="horizontal">

```

```

<ImageView
    android:id="@+id/listicon"
    android:layout_width="50dp"    android:layout_height="50dp"
    android:layout_margin="10dp"    android:padding="10dp"
    android:paddingLeft="10dp"    android:paddingRight="10dp" />

<LinearLayout
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:gravity="center_vertical"
    android:orientation="vertical">
    <TextView
        android:id="@+id/c_name"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:gravity="center_vertical"
        android:textSize="15sp"    android:textStyle="bold" />
    <TextView
        android:id="@+id/c_phone"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:textSize="15sp"    android:autoLink="all" />
    </LinearLayout>
</LinearLayout>
</android.support.v7.widget.CardView>

```

실행 결과

