* **Class StatesItems**

public class StatesItems {

//information that each item is going to hold

private int mStateFlag;

private String mNameState;

private String mManual;

//Constructorz

public StatesItems(int imageResource, String text1, String text2) {

mStateFlag = imageResource;

mNameState = text1;

mManual = text2;

}

**Description**

We use this class to create an object that will contain the information to populate our card views in the RecyclerView.

**Parameters**

Imageresource: it will hold the id of the flag image

Text1: it holds the name of the state

Text2: it holds the text manual in it

**What it returns**

It can be used to create an object of the class StatesItems

* **Class Manuals**

public class Manuals extends AppCompatActivity {

// set title and the content view to its layout

@Override

protected void onCreate(Bundle savedInstanceState) {

super.onCreate(savedInstanceState);

setContentView(R.layout.*activity\_manuals*);

setTitle("Manuals");

}

} }

**Description**

It populates the current activity with the views from the activity\_manuals layout

**What it returns**

The manuals activity populated by the layout

* **For all Fragments**

public class HomeFragment extends Fragment {

//Override onCreateView, set title of the current fragment

//and inflates the content for this fragment

@Nullable

@Override

public View onCreateView(@NonNull LayoutInflater inflater, @Nullable ViewGroup container, @Nullable Bundle savedInstanceState) {

getActivity().setTitle("Home");

return inflater.inflate(R.layout.*frag\_home*,container,false);

}

}

**Description**

It inflates the content to show on the current fragment

**Parameters**

We override the onCreateView method, it automatically loads the parameters needed.

**What it returns**

The fragment populated with data

* **Class Adapter**

[The code is too long to be displayed on here if you want to know more about it please check the source code, open the Adapter class file and read the comments]

**Description**

It gets the data passed from our array of state items and applies it on the recyclerView so the info can be displayed

**Methods**

* setOnItemClickListener: sets the parameter provided to set the info to the @mListener variable
* ViewHolder: Here we hold the views that we are going to be pass information to
* onCreateViewHolder method: it inflates the content the layout that is going to contain the information about our items
* onBindViewHolder: it binds or fills the info that it is going to be shown in

the current state item by its position

* getItemCount method: Here we define how many items are going to be in our list

**What it returns**

It does not return anything, but instead, handle the data that is going to be applied to the RecyclerView class.

* **Class Main Activity**

[The code is too long to be displayed on here if you want to know more about it please check the source code, open the Adapter class file and read the comments]

**Description**

It is the main class or activity that takes care of switching between fragments and also handles the switch between other activities when an user clicks any button under the main menu or a fragment.

**Methods**

* onCreate method: Takes care of creating and inflating the views on the respective layout
* BottomNavigationView.OnNavigationItemSelectedListener method: it gets the id of the item clicked to be changed.
* switchingFrag: It handles the intent to switch between the different fragments when the id is passed

**What it returns**

It is the foundation of our app, This class is what it contains most the majority of our app actions. It is the main interface where most of the actions take place.

* **Class RecyclerViewMain**

[The code is too long to be displayed on here if you want to know more about it please check the source code, open the Adapter class file and read the comments]

**Description**

It populates the array with data that is going to be shown in the manual section

and passes it to the adapter which applies the data to our RecyclerView and then the recycle view is passed to our LayoutManager.

**Methods**

* onCreate method: Here we initialized our array list of state items and populate it. Also, the recyclerView, the layout manager and the adapter. we applied the adapter to our recyclerView and the recyclerView to our LayoutManager
* browserIntent: It handles the intent that takes the user to the webpage to the respective states’ DMV manual.

**What it returns**

As a result, the data passed in our array list is displayed in our state manual section with a listener that takes the user to the respective DMV website.