First\_activity.java

**package** com.example.recipemng;  
  
**import** android.content.Intent;  
**import** androidx.appcompat.app.AppCompatActivity;  
**import** android.os.Bundle;  
**import** android.view.View;  
**import** android.content.res.Resources;  
**import** android.widget.Button;  
**import** android.widget.TextView;  
  
  
**public class** FirstActivity **extends** AppCompatActivity {  
 **private** Button **btn\_make**, **btn\_cake**;  
 TextView **header**;  
  
 @Override  
 **protected void** onCreate(Bundle saveInstanceState)  
 {  
 **super**.onCreate(saveInstanceState);  
 setContentView(R.layout.***activity\_first***);  
 **header** = findViewById(R.id.***header***);  
 **final** Resources resources = getResources();  
  
 **btn\_cake**=(Button)findViewById(R.id.***btn\_cake***);  
 **btn\_cake**.setOnClickListener(**new** View.OnClickListener(){  
 @Override  
 **public void** onClick(View view){  
 openGameActivity(**"cake"**);  
 }  
 });  
  
 **btn\_make**=(Button)findViewById(R.id.***btn\_make***);  
 **btn\_make**.setOnClickListener(**new** View.OnClickListener(){  
 @Override  
 **public void** onClick(View view){  
 openGameActivity(**"make"**);  
 }  
 });  
 }  
 **public void** openGameActivity(String recipe){  
 Intent intent=**new** Intent(FirstActivity.**this**,MainActivity.**class**);  
 intent.putExtra(**"recipe"**,recipe);  
 startActivity(intent);  
 }  
}

MainActivity.java

**package** com.example.recipemng;  
*//https://console.firebase.google.com/project/recipemanage/database/recipemanage/data***import** android.app.Fragment;  
**import** android.app.FragmentManager;  
**import** android.app.FragmentTransaction;  
**import** android.app.SearchManager;  
**import** android.content.Intent;  
**import** android.database.Cursor;  
**import** android.graphics.Bitmap;  
**import** android.net.Uri;  
**import** android.os.Bundle;  
  
**import** androidx.appcompat.app.AppCompatActivity;  
**import** android.provider.DocumentsContract;  
**import** android.provider.MediaStore;  
**import** android.text.TextUtils;  
**import** android.util.Log;  
**import** android.view.View;  
**import** android.widget.Button;  
**import** android.widget.EditText;  
**import** android.widget.ImageView;  
**import** android.widget.TextView;  
**import** android.widget.Toast;  
  
**import** com.google.firebase.database.DataSnapshot;  
**import** com.google.firebase.database.DatabaseError;  
**import** com.google.firebase.database.DatabaseReference;  
**import** com.google.firebase.database.FirebaseDatabase;  
**import** com.google.firebase.database.ValueEventListener;  
  
**import** java.io.IOException;  
  
**public class** MainActivity **extends** AppCompatActivity {  
  
 *// main activity is the main page associated with creating new recipes* **private static final** String ***TAG*** = MainActivity.**class**.getSimpleName();  
 **private** TextView **textInfo**, **textIngridients**;  
 **private** EditText **inputName**, **inputInstructions**, **urlString**;  
 **private** Button **btnSave**, **btnLoad**, **btnNext**, **btnShowMore**;  
 **private** ImageView **imageView**;  
 **private** DatabaseReference **mFirebaseDatabase**;  
 **private** FirebaseDatabase **mFirebaseInstance**;  
 **private** String **itemId**;  
 **private** Fragment **fr**;  
 **private** Uri **filePath**;  
 **private final int PICK\_IMAGE\_REQUEST** = 71;  
 String **recipe**; *//* @Override  
 **protected void** onCreate(Bundle savedInstanceState) {  
 **super**.onCreate(savedInstanceState);  
 setContentView(R.layout.***activity\_main***);  
 Intent intent = getIntent();  
 **recipe** = intent.getStringExtra(**"recipe"**); *// are we handling bakes or makes?* Log.*e*(***TAG***, **"running main activity with recipe $"** + **recipe** + **"$"**);  
  
 setContentView(R.layout.***activity\_main***);  
 getSupportActionBar().setDisplayShowHomeEnabled(**true**);  
 getSupportActionBar().setIcon(R.mipmap.***ic\_launcher***);  
 handleIntent(getIntent());  
  
 *// invoking fragment GetIngridients from MainActivity, passing "1" to denote Main page rather than Input page* Bundle bundle=**new** Bundle();  
 bundle.putString(**"activity"**, **"1"**);  
 bundle.putString(**"recipe"**, **recipe**);  
  
 **fr** = **new** GetIngridients();  
 **fr**.setArguments(bundle);  
 FragmentManager fm = getFragmentManager();  
 FragmentTransaction fragmentTransaction = fm.beginTransaction();  
 fragmentTransaction.replace(R.id.***fragment\_place***, **fr**);  
 fragmentTransaction.commit();  
  
 *// initializing display variables* **textInfo** = findViewById(R.id.***info***);  
 **inputName** = findViewById(R.id.***name***);  
 **textIngridients** = findViewById(R.id.***ingridients***);  
 **urlString** = findViewById(R.id.***imageUrl***);  
 **inputInstructions** = findViewById(R.id.***instructions***);  
  
 **btnSave** = findViewById(R.id.***btn\_save***);  
 **btnLoad** = findViewById(R.id.***btn\_load***);  
 **btnNext** = findViewById(R.id.***btn\_next***);  
 **btnShowMore** = (Button) findViewById(R.id.***btn\_Showmore***);  
 **imageView** = (ImageView) findViewById(R.id.***imgView***);  
  
 *// initializing "Realtime Datrabase" firebase RecipeManage - MayaRecipe* **mFirebaseInstance** = FirebaseDatabase.*getInstance*();  
 **mFirebaseDatabase** = **mFirebaseInstance**.getReference(**"recipemanage"**);  
 **mFirebaseInstance**.getReference(**"app\_title"**).setValue(**"Maya Recipe"**);  
 **mFirebaseInstance**.getReference(**"app\_title"**).addValueEventListener(**new** ValueEventListener() {  
 @Override  
 **public void** onDataChange(DataSnapshot dataSnapshot) {  
 Log.*e*(***TAG***, **"App title updated"**);  
 String appTitle = dataSnapshot.getValue(String.**class**);  
 getSupportActionBar().setTitle(appTitle);  
 }  
 @Override  
 **public void** onCancelled(DatabaseError error) {  
 Log.*e*(***TAG***, **"Failed to read app title value."**, error.toException());  
 }  
 });  
  
 *// defining button - next to arrive to find a recipe* **btnNext**.setOnClickListener(**new** View.OnClickListener() {  
 @Override  
 **public void** onClick(View view) {  
 reLoadFragment();  
 Intent intent=**new** Intent(MainActivity.**this**,InputActivity.**class**);  
 intent.putExtra(**"recipe"**,**recipe**);  
 startActivity(intent);  
 }  
 });  
  
 *// defining button - showmore to expand instructions tab* **btnShowMore**.setOnClickListener(**new** View.OnClickListener() {  
 @Override  
 **public void** onClick(View view) {  
 **if** (**btnShowMore**.getText().toString().equalsIgnoreCase(**"Show More"**))  
 {  
 **inputInstructions**.setMaxLines(Integer.***MAX\_VALUE***);*//your TextView* **inputInstructions**.setSelection(**inputInstructions**.getText().length());  
 **btnShowMore**.setText(**"Show Less"**);  
 }  
 **else** {  
 **inputInstructions**.setMaxLines(1);*//collapse instructions to 1 line* **inputInstructions**.setSelection(0); *// set focus on first character so that 1st line is shown* **btnShowMore**.setText(**"Show More"**);  
 }  
 }  
 });  
  
 */\*  
 // defining button - display an image by the url of the urlstring field - note if not used, no need for picassotrustall  
 // invokes a picassotrustall class that overrides picasso in order to support also http files (youtibe https issues  
 btnDisplay.setOnClickListener(new View.OnClickListener() {  
 @Override  
 public void onClick(View v) {  
 String pic = urlString.getText().toString();  
 if (TextUtils.isEmpty(pic))  
 textInfo.setText("cannot display an empty url");  
 else  
 PicassoTrustAll.getInstance(MainActivity.this).load(pic).into(imageView);  
  
 }  
 });  
 \*/  
 // set hint when there is focus on url string* **urlString**.setOnFocusChangeListener(**new** View.OnFocusChangeListener() {  
 @Override  
 **public void** onFocusChange(View v, **boolean** hasFocus) {  
 **if** (hasFocus) {  
 **urlString** .setHint(**"http://pngimg.com/uploads/cake/cake\_PNG13143.png"**);  
 } **else** {  
 **urlString** .setHint(**""**);  
 }  
 }  
 });  
  
 *// defining button - load an image from gallery - calls startActivity and sets urlString* **btnLoad**.setOnClickListener(**new** View.OnClickListener() {  
 @Override  
 **public void** onClick(View view) {  
 Intent intent = **new** Intent();  
 intent.setType(**"image/\*"**);  
 intent.setAction(Intent.***ACTION\_GET\_CONTENT***);  
 startActivityForResult(Intent.*createChooser*(intent, **"Select Picture"**), **PICK\_IMAGE\_REQUEST**);  
 }  
 });  
  
 *// defining button - save to save data to firebase (or update if itemId exists)  
 // name must be not-null to save and item. the rest may be null.  
 // calls createItem or updateItem* **btnSave**.setOnClickListener(**new** View.OnClickListener() {  
 @Override  
 **public void** onClick(View view) {  
 Log.*e*(***TAG***, **"btn save is called"**);  
 String name = **inputName**.getText().toString();  
 String ingridients = **textIngridients**.getText().toString();  
 String imageUrl = **urlString**.getText().toString();  
 String instructions = **inputInstructions**.getText().toString();  
  
 **if** (TextUtils.*isEmpty*(**itemId**)) {  
 **if** (TextUtils.*isEmpty*(name))  
 **textInfo**.setText(**"cannot save an empty entry"**);  
 **else** {  
 **textInfo**.setText(**"saving data"**);  
 createItem(name, ingridients, imageUrl, instructions);  
 }  
 } **else** {  
 **textInfo**.setText(**"updating data"** + **itemId**);  
 updateItem(name, ingridients, imageUrl, instructions);  
 }  
 }  
 });  
 }  
  
 @Override  
 **protected void** onNewIntent(Intent intent) {  
 handleIntent(intent);  
 }  
 **private void** handleIntent(Intent intent) {  
 **if** (Intent.***ACTION\_SEARCH***.equals(intent.getAction())) {  
 String query = intent.getStringExtra(SearchManager.***QUERY***);  
 *//use the query to search your data somehow* }  
 }  
  
 *// function to reload fragment getIngridients in order too clear history of used ingridients view item (squares)* **public void** reLoadFragment() {  
 **inputName**.setText(**""**);  
 **textIngridients**.setText(**""**);  
 **inputInstructions**.setText(**""**);  
 **itemId** = **""**;  
  
 Bundle bundle=**new** Bundle();  
 bundle.putString(**"activity"**, **"1"**);  
 bundle.putString(**"recipe"**, **recipe**);  
 **fr** = **new** GetIngridients();  
 **fr**.setArguments(bundle);  
 getFragmentManager().beginTransaction().replace(R.id.***fragment\_place***, **fr**).commit();  
 Log.*e*(***TAG***, **"reloading fragment done"**);  
 }  
  
 *// called by btnLoad onActivity - used to fetch image from gallery of pictures in cellphone* @Override  
 **protected void** onActivityResult(**int** requestCode, **int** resultCode, Intent data) {  
 **super**.onActivityResult(requestCode, resultCode, data);  
 **if**(requestCode == **PICK\_IMAGE\_REQUEST** && resultCode == ***RESULT\_OK*** && data != **null** && data.getData() != **null** ) {  
 **filePath** = data.getData();  
 **try** {  
 Bitmap bitmap = MediaStore.Images.Media.*getBitmap*(getContentResolver(), **filePath**);  
 **imageView**.setImageBitmap(bitmap);  
 **urlString**.setText(getRealPathFromURI(**filePath**));  
 }  
 **catch** (IOException e) {  
 e.printStackTrace();  
 }  
 }  
 }  
  
 *// get full path of gallery pic* **public** String getRealPathFromURI(Uri uri){  
 String filePath = **""**;  
 String wholeID = DocumentsContract.*getDocumentId*(uri);  
  
 String id = wholeID.split(**":"**)[1];  
 String[] column = { MediaStore.Images.Media.***DATA*** };  
 String sel = MediaStore.Images.Media.***\_ID*** + **"=?"**;  
  
 Cursor cursor = getApplicationContext().getContentResolver().query(MediaStore.Images.Media.***EXTERNAL\_CONTENT\_URI***,  
 column, sel, **new** String[]{ id }, **null**);  
  
 **int** columnIndex = cursor.getColumnIndex(column[0]);  
 **if** (cursor.moveToFirst()) {  
 filePath = cursor.getString(columnIndex);  
 }  
 Log.*e*(***TAG***, **"converting camera path from "**+uri+**" to "**+filePath);  
 cursor.close();  
 **return** filePath;  
 }  
  
 *// saves (with addItemChangeListener) and item into the firebase real-time database* **private void** createItem(String name,String ingridients,String imageUrl, String instructions) {  
 **if** (TextUtils.*isEmpty*(**itemId**)) {  
 **itemId** = **mFirebaseDatabase**.push().getKey();  
 **textInfo**.setText(**"creating Recipe: "** + **itemId**);  
 }  
 RecipeItem item = **new** RecipeItem(name, ingridients, imageUrl, instructions);  
 **mFirebaseDatabase**.child(**itemId**).setValue(item);  
 addItemChangeListener();  
 Toast.*makeText*(MainActivity.**this**, **"Recipe saved"**, Toast.***LENGTH\_SHORT***).show();  
 clear\_text();  
 }  
  
 *// clear text fields* **private void** clear\_text() {  
 **inputName**.setText(**""**);  
 **textIngridients**.setText(**""**);  
 **urlString**.setText(**""**);  
 **inputInstructions**.setText(**""**);  
 }  
  
 **private void** addItemChangeListener() {  
 **mFirebaseDatabase**.child(**itemId**).addValueEventListener(**new** ValueEventListener() {  
 @Override  
 **public void** onDataChange(DataSnapshot dataSnapshot) {  
 RecipeItem item = dataSnapshot.getValue(RecipeItem.**class**);  
 **if** (item == **null**) {  
 Log.*e*(***TAG***, **"RecipeItem data is null!"**);  
 **return**;  
 }  
 Log.*e*(***TAG***, **"RecipeItem data is changed!"** + item.**name** + **", "** + item.**ingridients** + **", "** + item.**instructions**);  
 **textInfo**.setText(item.**name** + **", "** + item.**ingridients** + **", "** + item.**instructions**);  
 reLoadFragment();  
 }  
 @Override  
 **public void** onCancelled(DatabaseError error) {  
 Log.*e*(***TAG***, **"Failed to read Recipe Item"**, error.toException());  
 }  
 });  
 }  
  
 *// updates an existing item into the firebase real-time database* **private void** updateItem(String name, String ingridients, String imageUrl, String instructions) {  
 *// updating the user via child nodes* **if** (!TextUtils.*isEmpty*(name))  
 **mFirebaseDatabase**.child(**itemId**).child(**"name"**).setValue(name);  
 **if** (!TextUtils.*isEmpty*(ingridients))  
 **mFirebaseDatabase**.child(**itemId**).child(**"ingridients"**).setValue(ingridients);  
 **if** (!TextUtils.*isEmpty*(imageUrl))  
 **mFirebaseDatabase**.child(**itemId**).child(**"imageUrl"**).setValue(imageUrl);  
 **if** (!TextUtils.*isEmpty*(instructions))  
 **mFirebaseDatabase**.child(**itemId**).child(**"instructions"**).setValue(instructions);  
  
 Toast.*makeText*(MainActivity.**this**, **"Recipe updated"**, Toast.***LENGTH\_SHORT***).show();  
 }  
 */\*  
 // handle search image. implemented but not intended for use  
 @Override  
 public boolean onCreateOptionsMenu(Menu menu) {  
 MenuInflater inflater = getMenuInflater();  
 inflater.inflate(R.menu.options\_menu, menu);  
 // Associate searchable configuration with the SearchView  
 SearchManager searchManager = (SearchManager) getSystemService(this.SEARCH\_SERVICE);  
 SearchView searchView = (SearchView) menu.findItem(R.id.search).getActionView();  
 searchView.setSearchableInfo(searchManager.getSearchableInfo(getComponentName()));  
 searchView.setIconifiedByDefault(false);  
 return true;  
 }  
 @Override  
 public boolean onOptionsItemSelected(MenuItem item) {  
 //Handle item selection  
 switch (item.getItemId()) {  
 case R.id.search:  
 Toast.makeText(getApplicationContext(), "Search button clicked", Toast.LENGTH\_SHORT).show();  
 return true;  
 case R.id.exit:  
 //perform any action;  
 return true;  
 default:  
 return super.onOptionsItemSelected(item);  
 }  
 }  
 \*/  
  
 // public function called from fragment to update textIngridients with selected fragemnt ingridients* **public void** update\_ingridients(String cur\_ingr) {  
 **textIngridients**.setText(cur\_ingr);  
 }  
}

InputActivity.java

**package** com.example.recipemng;  
*//https://console.firebase.google.com/project/recipemanage/database/recipemanage/data realtime database***import** android.app.Fragment;  
**import** android.app.FragmentManager;  
**import** android.app.FragmentTransaction;  
**import** android.content.Intent;  
**import** android.graphics.Bitmap;  
**import** android.os.Bundle;  
**import** androidx.appcompat.app.AppCompatActivity;  
  
**import** android.os.Environment;  
**import** android.util.Log;  
**import** android.view.Menu;  
**import** android.view.MenuInflater;  
**import** android.view.MenuItem;  
**import** android.view.View;  
**import** android.widget.Button;  
**import** android.widget.GridView;  
**import** android.widget.ImageView;  
**import** android.widget.TextView;  
**import** android.widget.ListView;  
  
**import** java.io.File;  
**import** java.util.ArrayList;  
**import** java.util.Collections;  
**import** java.util.HashSet;  
**import** java.util.Iterator;  
**import** java.util.List;  
**import** java.util.Set;  
  
**import** android.widget.AdapterView;  
**import** android.widget.ArrayAdapter;  
**import** android.widget.Toast;  
  
**import** com.google.firebase.database.DataSnapshot;  
**import** com.google.firebase.database.DatabaseError;  
**import** com.google.firebase.database.DatabaseReference;  
**import** com.google.firebase.database.FirebaseDatabase;  
**import** com.google.firebase.database.ValueEventListener;  
**import** com.squareup.picasso.Picasso;  
  
**public class** InputActivity **extends** AppCompatActivity {  
  
 *// this activity searches a recipe in db and displays to user* **private static final** String ***TAG*** = MainActivity.**class**.getSimpleName();  
 **private** TextView **textInfo**, **textIngridients**;  
 **private** Button **btnLoad**, **btnFind**;  
 **private** DatabaseReference **mFirebaseDatabase**;  
 **private** FirebaseDatabase **mFirebaseInstance**;  
 List<String> **items**;  
 **private** ListView **listView**;  
 **private** ImageView **imageView**;  
 **private boolean SearchWith** = **true**; *// default is search with item and not search from items* String **recipe**;  
  
 *// state global parameters fragment manager in order to allow change of fragment from input activity* Fragment **fr**;  
 FragmentManager **fm**;  
  
 @Override  
 **protected void** onCreate(Bundle savedInstanceState) {  
 **super**.onCreate(savedInstanceState);  
 Intent intent = getIntent();  
 **recipe** = intent.getStringExtra(**"recipe"**); *// are we handling bakes or makes?* setContentView(R.layout.***activity\_input***);  
 Log.*e*(***TAG***, **"running input activity with recipe $"** + **recipe** + **"$"**);  
  
 *// sets activity with fragment GetIngridients, this time by parameter 2* Bundle bundle=**new** Bundle();  
 bundle.putString(**"activity"**, **"2"**);  
 bundle.putString(**"recipe"**,**recipe**);  
 **fr** = **new** GetIngridients();  
 **fr**.setArguments(bundle);  
 **fm** = getFragmentManager();  
 FragmentTransaction fragmentTransaction = **fm**.beginTransaction();  
 fragmentTransaction.replace(R.id.***fragment\_place***, **fr**);  
 fragmentTransaction.commit();  
 *//getSupportActionBar().setDisplayShowHomeEnabled(true);  
 //getSupportActionBar().setIcon(R.mipmap.ic\_launcher);  
  
 // defining display widgets* **textInfo** = findViewById(R.id.***info***);  
 **textInfo**.setText(**"Welcome. enter ingridients to find recipe"**);  
  
 **textIngridients** = findViewById(R.id.***ingridients***);  
 **btnLoad** = (Button) findViewById(R.id.***btn\_load***);  
 **btnFind** = (Button) findViewById(R.id.***btn\_find***);  
 **imageView** = (ImageView) findViewById(R.id.***imgView***);  
  
 *// listview shows the item that were selected by eitherload or find and enable their picture display  
 // defining the search list of items from database, define activity of list item click  
 // needs to be changed that when clicking a line - opens the ingridients / updates list and adds image* **listView** = (ListView) findViewById(R.id.***list\_view***);  
 **listView**.setClickable(**true**);  
 **listView**.setOnItemClickListener(**new** AdapterView.OnItemClickListener() {  
 @Override  
 **public void** onItemClick(AdapterView<?> arg0, View arg1, **int** position, **long** arg3) {  
 Object o = **listView**.getItemAtPosition(position);  
 **textInfo**.setText(**"selecting "**+o.toString());  
  
 String textstr = **items**.get(position);  
 **int** pos1 = textstr.lastIndexOf(**"$"**);  
 **int** pos2 = textstr.lastIndexOf(**"@"**);  
 String db\_ingridients = textstr.substring(pos1+1,pos2);  
 String pic = textstr.substring(pos2+1);  
 Log.*e*(***TAG***, **"selecting for display - "** + pic);  
  
 String usr\_ingridients = **textIngridients**.getText().toString();  
 changeFragmentGridView(usr\_ingridients,db\_ingridients); *// mark in gray additional entries per selected recipe line  
  
 // if item is content - fetch from image library, else fetches from the internet* **if** (pic.startsWith(**"/storage"**)) { *// gallery* File file\_name = **new** File (pic);  
 **if**(file\_name.exists()) *// path does not exist. this is the problem.* Picasso.*with*(InputActivity.**this**).load(**"file://"** + file\_name).config(Bitmap.Config.***RGB\_565***).  
 fit().centerCrop().into(**imageView**);  
 **else** Log.*e*(***TAG***, **"file does not exist "** + file\_name.toString());  
 }  
 **else if** (pic != **null** && !pic.isEmpty()) *// internet works!* PicassoTrustAll.*getInstance*(InputActivity.**this**).load(pic).into(**imageView**);  
 **else** *// empty pic* **textInfo**.setText(**"cannot display empty picture"**);  
 }  
 });  
  
 *// initializes database* **mFirebaseInstance** = FirebaseDatabase.*getInstance*();  
 **mFirebaseDatabase** = **mFirebaseInstance**.getReference(**"recipemanage"**);  
 **mFirebaseInstance**.getReference(**"app\_title"**).setValue(**"Maya Recipe"**);  
 **mFirebaseInstance**.getReference(**"app\_title"**).addValueEventListener(**new** ValueEventListener() {  
 @Override  
 **public void** onDataChange(DataSnapshot dataSnapshot) {  
 Log.*e*(***TAG***, **"App title updated"**);  
 String appTitle = dataSnapshot.getValue(String.**class**);  
 getSupportActionBar().setTitle(appTitle);  
 }  
 @Override  
 **public void** onCancelled(DatabaseError error) {  
 Log.*e*(***TAG***, **"Failed to read app title value."**, error.toException());  
 }  
 });  
  
 *// defines load button - traverses firebase database, loads all data* **btnLoad**.setOnClickListener(**new** View.OnClickListener() {  
 @Override  
 **public void** onClick(View view) {  
 Log.*e*(***TAG***, **"btn load is called"**);  
 Fragment frag = **fm**.findFragmentById(R.id.***fragment\_place***);  
 GridView gv = frag.getView().findViewById(R.id.***gridview***);  
 GetIngridients.*clear\_selection*(gv); *// remove all marks from grid. fetching all data* **textIngridients**.setText(**""**);  
  
 ValueEventListener valueEventListener = **new** ValueEventListener() {  
 @Override  
 **public void** onDataChange(DataSnapshot dataSnapshot) {  
 fetch\_data(dataSnapshot,**true**);  
 }  
  
 @Override  
 **public void** onCancelled(DatabaseError databaseError) {  
 }  
 };  
 **mFirebaseDatabase**.addListenerForSingleValueEvent(valueEventListener);  
 }  
 });  
  
 *// defines find button - traverses firebase database, loads only matched data (fetch\_datea = false)* **btnFind**.setOnClickListener(**new** View.OnClickListener() {  
 @Override  
 **public void** onClick(View view) {  
 Log.*e*(***TAG***, **"btn find is called"**);  
  
 ValueEventListener valueEventListener = **new** ValueEventListener() {  
 @Override  
 **public void** onDataChange(DataSnapshot dataSnapshot) {  
 fetch\_data(dataSnapshot,**false**);  
 }  
  
 @Override  
 **public void** onCancelled(DatabaseError databaseError) {  
 }  
 };  
 **mFirebaseDatabase**.addListenerForSingleValueEvent(valueEventListener);  
 }  
 });  
 }  
  
 *// fetches data from firebase either all or selected by ticked ingridients  
 // called by btnLoad or btnFind* **void** fetch\_data(DataSnapshot dataSnapshot, **boolean** all) {  
 **items** = **new** ArrayList<>();  
 List<String> listview\_items = **new** ArrayList<>();  
 **for**(DataSnapshot ds : dataSnapshot.getChildren()) {  
 String key = ds.getKey();  
 String name = ds.child(**"name"**).getValue(String.**class**);  
 String ingridients = ds.child(**"ingridients"**).getValue(String.**class**);  
 String instructions = ds.child(**"instructions"**).getValue(String.**class**);  
 String url = ds.child(**"imageUrl"**).getValue(String.**class**);  
 **boolean** match = all || ingridients\_match(ingridients,**textIngridients**.getText().toString());  
 **if** (match) {  
 **items**.add(name + **"$"** + ingridients + **"@"** + url); *// complete version with urls (not for display)* listview\_items.add(**"recipe: "** + name + **":["** + ingridients + **"]"**); *// short version (for display)* Log.*d*(***TAG***, **"fetching "** + name + **" / "** + ingridients + **" / "** + url);  
 }  
 }  
 ArrayAdapter<String> adapter = **new** ArrayAdapter<String>(InputActivity.**this**, android.R.layout.***simple\_list\_item\_1***, listview\_items);  
 **if** (listview\_items.isEmpty()) {  
 **textInfo**.setText(**"Recipe array is empty"**);  
 adapter.clear();  
 adapter.notifyDataSetChanged();  
 }  
 **else** {  
 *//textInfo.setText("");* }  
 **listView**.setAdapter(adapter);  
 }  
  
 *//utility to convert a string of ingridients to a set of ingridients for manipulation of found items* Set convert\_str\_to\_set (String str) {  
 String[] array = str.split(**" "**); *// convert db ingr via array ot strings to set1* Set set = **new** HashSet();  
 Collections.*addAll*(set, array);  
 Log.*d*(***TAG***, **"set "**+ str + **" "** + set.toString());  
 **return** set;  
 }  
 *// tests if selected ingridients are subset of firebase ingridients* **boolean** ingridients\_match(String db\_ingridients,String usr\_ingridients) {  
 **boolean** containing = **true**;  
 Set db\_set = convert\_str\_to\_set (db\_ingridients);  
 Set usr\_set = convert\_str\_to\_set (usr\_ingridients);  
  
 **if** (**SearchWith**) { *// find all db recipes that has the specified items in them* **textInfo**.setText(**"The following recipes contain marked items"**);  
 Iterator iter = usr\_set.iterator();  
 **while** (containing && iter.hasNext()) {  
 containing = containing && db\_set.contains(iter.next()); *// => rational I'd like some chocolate* }  
 Log.*d*(***TAG***, **"db set includes items? "**+containing);  
 }  
 **else** {  
 **textInfo**.setText(**"The following recipes could be made with marked items"**);  
 Iterator iter = db\_set.iterator();  
 **while** (containing && iter.hasNext()) {  
 containing = containing && usr\_set.contains(iter.next()); *// => rational I only have these ingridients at home* }  
 Log.*d*(***TAG***, **"usr set includes items? "**+containing);  
 }  
 **return** (containing); *// returns if user ingridinets are in db ingridients* }  
  
 *// handle search image. implemented but not intended for use* @Override  
 **public boolean** onCreateOptionsMenu(Menu menu) {  
 MenuInflater inflater = getMenuInflater();  
 inflater.inflate(R.menu.***options\_menu***, menu);  
 *// Associate searchable configuration with the SearchView  
 //SearchManager searchManager = (SearchManager) getSystemService(this.SEARCH\_SERVICE);  
 //SearchView searchView = (SearchView) menu.findItem(R.id.search).getActionView();  
 //searchView.setSearchableInfo(searchManager.getSearchableInfo(getComponentName()));  
 //searchView.setIconifiedByDefault(false);* **return true**;  
 }  
 @Override  
 **public boolean** onOptionsItemSelected(MenuItem item) {  
 *// clear all selections when changing options* Fragment frag = **fm**.findFragmentById(R.id.***fragment\_place***);  
 GridView gv = frag.getView().findViewById(R.id.***gridview***);  
 GetIngridients.*clear\_selection*(gv);  
 **textIngridients**.setText(**""**);  
  
 *//Handle item selection* **switch** (item.getItemId()) {  
 *//case R.id.search:  
 // Toast.makeText(getApplicationContext(), "Search button clicked", Toast.LENGTH\_SHORT).show();  
 // return true;* **case** R.id.***recipeWith***:  
 Toast.*makeText*(getApplicationContext(), **"Setting search - recipe with ingridients"**, Toast.***LENGTH\_SHORT***).show();  
 **SearchWith** = **true**;  
 **return true**;  
 **case** R.id.***recipeFrom***:  
 Toast.*makeText*(getApplicationContext(), **"Setting search - recipe from ingridients"**, Toast.***LENGTH\_SHORT***).show();  
 **SearchWith** = **false**;  
 **return true**;  
 **default**:  
 **return super**.onOptionsItemSelected(item);  
 }  
 }  
 *// set missing ingridients in fragment from main activity according to difference in sets: db .vs. selection* **public void** changeFragmentGridView(String usr\_ingridients, String db\_ingridients) {  
 Fragment frag = **fm**.findFragmentById(R.id.***fragment\_place***);  
 GridView gv = frag.getView().findViewById(R.id.***gridview***);  
 Set db\_set = convert\_str\_to\_set (db\_ingridients);  
 Set usr\_set = convert\_str\_to\_set (usr\_ingridients);  
 GetIngridients.*mark\_missing*(gv,usr\_set,db\_set,**SearchWith**);  
 }  
  
 *// public function called from fragment to update textIngridients with selected fragemnt ingridients* **public void** update\_ingridients(String cur\_ingr) {  
 **textIngridients**.setText(cur\_ingr);  
 }  
  
 **private static** File getGalleryPath() {  
 **return** Environment.*getExternalStoragePublicDirectory*(Environment.*DIRECTORY\_DCIM*);  
 }  
 */\*  
 private void OpenGallery(){  
 Intent getImageIntent = new Intent(Intent.ACTION\_GET\_CONTENT);  
 getImageIntent .setType("image/\*");  
 startActivityForResult(getImageIntent , IMAGE\_PICKER );  
 }  
 @Override  
 public void onActivityResult(int requestCode, int resultCode, Intent data) {  
 if (requestCode== IMAGE\_PICKER && resultCode == RESULT\_OK) {  
 Uri fullPhotoUri = data.getData();  
 imageView.setImageURI(fullPhotoUri);  
 }  
 }  
 \*/*}

GetIngridients.java

**package** com.example.recipemng;  
  
**import** android.app.Fragment;  
**import** android.graphics.Color;  
**import** android.os.Bundle;  
**import** android.util.Log;  
**import** android.view.LayoutInflater;  
**import** android.view.View;  
**import** android.view.ViewGroup;  
**import** android.widget.AdapterView;  
**import** android.widget.ArrayAdapter;  
**import** android.widget.GridView;  
  
**import** java.util.Set;  
  
**import static** android.content.ContentValues.***TAG***;  
  
**public class** GetIngridients **extends** Fragment {  
  
 **public static** String[] *Titles\_bake* =  
 {**"Flour"**, **"Eggs"**, **"Milk"**, **"Honey"**, **"Sugar"**, **"Oil"**, **"Chocolate"**, **"Cacao"**, **"Butter"**, **"Cream"**, **"Baking-Powder"**, **"Peanuts"**};  
 **public static** String[] *Titles\_make* =  
 {**"Vegetables"**, **"Herbs"**, **"Tomato-sauce"**, **"onions"**, **"Eggs"**, **"Oil"**, **"Pasta"**, **"Flour"**, **"Garlic"**, **"Bread-crumbs"**, **"Rice"**,**"Fruits"**};  
 **public static** String[] *Titles* = **new** String[12];  
 **private** Boolean[] **pressed**;  
 String **father** = **"1"**; *// container activity - mainActivity (1) or inputActivity (2)* String **recipe** = **"bake"**; *// container titles - bake or make* GridView **gv**;  
 String **cur\_ingr** = **""**; *// this string will be passed from the fragment to the mainactivity by public function* ArrayAdapter **adapterCB**;  
 View **view**;  
  
 @Override  
 **public void** onCreate(Bundle savedInstanceState) {  
 **super**.onCreate(savedInstanceState);  
  
 **if** (getArguments() != **null**) {  
 **father** = getArguments().getString(**"activity"**);  
 **recipe** = getArguments().getString(**"recipe"**);  
 }  
 **if** (**recipe**.equals(**"cake"**)) {  
 Log.*e*(***TAG***, **"running get ingredients fragment with cake recipe $"** + **recipe** + **"$ activity "**+ **father**);  
 **for** (**int** i = 0; i < *Titles\_bake*.**length**; i++)  
 *Titles*[i] = *Titles\_bake*[i];  
 }  
 **else** {  
 Log.*e*(***TAG***, **"running get ingredients fragment with make recipe $"** + **recipe** + **"$ activity "**+ **father**);  
 **for** (**int** i = 0; i < *Titles\_make*.**length**; i++)  
 *Titles*[i] = *Titles\_make*[i];  
 }  
 **pressed** = **new** Boolean[*Titles*.**length**];  
 Log.*e*(***TAG***, **"setting titles and pressed"**);  
  
 }  
  
 *// clears all selections, called from inputActivity when changing selection option, and internally* **public static void** clear\_gray\_marks(GridView gridV) {  
 **for** (**int** i = 0; i < *Titles*.**length**; i++) {  
 gridV.getChildAt(i).setBackgroundColor(Color.***TRANSPARENT***); *// clean all markers* }  
 }  
  
 *// clears all selections, called from inputActivity when changing selection option, and internally* **public static void** clear\_selection(GridView gridV) {  
 **for** (**int** i = 0; i < *Titles*.**length**; i++) {  
 gridV.setItemChecked(i, **false**);  
 gridV.getChildAt(i).setTag(**"notselected"**); *// DOES NOT WORK!* }  
 }  
 *// if searchwith, it would highlight (gray) all the additional ingridients  
 // otherwise, it would highlight (gray) the required ingridients out of those marked.* **public static void** mark\_missing(GridView gridV, Set usr\_set, Set db\_set, **boolean** SearchWith) {  
 String ing;  
 *clear\_gray\_marks*(gridV);  
  
 **if** (SearchWith) { *// highlight the additional ingridients that are in db\_set but not in usr\_set* db\_set.removeAll(usr\_set);  
 }  
 **else** { *// highlight the ingridients that are in both db\_set and usr\_set - only they are required* db\_set.retainAll(usr\_set);  
 }  
  
 *// traverse list and for each ingridient that is in working set set light gray background coloe* **for** (**int** i = 0; i < *Titles*.**length**; i++) {  
 ing = gridV.getItemAtPosition(i).toString();  
 **if** (db\_set.contains(ing))  
 gridV.getChildAt(i).setBackgroundColor(Color.***LTGRAY***);  
 }  
 }  
  
 @Override  
 **public** View onCreateView(LayoutInflater inflater,  
 ViewGroup container, Bundle savedInstanceState) {  
 **view** = inflater.inflate(R.layout.***get\_ingridients***, container, **false**);  
 **gv** = (GridView) **view**.findViewById(R.id.***gridview***);  
 **gv**.setChoiceMode(GridView.***CHOICE\_MODE\_MULTIPLE***);  
  
 **adapterCB** = **new** ArrayAdapter(getActivity(), android.R.layout.***simple\_list\_item\_multiple\_choice***, *Titles*);  
 **gv**.setAdapter(**adapterCB**);  
 **for** (**int** i = 0; i < *Titles*.**length**; i++)  
 **pressed**[i] = **false**;  
 *//for (int i = 0; i < adapterCB.getCount(); i++)  
 // gv.setItemChecked(i, false);* **gv**.setOnItemClickListener(**new** AdapterView.OnItemClickListener() {  
 **public void** onItemClick(AdapterView<?> parent, View v, **int** position, **long** id) {  
 **for** (**int** i = 0; i < *Titles*.**length**; i++)  
 **gv**.getChildAt(i).setBackgroundColor(Color.***TRANSPARENT***); *// clean all markers* String selectedItem = parent.getItemAtPosition(position).toString();  
 **if** (!**pressed**[position]) { *// add grid element* **cur\_ingr** = **cur\_ingr** + selectedItem + **" "**;  
 v.setTag(**"selected"**);  
 **gv**.setItemChecked(position, **true**);  
 *// gv.getChildAt(position).setBackgroundColor(Color.LTGRAY);* **pressed**[position] = **true**;  
 } **else** { *// remove grid element* **cur\_ingr** = **cur\_ingr**.replace(selectedItem+**" "**,**""**);  
 **gv**.setItemChecked(position, **false**);  
 v.setTag(**"notselected"**);  
 *// gv.getChildAt(position).setBackgroundColor(Color.TRANSPARENT);* **pressed**[position] = **false**;  
 }  
 **if** (**father** == **"1"**)  
 ((MainActivity)getActivity()).update\_ingridients(**cur\_ingr**);  
 **else** ((InputActivity)getActivity()).update\_ingridients(**cur\_ingr**);  
 }  
 });  
 **return view**;  
 }  
}

RecipeItem.java

**package** com.example.recipemng;  
**import** com.google.firebase.database.IgnoreExtraProperties;  
  
@IgnoreExtraProperties  
**public class** RecipeItem {  
 **public** String **name**;  
 **public** String **ingridients**;  
 **public** String **imageUrl**;  
 **public** String **instructions**;  
  
 **public** RecipeItem() {  
 }  
  
 **public** RecipeItem(String name, String ingridients, String imageUrl, String instructions) {  
 **this**.**name** = name;  
 **this**.**ingridients** = ingridients;  
 **this**.**imageUrl** = imageUrl;  
 **this**.**instructions** = instructions;  
 }  
}

activity\_first.xml

*<?***xml version="1.0" encoding="utf-8"***?>*<**FrameLayout xmlns:android="http://schemas.android.com/apk/res/android"  
 xmlns:tools="http://schemas.android.com/tools"  
 android:layout\_width="fill\_parent"  
 android:layout\_height="fill\_parent"  
 tools:context=".FirstActivity"**>  
  
 <**ImageView  
 android:id="@+id/sneakycar"  
 android:layout\_width="fill\_parent"  
 android:layout\_height="fill\_parent"  
 android:layout\_centerHorizontal="true"  
 android:alpha="0.4"  
 android:background="@drawable/goodfood"** />  
  
 <**TextView  
 android:id="@+id/header"  
 android:layout\_width="350dp"  
 android:layout\_height="wrap\_content"  
 android:layout\_centerHorizontal="true"  
 android:layout\_marginTop="80dp"  
 android:contentDescription="@string/app\_name"** />  
  
 <**LinearLayout  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:id="@+id/buttons"  
 android:orientation="horizontal"** >  
  
 <**Button  
 android:id="@+id/btn\_cake"  
 android:layout\_width="200dp"  
 android:layout\_height="225dp"  
 android:layout\_alignParentTop="true"  
 android:layout\_marginTop="400dp"  
 android:layout\_marginLeft="0dp"  
 android:strokeColor="#010101"  
 android:strokeWidth="2"  
 android:alpha="1.0"  
 android:background="@drawable/cake"**/>  
  
 <**Button  
 android:id="@+id/btn\_make"  
 android:layout\_width="200dp"  
 android:layout\_height="225dp"  
 android:layout\_alignParentTop="true"  
 android:layout\_marginTop="420dp"  
 android:layout\_marginLeft="0dp"  
 android:strokeColor="#010101"  
 android:strokeWidth="2"  
 android:alpha="1.0"  
 android:background="@drawable/make"**/>  
 </**LinearLayout**>  
</**FrameLayout**>

Activity\_main.xml

*<?***xml version="1.0" encoding="utf-8"***?>*<**LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"  
 xmlns:tools="http://schemas.android.com/tools"  
 android:id="@+id/activity\_main"  
 android:layout\_width="fill\_parent"  
 android:layout\_height="match\_parent"  
 android:orientation="vertical"  
 android:paddingBottom="@dimen/activity\_vertical\_margin"  
 android:paddingLeft="@dimen/activity\_horizontal\_margin"  
 android:paddingRight="@dimen/activity\_horizontal\_margin"  
 android:paddingTop="@dimen/activity\_vertical\_margin"  
 tools:context=".MainActivity"**>  
  
 <**TextView  
 android:id="@+id/info"  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:textSize="20sp"  
 android:text="Welcome. please enter your recipes"**/>  
  
  
 <**EditText  
 android:id="@+id/name"  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:hint="Enter name"** />  
  
 <**FrameLayout android:id="@+id/fragment\_place"  
 android:name="com.example.recipemng.GetIngridients"  
 android:layout\_width="fill\_parent"  
 android:layout\_height="200dp"** />  
  
 <**TextView  
 android:id="@+id/ingridients"  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:textSize="20sp"** />  
  
 <**View style="@style/Divider"**/>  
  
 <**EditText  
 android:id="@+id/instructions"  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:maxLines="1"  
 android:hint="Enter instructions"** />  
  
 <**EditText  
 android:id="@+id/imageUrl"  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:hint="Enter url"**/>  
  
 <**LinearLayout  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:orientation="horizontal"** >  
  
 <**Button  
 android:id="@+id/btn\_load"  
 android:text="Load Image"  
 android:layout\_marginTop="16dp"  
 android:background="@color/colorPrimary"  
 android:layout\_margin="2dp"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:textColor="@android:color/white"** />  
  
 <**Button  
 android:id="@+id/btn\_Showmore"  
 android:text="Show More"  
 android:layout\_marginTop="16dp"  
 android:background="@color/colorPrimary"  
 android:layout\_margin="2dp"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:textColor="@android:color/white"** />  
  
 <**Button  
 android:id="@+id/btn\_save"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:layout\_marginTop="16dp"  
 android:background="@color/colorPrimary"  
 android:layout\_margin="2dp"  
 android:text="save RCP"  
 android:textColor="@android:color/white"  
 android:textStyle="bold"** />  
  
 <**Button  
 android:id="@+id/btn\_next"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:layout\_marginTop="16dp"  
 android:background="@color/colorPrimary"  
 android:layout\_margin="2dp"  
 android:text="Find->"  
 android:textColor="@android:color/white"** />  
 </**LinearLayout**>  
  
 <**ImageView  
 android:id="@+id/imgView"  
 android:layout\_width="match\_parent"  
 android:layout\_height="match\_parent"** />  
</**LinearLayout**>

Activity\_input.xml

*<?***xml version="1.0" encoding="utf-8"***?>*<**LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"  
 xmlns:tools="http://schemas.android.com/tools"  
 android:id="@+id/activity\_input"  
 android:layout\_width="fill\_parent"  
 android:layout\_height="match\_parent"  
 android:orientation="vertical"  
 android:paddingBottom="@dimen/activity\_vertical\_margin"  
 android:paddingLeft="@dimen/activity\_horizontal\_margin"  
 android:paddingRight="@dimen/activity\_horizontal\_margin"  
 android:paddingTop="@dimen/activity\_vertical\_margin"  
 tools:context=".InputActivity"**>  
  
 <**TextView  
 android:id="@+id/info"  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:textSize="20sp"  
 android:text="Welcome. enter ingridients to find recipe"**/>  
  
 <**fragment  
 android:id="@+id/fragment\_place"  
 android:name="com.example.recipemng.GetIngridients"  
 android:layout\_width="fill\_parent"  
 android:layout\_height="200dp"** />  
  
 <**TextView  
 android:id="@+id/ingridients"  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:textSize="20sp"  
 android:text="Empty list"**/>  
  
 <**View style="@style/Divider"**/>  
  
 <**LinearLayout  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:id="@+id/buttons"  
 android:orientation="horizontal"** >  
  
 <**Button  
 android:id="@+id/btn\_load"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:layout\_marginTop="16dp"  
 android:background="@color/colorPrimary"  
 android:text="load all"  
 android:layout\_margin="10dp"  
 android:textColor="@android:color/white"  
 android:textStyle="bold"** />  
  
 <**Button  
 android:id="@+id/btn\_find"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:layout\_marginTop="16dp"  
 android:background="@color/colorPrimary"  
 android:layout\_margin="10dp"  
 android:text="Find Recipe"  
 android:textColor="@android:color/white"  
 android:textStyle="bold"** />  
 </**LinearLayout**>  
  
 <**ListView  
 android:id="@+id/list\_view"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="200dp"  
 android:layout\_below="@id/buttons"** />  
  
 <**ImageView  
 android:id="@+id/imgView"  
 android:layout\_width="match\_parent"  
 android:layout\_height="match\_parent"** />  
  
</**LinearLayout**>

Get\_ingridients.xml

*<?***xml version="1.0" encoding="utf-8"***?>*<**RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"  
 xmlns:tools="http://schemas.android.com/tools"  
 android:id="@+id/rootContainer"  
 android:layout\_width="match\_parent"  
 android:layout\_height="match\_parent"  
 android:orientation="vertical"  
 tools:context=".InputActivity"**>  
  
 <**GridView  
 android:id="@+id/gridview"  
 android:layout\_width="fill\_parent"  
 android:layout\_height="wrap\_content"  
 android:layout\_margin="5dp"  
 android:layout\_marginTop="20dp"  
 android:padding="0dp"  
 android:stretchMode="columnWidth"  
 android:numColumns="3"  
 android:verticalSpacing="2dp"  
 android:horizontalSpacing="0dp"  
 android:gravity="center"**>  
 </**GridView**>  
  
</**RelativeLayout**>

Options\_menu.xml

*<?***xml version="1.0" encoding="utf-8"***?>*<**menu xmlns:android="http://schemas.android.com/apk/res/android"  
 xmlns:app="http://schemas.android.com/apk/res-auto"**>  
 *<!--item android:id="@+id/search"  
 android:title="search"  
 android:icon="@drawable/ic\_search"  
 app:showAsAction="always|collapseActionView"  
 app:actionViewClass="android.widget.SearchView" /-->* <**item android:id="@+id/recipeWith"  
 android:title="recipe with"**/>  
 *<!--android:icon="@drawable/ic\_search" /-->* <**item android:id="@+id/recipeFrom"  
 android:title="recipe from"**/>  
 *<!--android:icon="@drawable/ic\_search" /-->*</**menu**>

Colors.xml

*<?***xml version="1.0" encoding="utf-8"***?>*<**resources**>  
 <**color name="colorPrimary"**>#008577</**color**>  
 <**color name="colorPrimaryDark"**>#00574B</**color**>  
 <**color name="colorAccent"**>#D81B60</**color**>  
</**resources**>

Strings.xml

<**resources**>  
 <**string name="app\_name"**>Recipe Manager</**string**>  
 <**string name="search\_title"**>Search Image</**string**>  
 <**string name="app\_label"**>RecipeMng</**string**>  
 <**string name="search\_hint"**>search image</**string**>  
</**resources**>

Styles.xml

<**resources**>  
  
 *<!-- Base application theme. -->* <**style name="AppTheme" parent="Theme.AppCompat.Light.DarkActionBar"**>  
 *<!-- Customize your theme here. -->* <**item name="colorPrimary"**>@color/colorPrimary</**item**>  
 <**item name="colorPrimaryDark"**>@color/colorPrimaryDark</**item**>  
 <**item name="colorAccent"**>@color/colorAccent</**item**>  
 </**style**>  
  
 <**style name="Divider"**>  
 <**item name="android:layout\_width"**>match\_parent</**item**>  
 <**item name="android:layout\_height"**>1dp</**item**>  
 <**item name="android:background"**>?android:attr/listDivider</**item**>  
 </**style**>  
</**resources**>