



You are required to build an application called “Foodagram” that helps the user store food recipes and browse them by name or by category. In addition, you need to create and manage a simple SQLite database with only one table called Dishes that has the following schema:

ID: int	Name: text	Category: text	Recipe: text
---------	------------	----------------	--------------

The application is composed of two main activities: the first one is for browsing dishes (figures a, b and c) while the second one is for adding a dish (figure d).

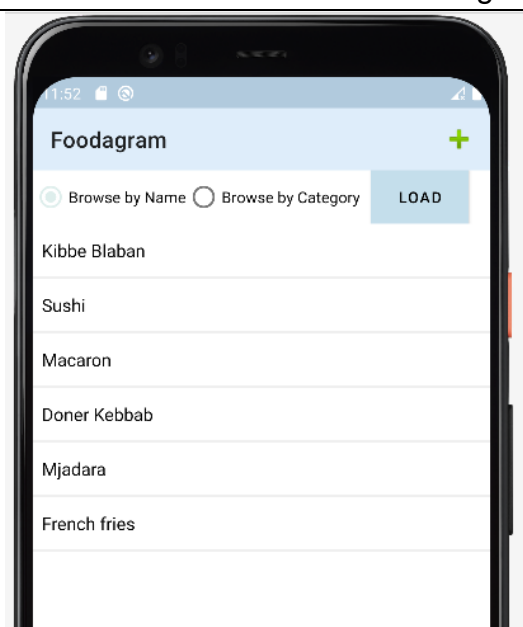


Figure a. Home Activity with browsing by name

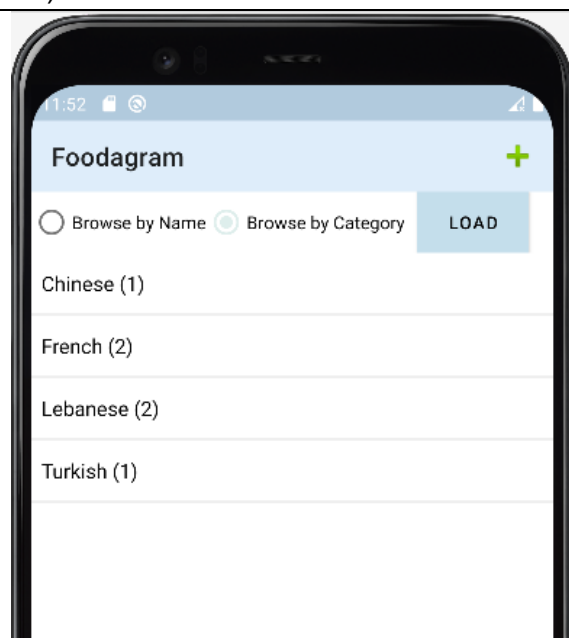


Figure b. Home Activity with browsing by category

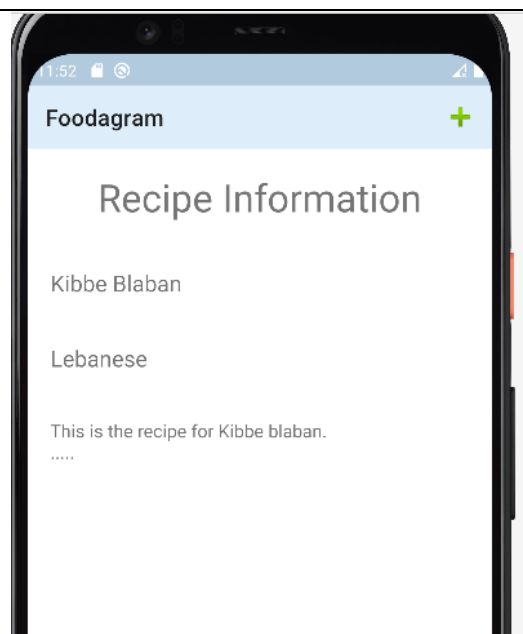


Figure c. View Fragment

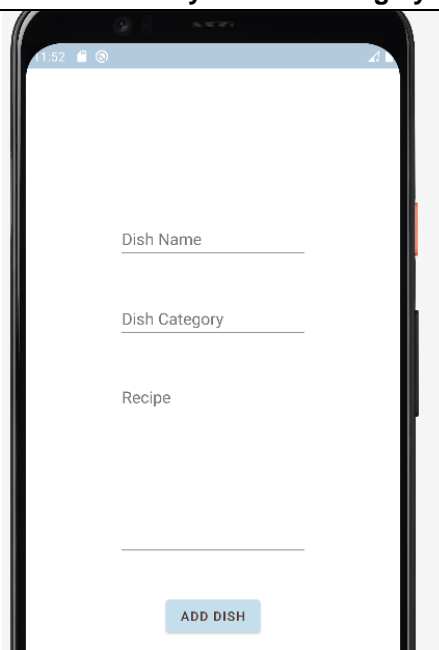


Figure d. Add Dish Activity

1. The Home Activity consists of 2 parts: An action bar containing 1 icon (+ with id='add'), and a container to add inside it a fragment (id='frmContainer'). Two fragments 'ListFragment' and 'ViewFragment' are defined to handle the browsing task.
  - a. ListFragment is a piece of UI containing the following widgets: (1) two radio buttons ('browse by name' and 'browse by category') with ids 'rb\_name' and 'rb\_category'; (2) a button with id 'btnLoad' to load the data in the listview; (3) a listview with id 'lstData' to browse the list of dishes or the list of categories depending on the radio button chosen. If a category is selected in the listview, all the dishes having this category are shown in the listview while staying in the same fragment.
  - b. If a dish is selected in the listview, the ViewFragment should replace the ListFragment in the main activity in order to show the information of the selected dish as shown in the figure c.
2. The AddDish Activity: The user reaches the Add Dish Activity after pressing the (+) icon on the action bar of the Home Activity. This activity is used to add a new dish to the database (figure d). It contains three EditText with the ids 'edtName', 'edtCategory', 'edtRecipe' and the button 'ADD DISH' with the id 'btnAdd'.

Question I - A database handler class has been defined for you in which you should complete the pieces of code (Question I.a, Question I.b and Question I.c) in the following figure while the others are given.

```
public class DataBaseHandler extends SQLiteOpenHelper {

    private static final int DATABASE_VERSION = 2;
    private static final String DATABASE_NAME = "DishesList";
    private static final String TABLE_DISHES= "Dishes";
    private SQLiteDatabase db;
    // Contacts Table Columns names
    private static final String KEY_ID = "ID";
    private static final String KEY_NAME = "Name";
    private static final String KEY_CAT= "Category";
    private static final String KEY_RECIPE= "Recipe";

    public DataBaseHandler(Context c) {
        super(c, DATABASE_NAME, null,DATABASE_VERSION);
    }
    @Override
    public void onCreate(SQLiteDatabase db) {
        String CREATE_DISHES_TABLE="CREATE TABLE " + TABLE_DISHES + "(" +KEY_ID +" INTEGER PRIMARY KEY,"
            + KEY_NAME + " Text," + KEY_CAT + " Text," + KEY_RECIPE +" Text)";
        db.execSQL(CREATE_DISHES_TABLE);
    }

    @Override
    public void onUpgrade(SQLiteDatabase db, int oldVersion, int newVersion) {
        db.execSQL("DROP TABLE IF EXISTS " + TABLE_DISHES );
        onCreate(db);
    }

    ArrayList<String> getAllCategories(){ /*this method is given. */
        /*It return the list of categories in the table dishes.*/ }

    ArrayList<Dish> getAllDishes(){ /*complete code here (Question I.a)*/ }

    ArrayList<Dish> getDishesCategory( String categ){ /*complete code here (Question I.b)*/ }

    public long AddDish(Dish d){ /*complete code here (Question I.c)*/ }

}
```

Question II – With the help of the above class, complete the code (Question II) in the next figure to add a dish to the database and to return back to the home activity

```
public class NewDish extends AppCompatActivity {
    EditText name,category, recipe;
    Button add;
    DataBaseHandler db;
    long l;
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_new_dish);
    }

    public void add_dish(View v){
        //This method is called when the button 'ADD DISH' is clicked
        //complete code here - Question II
    }
}
```

Question III – Complete the code of the home activity (Question III.a, Question III.b, and Question III.c) aiming to add the ListFragment to the frmContainer and to implement the navigation to the AddDish activity (and the return back) when the add button of the toolbar is clicked.

```
public class MainActivity extends AppCompatActivity {
    Toolbar myToolbar;
    DataBaseHandler mydb;
    ArrayAdapter<Dish> ar_dishes;
    ArrayList<Dish> dataset_dishes;
    ListView lst;
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
        myToolbar = findViewById(R.id.toolbar);
        setSupportActionBar(myToolbar);
        //Complete code here Question III.a
    }

    public boolean onCreateOptionsMenu(Menu menu) {
        MenuInflater inflater = getMenuInflater();
        inflater.inflate(R.menu.mytoolbar, menu);
        return super.onCreateOptionsMenu(menu);
    }

    public boolean onOptionsItemSelected(MenuItem item) {
        switch (item.getItemId()) {
            case R.id.add:
                //Complete code here Question III.b
                return true;
            default:
                return super.onOptionsItemSelected(item);
        }
    }

    public void onActivityResult(int reqcode, int rescode, Intent intent) {
        super.onActivityResult(reqcode, rescode, intent);
        if(reqcode == 123){
            //Complete code here Question III.c
        }
    }
}
```

Question IV – Complete the code of the ListFragment fragment (Question IV.a, Question IV.b, and Question IV.c) aiming to handle the browsing by name and by category and the replacement of ListFragment by ViewFragment when a dish is selected.

```
public class ListFragment extends Fragment {
    RadioButton rb_name, rb_category;
    Button btnLoad;
    DataBaseHandler db;
    ListView lst;
    ArrayAdapter<Dish> ar_dishes;
    ArrayAdapter<String> ar_categories;
    ArrayList<Dish> dataset_dishes;
    ArrayList<String> dataset_categories;

    @Override
    public void onActivityCreated(@Nullable Bundle savedInstanceState) {
        super.onActivityCreated(savedInstanceState);
        rb_name = getActivity().findViewById(R.id.rbByName);
        rb_category = getActivity().findViewById(R.id.rbByCategory);
        btnLoad = getActivity().findViewById(R.id.btnLoad);
        db = new DataBaseHandler(getContext());
        lst = getActivity().findViewById(R.id.lstData);

        btnLoad.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                //Complete code here Question IV.a
                //here you should check which radio button is checked and then load the appropriate data in the listview
            }
        });
        lst.setOnItemClickListener(new AdapterView.OnItemClickListener() {
            @Override
            public void onItemClick(AdapterView<?> parent, View view, int position, long id) {
                if(rb_category.isChecked()) {
                    //Complete code here Question IV.b
                    //here you should load all the dishes that has the selected category in the listview
                }
                else{
                    //Complete code here Question IV.c
                    //here you should replace the current fragment with the ViewFragment
                }
            }
        });
    }
}
```

The classes ViewFragment (below) and the class Dish (modeling a dish) are given.

```
public class ViewFragment extends Fragment {
    private String name, category, recipe;
    private TextView txtname, txtcategory, txtrecipe;
    public ViewFragment(String n, String c, String r) {
        super();
        this.name = n;
        this.category = c;
        this.recipe = r;
    }

    @Override
    public void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
    }

    @Override
    public View onCreateView(LayoutInflater inflater, ViewGroup container,
                             Bundle savedInstanceState) {
        // Inflate the layout for this fragment
        return inflater.inflate(R.layout.fragment_view, container, false);
    }

    @Override
    public void onActivityCreated(@Nullable Bundle savedInstanceState) {
        super.onActivityCreated(savedInstanceState);
        txtname = getActivity().findViewById(R.id.txtname);
        txtcategory = getActivity().findViewById(R.id.txtcategory);
        txtrecipe = getActivity().findViewById(R.id.txtrecipe);
        txtname.setText(this.name);
        txtcategory.setText(this.category);
        txtrecipe.setText(this.recipe);
    }
}
```