5 アクティビティ(「一覧」)を作成する



<u>ファイル名:</u> src/jp/edu/mie/View010. java (新規作成)

```
/*
 * View010
 */
package jp.edu.mie;
import android.app.Activity;
import android.content.Intent;
import android.os.Bundle;
import android view. View;
import android.view.View.OnClickListener;
import android.widget.Button;
public class View010 extends Activity
{
    Button listButton;
    protected void onCreate(Bundle savedInstanceState)
    {
        super. onCreate(savedInstanceState);
        setContentView(R. layout. activity main);
        listButton = (Button) findViewById(R.id.button_list);
        listButton.setOnClickListener(new MainOnClickListener());
    }
    class MainOnClickListener implements OnClickListener
    ł
        public void onClick(View view)
         {
            //一覧
            if (view == listButton)
            {
                 try{
                    Intent dbIntent = new Intent(getApplicationContext(),
                                                      ShowDataBase.class); //①
                    startActivity(dbIntent);
                }catch (Exception e) {
                     e.printStackTrace();
                }
            }
```

1 1 8:

```
※MainActivity. java を実行する前に次の作業を行ってください。

    ・余分な命令が作成されているので削除してください。

       ・res/menu/ MainActivity.xml を削除してください。
①Intent dbIntent = new Intent(getApplicationContext(), ShowDataBase.class);
     Intent のインスタンスを生成します。元画面と次画面を明示的に指定しています。
   Activity クラスの startActivity メソッドは、Intent での情報の受け渡しを開始(返り
   値無し)します。
ファイル名: src/jp/edu/mie/ShowDataBase.java
                                             (新規作成)
 package jp.edu.mie;
                                                        jp.edu.mie.ShowDataBase
 import android.app.Activity;
 import android.database.Cursor;
 import android.database.sqlite.SQLiteDatabase;
                                                     1001 山田太郎 三重県 24.5
 import android.os.Bundle;
                                                        佐藤華子 神奈川県 24.5
 import android.util.Log;
                                                        鈴木健二 茨城県 26.0
 import android.view.View;
                                                     1004 渡辺 進 東京都 0.0
 import android.view.View.OnClickListener;
 import android.widget.ArrayAdapter;
 import android.widget.Button;
 import android.widget.ListView;
 public class ShowDataBase extends Activity
 {
     SQLiteDatabase db;
     protected void onCreate (Bundle savedInstanceState)
     ł
         super. onCreate(savedInstanceState);
         setContentView(R. layout. show_database);
```

Button returnbutton = (Button) findViewById(R.id.button1);

ArrayAdapter<String> arrayadapter = new ArrayAdapter<String>

MyOpenHelper helper = new MyOpenHelper(this); SQLiteDatabase db = helper.getReadableDatabase();

String strSQL = "SELECT * FROM product";//②
Cursor cr = db.rawQuery(strSQL, null); //③

int i = cr.getColumnIndex("id");

int n = cr.getColumnIndex("name"); int a = cr.getColumnIndex("address"); int s = cr.getColumnIndex("shoeSize");

int id = cr.getInt(i); //idの内容を設定する

String name = cr.getString(n); //nameの内容を設定する String address = cr.getString(a); //addressの内容を設定する Double shoeSize = cr.getDouble(s); // shoeSizeの内容を設定する

//問い合わせを行う

while(cr.moveToNext()) //(5)

ListView listview = (ListView) findViewById(R.id.listView1); //①

```
      String row = id + " " + name + " " + address + " " + shoeSize;

      Android_SQLite 講座_画面 800×1280. docx
```

(this, android.R.layout.simple_list_item_1); //④

//列名を指定して列番号を取得する

Page 7

```
arrayadapter.add(row); //⑥ListViewの行を追加する
}
listview.setAdapter(arrayadapter); //⑦ListViewにアダプタを設定
cr.close();
db.close(); //データベースを閉じる
returnbutton.setOnClickListener(new ReturnClickListener());
}
class ReturnClickListener implements OnClickListener
{
    public void onClick(View v)
    {
        setResult(RESULT_OK);
        finish();
    }
}
```

①ListView listview = (ListView) findViewById(R.id.listView1);

結果を格納するリストを生成します。

②String strSQL = "SELECT * FROM product";

SQLの select 文の文字列を作成します。

③Cursor cr = db.rawQuery(strSQL, null);

DB オブジェクトに対して、②の select 文を実行します。

Android でデータの一覧を表示するには、ListViewを使うと便利です。ListViewを使 うと、配列やデータベース上のデータなど、複数のデータを一覧表示することができます。

SQLiteの検索結果であるCursorオブジェクトより、Adapterオブジェクトを作成して、 ListViewのAdapterに設定する事により、ListViewに簡単に検索結果を表示する事がで きます。

Adapter オブジェクト

Adapter は widget と表示するデータとの仲介役をつとめるオブジェクトです。

ArrayAdapter クラスはデータの一覧をリストなどのビューに渡すために使用されるク ラスです。ArrayAdapter クラスは BaseAdapter クラスのサブクラスです。

また、ArrayAdapter は、文字列を取り扱うためのアダプタです。引数には、アプリケーションのコンテキスト(通常は this)と、リスト内の1項目のレイアウト、そして表示する文字列の配列を指定します。

<String>の部分は、総称型のクラスを利用するときにデータ型を指定する記述です。総 称型のクラスでは、クラス内で利用するデータの型を変えることができます。

ArrayAdaputer では、String 型や CharSequence 型のデータが扱えるので、ここでは<>を使って型を特定しています。

「simple_list_item_1」は、Android にもともと用意されている定義済みのレイアウト ファイルの ID です。これは TextView をひとつ持つだけの非常に単純なレイアウトです。

Android(今回の場合は 4.2.2)を展開し、android - R.class - layout を確認すると、 simple_list_item_1 があることが確認できます。また、実際の xml を確認したい場合は、 C:¥android-sdk-windows¥platforms¥android-17¥data¥res¥layout フォルダ内の simple_list_item_1.xml を開くことで、確認をすることができます。 (5)while(cr.moveToNext())

select 文で検索した結果を1レコードずつ、カラムごとに row にセットします。

@arrayadapter.add(row);

rowを①のリストに追加します。

⑦listview.setAdapter(arrayadapter);

setAdapter メソッドを使って ArrayAdapter のオブジェクトを ListView にセットします。

<u>ファイル名:</u> src/jp/edu/mie/MyOpenHelper.java (新規作成)

```
package jp.edu.mie;
import android.content.Context;
import android.database.sqlite.SQLiteDatabase;
import android.database.sqlite.SQLiteOpenHelper;
public class MyOpenHelper extends SQLiteOpenHelper
   final static private int DB_VERSION = 1;
   public MyOpenHelper(Context context) //①
    ł
       super(context, "CustomerCard.db", null, DB_VERSION);
    }
   @Override
   public void onCreate(SQLiteDatabase db)
                                             //2
    ł
       String strSQL = "CREATE TABLE product" + "(id INTEGER PRIMARY KEY,
                              name STRING, address STRING, shoeSize double)";
       db.execSQL(strSQL);
       strSQL = "INSERT INTO product(id, name, address, shoeSize)" +
                                  "VALUES(1001, '山田太郎', '東京都', 26.5)";
       db. execSQL (strSQL); //product テーブルヘデータの追加
       strSQL = "INSERT INTO product(name, address, shoeSize)" +
                                  "VALUES('佐藤華子','神奈川県', 24.5)";
       db.execSQL(strSQL); //product テーブルヘデータの追加
       strSQL = "INSERT INTO product(name, address, shoeSize)" +
                                  "VALUES('鈴木健二','茨城県', 26.0)";
       db. execSQL(strSQL); //product テーブルヘデータの追加
       strSQL = "INSERT INTO product(name, address, shoeSize)" +
                                  "VALUES('渡辺 進','東京都', 0.0)";
       db. execSQL(strSQL); //product テーブルヘデータの追加
   }
   @Override
    //③
   public void onUpgrade (SQLiteDatabase db, int oldVersion, int newVersion) { }
```

①public MyOpenHelper(Context context)

データベースのファイル名とバージョン番号

データベースのファイル名とバージョン番号は、コンストラクタで指定しています。こ のクラス (MyOpenHelper)を使って実際にデータベースをオープンする際、万一、指定し た名前のファイル (CustomerCard.db)が存在しない場合は自動的に作成してくれます。

②public void onCreate(SQLiteDatabase db)

データベースファイルを作成した直後に、onCreate()メソッドが呼ばれ、そこに記述さ

れているテーブル作成の処理などが実行されます。なお、テーブルの作成には、 SQLiteDatabaseの execSQL()メソッドを使用しています。execSQL()は、SQL をそのまま 実行するメソッドです。

③public void onUpgrade(SQLiteDatabase db, int oldVersion, int newVersion) { }

onUpgrade()メソッドは、今回は特に何も実装していません。このメソッドはデータベースバージョンの更新時に実行される処理なので今回は割愛しています。

6 実習用の共通ソースを作成する

実習をする過程で変更するプログラムを作成します。

プログラムを作成する都度、指定の箇所を変更してください。

<u>ファイル名: MainActivity.java</u> (新規作成)

```
/*
     年
           組
                 席 名前
 * MainActivity
 */
package jp.edu.mie;
import android.app.Activity;
import android.content.Intent;
import android.os.Bundle;
import android.view.Window;
public class MainActivity extends Activity
{
    public void onCreate(Bundle bundle)
    {
       super.onCreate(bundle);
        requestWindowFeature(Window.FEATURE_NO_TITLE);//タイトルの非表示
       Intent intent = new Intent(getApplicationContext(),
                                                        (1) . class);
       startActivity(intent);
   }
```

①該当の Activity 名を入力してください。

「一覧」の場合は、View010 と入力してください。

<u>ファイル名: AndroidManifest.xlm</u>

```
</activity>
<activity android:name=".MyOpenHelper"
android:label="@string/app_name">
</activity>
<activity android:name=".ShowDataBase" >
</activity>
<activity android:name=".__________">
</activity>
```

①該当の Activity 名を入力してください。

「一覧」の場合は、View010 と入力してください。