

Курс:

Программирование под Android с нуля

Тренер:

Пономарев Алексей

oleksii_android@web-academy.com.ua

Лекция 13

Loaders, ContentProviders



Loader

AsyncTaskLoader – объект, который выполняет работу в фоновом потоке. В отличие от **AsyncTask**, может возвращать результат предыдущей работы. В основном используются для загрузки данных.

Loader create

```
public class StudentsLoader
    extends AsyncTaskLoader<ArrayList<Student>> {

    private DataBaseHelper mDBHelper;

    public StudentsLoader(Context context) {
        super(context);

        this.mDBHelper = new DataBaseHelper(context);
    }

    @Override
    public ArrayList<Student> loadInBackground() {
        return mDBHelper.getStudents();
    }

    @Override
    protected void onStartLoading() {
        forceLoad();
    }
}
```

Init/Restart

```
public class MainActivity extends AppCompatActivity
    implements LoaderManager.LoaderCallbacks<List<Student>> {
    ...
    public void OnClick(View v) {
        switch (v.getId()) {
            case R.id.button:
                getLoaderManager().initLoader(0, null, this);           //Вызов
                break;
            case R.id.button2:
                getLoaderManager().restartLoader(0, null, this);        //Перезапуск
                break;
        }
    }

    @Override
    public Loader<List<Student>> onCreateLoader(int i, Bundle bundle) {
        return new StudentsLoader(this);    //Создание
    }

    @Override
    public void onLoadFinished(Loader<List<Student>> loader, List<Student> students)
        //Получение ответа
    }

    @Override
    public void onLoaderReset(Loader<List<Student>> loader) {
    }
}
```

Content provider

Content provider – обертка над базой данных, которая позволяет дать доступ к базе данных другим приложениям. В Android есть стандартные content provider для доступа к контактам, смс, журналу звонков и т.д.

Contacts provider

```
private List<String> getContacts() {
    List<String> names = new ArrayList<>();

    Cursor cursor = getContentResolver().query(
        ContactsContract.CommonDataKinds.Phone.CONTENT_URI, //Uri
        new String[]{ContactsContract.CommonDataKinds.Phone.DISPLAY_NAME}, //Колонки
        null, null, null);

    if (cursor.moveToFirst()) {
        while (!cursor.isAfterLast()) {
            names.add(cursor.getString(
                cursor.getColumnIndex(
                    ContactsContract.CommonDataKinds.Phone.DISPLAY_NAME)));

            cursor.moveToNext();
        }
    }

    cursor.close();

    return names;
}
```

```
//Manifest
<uses-permission android:name="android.permission.READ_CONTACTS" />
```

Custom content provider

//Uri

content://<authority>/<path>/<id>

content://ua.com.webacademy.lection11/students

content://ua.com.webacademy.lection11/students/1

```
<application
```

```
...
```

```
<provider android:name=".StudentsProvider"
```

```
    android:authorities="ua.com.webacademy.lection11">
```

```
</provider>
```

```
</application>
```

Custom content provider

```
public class StudentsProvider extends ContentProvider {

    @Override
    public boolean onCreate() {
        //Создание content provider, подключение к БД
    }

    @Nullable
    @Override
    public Cursor query(Uri uri, String[] strings, String s, String[] strings1, String s1) {
        //Обработка запроса на чтение
    }

    @Nullable
    @Override
    public Uri insert(Uri uri, ContentValues contentValues) {
        //Обработка запроса на вставку
    }

    @Override
    public int delete(Uri uri, String s, String[] strings) {
        //Обработка запроса на удаление
    }

    @Override
    public int update(Uri uri, ContentValues contentValues, String s, String[] strings) {
        //Обработка запроса на обновление
    }

    @Nullable
    @Override
    public String getType(Uri uri) {
        //MIME типы данных
    }
}
```


Custom content provider

```
private List<Student> getStudents() {
    List<Student> students = new ArrayList<>();

    Uri uri = Uri.parse("content://ua.com.webacademy.lection11/students");

    Cursor cursor = getContentResolver().query(uri, null, null, null, null);

    if (cursor.moveToFirst()) {
        while (!cursor.isAfterLast()) {
            Student student = new Student();

            student.id = cursor.getLong(cursor.getColumnIndex("_id"));
            student.FirstName = cursor.getString(cursor.getColumnIndex("FirstName"));
            student.LastName = cursor.getString(cursor.getColumnIndex("LastName"));
            student.Age = cursor.getLong(cursor.getColumnIndex("Age"));

            students.add(student);
            cursor.moveToNext();
        }
        cursor.close();
    }

    return students;
}
```

Спасибо за внимание

Тренер:

Пономарев Алексей

oleksii_android@web-academy.com.ua

Лекция 13

Loaders, ContentProviders

