

Андройд – Лекц 8

Ж.Золжаргал
(j.zoljargal@csms.edu.mn)



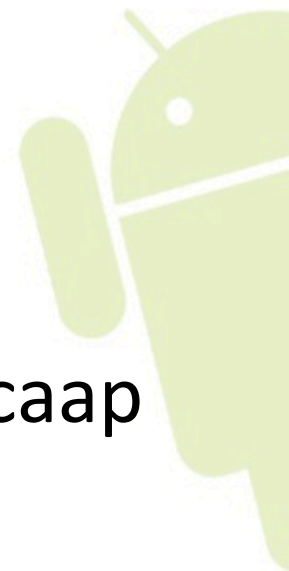
Introducing Content Providers



- Content Provider нь интерфэйсийн түвшний механизм бөгөөд Апп хооронд өгөгдөл хуваалцахад ашиглагддаг
- Permission тодорхойлох боломжтой бөгөөд URI загвараар хандана
- Таны Апп-аас тухайн утасны Contact Manager, Media Player, ... хандах боломжтой

Using Content Providers

- Content Provider-г ContentResolver классаар хэрэгжүүлдэг
- Апп бүр ContentResolver классын нэг л тохиолдолтой байна



Querying for Content

- Content Provider-н query нь ӨС-н query функцтэй төстэй
- ContentResolver-н query функц нь дараах параметруудыг авдаг
 - Content Provider-ийн URI
 - Үр дүнд сонгож авах талбарууд
 - Where нөхцөл (? Хэрэглэх боломжтой)
 - ? Тэмдэгийн оронд солигдох утгууд
 - Эрэмблэх талбар



Adding, Updating, and Deleting Content



- Content Provider-руу бичлэг хийх insert болон bulkInsert гэсэн 2 функц Content Resolver-д байдаг
- Insert функц нь шинээр нэмэгдсэн бичлэгийн URI-г буцаах бол bulkInsert функц нь нэмэгдсэн бичлэгийн тоог буцаана

Insert

```
// Create a new row of values to insert.
ContentValues newValues = new ContentValues();
```

```
// Assign values for each row.
newValues.put(COLUMN_NAME, newValue);
[ ... Repeat for each column ... ]
```

```
Uri myRowUri = getContentResolver().insert(MyProvider.CONTENT_URI,
                                           newValues);
```

```
// Create a new row of values to insert.  
ContentValues[] valueArray = new ContentValues[5];
```

```
// TODO: Create an array of new rows
```

```
int count = getResolver().bulkInsert(MyProvider.CONTENT_URI,
                                     valueArray);
```

Deletes

- URI-г дамжуулж нэг бичлэг устгах боломжтой эсвэл where нөхцөлийн тусламжтай олон бичлэг устгаж болно

```
// Remove a specific row.
```

```
getContentResolver().delete(myRowUri, null, null);
```

```
// Remove the first five rows.
```

```
String where = "_id < 5";
```

```
getContentResolver().delete(MyProvider.CONTENT_URI, where, null);
```



Updates



- Update функцээр нь харгалцах Content Provider-н URI, талбарт харгалзах утгыг агуулсан ContentValues объект ба where нөхцөлийг дамжуулна

```
// Create a new row of values to insert.
ContentValues newValues = new ContentValues();

// Create a replacement map, specifying which columns you want to
// update, and what values to assign to each of them.
newValues.put(COLUMN_NAME, newValue);

// Apply to the first 5 rows.
String where = "_id < 5";

getContentResolver().update(MyProvider.CONTENT_URI, newValues, where,
                             null);
```

Accessing Files in Content Providers



- Content Provider-руу файл хийх болон хадгалагдсан файл руу хандахдаа Content Resolvers-н `openOutputStream` эсвэл `openInputStream` функцуудыг ашиглана

```
// Insert a new row into your provider, returning its unique URI.
Uri uri = getContentResolver().insert(MyProvider.CONTENT_URI,
                                     newValues);

try {
    // Open an output stream using the new row's URI.
    OutputStream outputStream = getContentResolver().openOutputStream(uri);
    // Compress your bitmap and save it into your provider.
    sourceBitmap.compress(Bitmap.CompressFormat.JPEG, 50, outputStream);
}
catch (FileNotFoundException e) { }
```

Native Android Content Providers

- Android.provider класс нь түгээмэл provider-руу хялбар хандах боломжыг олгодог
 - Browser – Броузерын Content Provider-руу хандах ба Bookmark, Browser history, web search зэргийг унших болон шинэчлэх боломжтой
 - CallLog – Дуудлагын жагсаалтыг үзэх болон шинэчлэх боломжтой. (incoming calls, outgoing calls, missed calls, call details, caller ID, call duration)
 - Contacts – Contact-ийн жагсаалтыг авах болон шинэчлэх боломжтой



Native Android Content Providers

- MediaStore – Тухайн төхөөрөмжний Multimedia руу хандах болон удирдах. (audio, video, image)
- Settings – Тухайн төхөөрөмжны тохиргоо руу хандах болон өөрчлөх. (Bluetooth settings, ring tones, гэх мэт)



Using the Media Store Provider



- Андройдын файл систем дээр шинэ multimedia файл нэмэгдэхэд Media Store-д давхар нэмэх ёстой ба үүгээр бусад апп хандаж чадна

```
android.provider.MediaStore.Images.Media.insertImage(  
    getResolver(),  
    sourceBitmap,  
    "my_cat_pic",  
    "Photo of my cat!");
```

Using the Contacts Provider



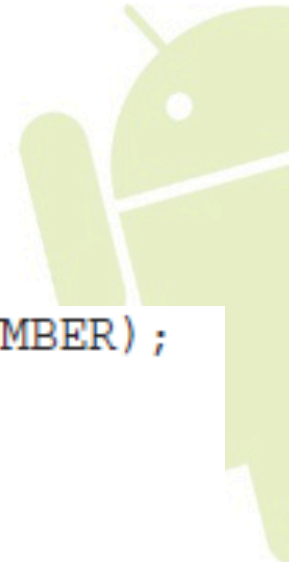
- Андройдод бүх апп Contact-н бүхий л мэдээлэлд хандах боломжтой байдаг ба READ_CONTACTS зөвшөөрөлтэй авдаг

```
// Get a cursor over every contact.
Cursor cursor = getContentResolver().query(People.CONTENT_URI,
                                           null, null, null, null);

// Let the activity manage the cursor lifecycle.
startManagingCursor(cursor);

// Use the convenience properties to get the index of the columns
int nameIdx = cursor.getColumnIndexOrThrow(People.NAME);
```

Using the Contacts Provider



```
int phoneIdx = cursor.getColumnIndexOrThrow(People.NUMBER);

String[] result = new String[cursor.getCount()];
if (cursor.moveToFirst())
    do {
        // Extract the name.
        String name = cursor.getString(nameIdx);
        // Extract the phone number.
        String phone = cursor.getString(phoneIdx);
        result[cursor.getPosition()] = name + " (" + phone + ") ";
    } while (cursor.moveToNext());
```

- READ_CONTACTS зөвшөөрлийг тухайн апп-д авах шаардлагатай

Creating a New Content Provider



- ContentProvider хийсвэр классаас удамшуулах бөгөөд onCreate үзэгдэлийг даран тодорхойлох шаардлагатай

```
import android.content.*;
import android.database.Cursor;
import android.net.Uri;
import android.database.SQLException;
```

```
public class MyProvider extends ContentProvider {

    @Override
    public boolean onCreate() {
        // TODO: Construct the underlying database.
        return true;
    }
}
```


Creating a New Content Provider



- Програмд public static CONTENT_URI тодорхойлж өгөх хэрэгтэй бөгөөд content URI нь давхцахааргүй байх хэрэгтэй
- Тиймээс URI зам нь пакежын нэр байх нь **ЗОХИМЖТОЙ**

```
content://com.<CompanyName>.provider.<ApplicationName>/<DataPath>
```

For example:

```
content://com.paad.provider.myapp/items
```

Creating a New Content Provider



- Өмнөх жишээнд бүх элемент рүү хандаж байсан бол /<rownumber> нь тухайн заасан бичлэг рүү хандана.

```
content://com.paad.provider.myapp/items/5
```

- UriMatcher нь тухайн URI-г шалгаж хаашаа хандах гэж байгааг тодорхойлно

```
private static final String myURI =  
    "content://com.paad.provider.myapp/items";  
public static final Uri CONTENT_URI = Uri.parse(myURI);
```

```
@Override
public boolean onCreate() {
    // TODO: Construct the underlying database.
    return true;
}
```

```
// Create the constants used to differentiate between the different
// URI requests.
private static final int ALLROWS = 1;
private static final int SINGLE_ROW = 2;

private static final UriMatcher uriMatcher;

// Populate the UriMatcher object, where a URI ending in 'items' will
// correspond to a request for all items, and 'items/[rowID]'
// represents a single row.
static {
    uriMatcher = new UriMatcher(UriMatcher.NO_MATCH);
    uriMatcher.addURI("com.paad.provider.myApp", "items", ALLROWS);
    uriMatcher.addURI("com.paad.provider.myApp", "items/#",
        SINGLE_ROW);
}
```

}

```
@Override
public Cursor query(Uri uri,
                    String[] projection,
                    String selection,
                    String[] selectionArgs,
                    String sort) {

    // If this is a row query, limit the result set to the passed in row.
    switch (uriMatcher.match(uri)) {
        case SINGLE_ROW :
            // TODO: Modify selection based on row id, where:
            // rowNumber = uri.getPathSegments().get(1));
    }
    return null;
}
```

```
@Override
public Uri insert(Uri _uri, ContentValues _initialValues) {
    long rowID = [ ... Add a new item ... ]

    // Return a URI to the newly added item.
    if (rowID > 0) {
        return ContentUris.withAppendedId(CONTENT_URI, rowID);
    }
    throw new SQLException("Failed to add new item into " + _uri);
}
```

[illegible][illegible]

Exposing Access to the Data Source



- Content Provider үүсгэхэд төгсгөлд нь MIME type-г тодорхойлж өгөх шаардлагатай.
- getType функцыг даран тодорхойлох бөгөөд өгөгдлийн төрөл нь давхардахгүй байх ёстой.

☐ Single Item

```
vnd.<companyname>.cursor.item/<contenttype>
```

☐ All Items

```
vnd.<companyName>.cursor.dir/<contenttype>
```

Exposing Access to the Data Source

```
@Override
public String getType(Uri _uri) {
    switch (uriMatcher.match(_uri)) {
        case ALLROWS: return "vnd.paad.cursor.dir/myprovidercontent";
        case SINGLE_ROW: return "vnd.paad.cursor.item/myprovidercontent";
        default: throw new IllegalArgumentException("Unsupported URI: " +
                                                    _uri);
    }
}
```



Registering Your Provider



- Content provider-аа үүсгэсэн бол Manifest XML файлд application tag-д тодорхойлж өгнө
- Authorities атрибутаг хаягийг нь зааж өгнө

```
<provider android:name="MyProvider"  
          android:authorities="com.paad.provider.myapp" />
```