

데이터 관리 1

성신여자대학교 컴퓨터공학과
우 종 정 교수

데이터베이스

■ 의미

- 안드로이드는 SQLite라는 내장형 데이터베이스를 제공
- 파일과 마찬가지로 DB는 기본적으로 생성된 애플리케이션에 종속
- 가벼운 파일 기반인 SQLite DB는 내장형 디바이스에 적합
- SQLite DB는 DB의 생성 및 오픈과 버전 관리를 담당하는 SQLiteOpenHelper 클래스와 DB에 대한 데이터의 추가 · 삭제 · 수정 및 질의 작업을 담당하는 SQLiteDatabase 클래스를 제공

■ DB의 위치

- data/data/<package_name>/databases/<데이터베이스이름>.db

데이터베이스

■ DB adapter의 구성

- 일반적으로 생성자
- SQLiteOpenHelper 클래스를 확장한 내부 클래스
- DB의 열기 및 닫기 메소드
- 레코드의 추가 · 삭제 · 수정 및 질의 작업을 수행하는 메소드

■ SQLiteOpenHelper 클래스의 확장

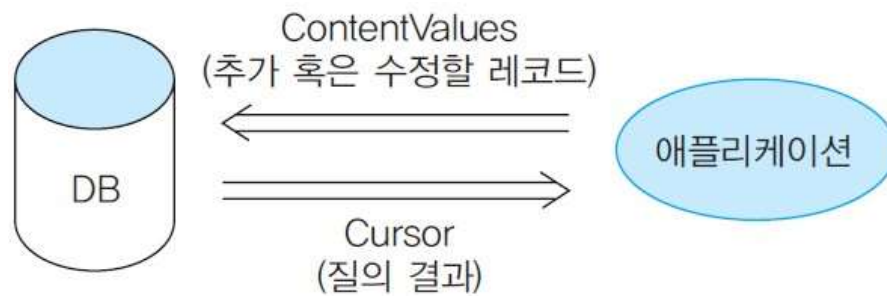
- 생성자를 구현
- 또한 상속된 다음 2개의 추상 메소드를 구현
 - `public abstract void onCreate (SQLiteDatabase db)`
 - `public abstract void onUpgrade (SQLiteDatabase db, int oldVersion, int newVersion)`

레코드 접근

■ 데이터베이스의 목적

- 궁극적으로 레코드를 추가 · 삭제 · 수정
- 질의한 결과인 레코드 내용을 화면에 출력

■ ContentValues와 Cursor



레코드 접근

■ 레코드의 추가 및 갱신

- ContentValues는 레코드의 갱신 대상 필드의 값을 준비하기 위하여 사용하는 클래스
- ContentValues의 빈 객체를 생성한 후 다음과 같은 put() 메소드를 사용하여 레코드를 추가 혹은 갱신할 준비
 - public void put (String key, Integer value)
 - public void put (String key, byte[] value)
 - public void put (String key, String value)
- 메소드
 - public long insert (String table, String nullColumnHack, ContentValues values)
 - public int update (String table, ContentValues values, String whereClause, String[] whereArgs)
 - public int delete (String table, String whereClause, String[] whereArgs)

레코드 접근

■ 레코드의 질의

- 커서 객체는 질의에 의하여 반환
- 결과의 레코드 집합에 대한 무작위 접근을 제공
- 메소드
 - `public Cursor query (String table, String[] columns, String selection, String[] selectionArgs, String groupBy, String having, String orderBy)`
- Cursor 인터페이스가 무작위 접근을 위하여 제공하는 메소드
 - `public abstract int getColumnCount ()`
 - `public abstract int getCount ()`
 - `public abstract int getInt (int columnIndex)`
 - `public abstract int getPosition ()`
 - `public abstract String getString (int columnIndex)`
 - `public abstract boolean moveToFirst ()`
 - `public abstract boolean moveToLast ()`
 - `public abstract boolean moveToNext ()` 등

레코드 접근

■ 커서와 관련된 adapter

- 커서를 adapter에 바인딩한 후 adapter뷰를 통하여 출력 가능
- 안드로이드는 커서가 지정하는 열 혹은 필드를 텍스트뷰와 같은 위젯에 사상하는 SimpleCursorAdapter라는 adapter를 제공
- 생성자
`public SimpleCursorAdapter (Context context, int layout, Cursor c, String[] from, int[] to)`
- 그러나 SimpleCursorAdapter는 현재 폐기중

레코드 접근

■ Cursor.moveToFirst(), Cursor.getInt() 등

RecNo	_id	name	contact	email
1	1	김태희	01011111111	one@first.com
2	2	송혜교	01022222222	two@second.com
3	3	제시카	01033333333	three@third.com
4	4	김연아	01044444444	four@fourth.com
5	5	황진이	01055555555	six@sixth.com
6	6	배장화	01066666666	seven@seventh.com

Cursor.moveToFirst ();
현재 커서가 위치한
레코드의 값이 대상

Cursor.getInt(0) = 1

Cursor.getInt(3) = one@first.com

- DB에 정의된 필드의 타입에 의하여 getInt(), getString()으로 값을 획득

레코드 접근

■ Cursor.moveToFirst(), Cursor.getInt() 등

RecNo	_id	name	contact	email
1	1	김태희	01011111111	one@first.com
2	2	송혜교	01022222222	two@second.com
3	3	제시카	01033333333	three@third.com
4	4	김연아	01044444444	four@fourth.com
5	5	황진이	01055555555	six@sixth.com
6	6	배장화	01066666666	seven@seventh.com

← Cursor.moveToFirst ();
현재 커서가 위치한
레코드의 값이 대상

↓
Cursor.getColumnIndex("_id") = 0

↓
Cursor.getColumnIndex("email") = 3

- Cursor.getColumnIndex(String name) : name 필드에 대응하는 인덱스를 얻음

레코드 접근

■ Cursor.getPosition()

- 현재 Cursor가 참조하고 있는 행(row)의 인덱스를 얻어올 수 있음

RecNo	_id	name	contact	email
1	1	김태희	01011111111	one@first.com
2	2	송혜교	01022222222	two@second.com
3	3	제시카	01033333333	three@third.com
4	4	김연아	01044444444	four@fourth.com
5	5	황진이	01055555555	six@sixth.com
6	6	배장화	01066666666	seven@seventh.com

← Cursor.getPosition() = 0

← Cursor.getPosition() = 5

레코드 접근

■ Cursor.getColumnName()

- 현재 필드의 이름을 얻어올 수 있음

RecNo	_id	name	contact	email
1	1	김태희	01011111111	one@first.com
2	2	송혜교	01022222222	two@second.com
3	3	제시카	01033333333	three@third.com
4	4	김연아	01044444444	four@fourth.com
5	5	황진이	01055555555	six@sixth.com
6	6	배장화	01066666666	seven@seventh.com

Cursor.getColumnName(0)
= _id

Cursor.getColumnName(3)
= email

응용

■ 예제

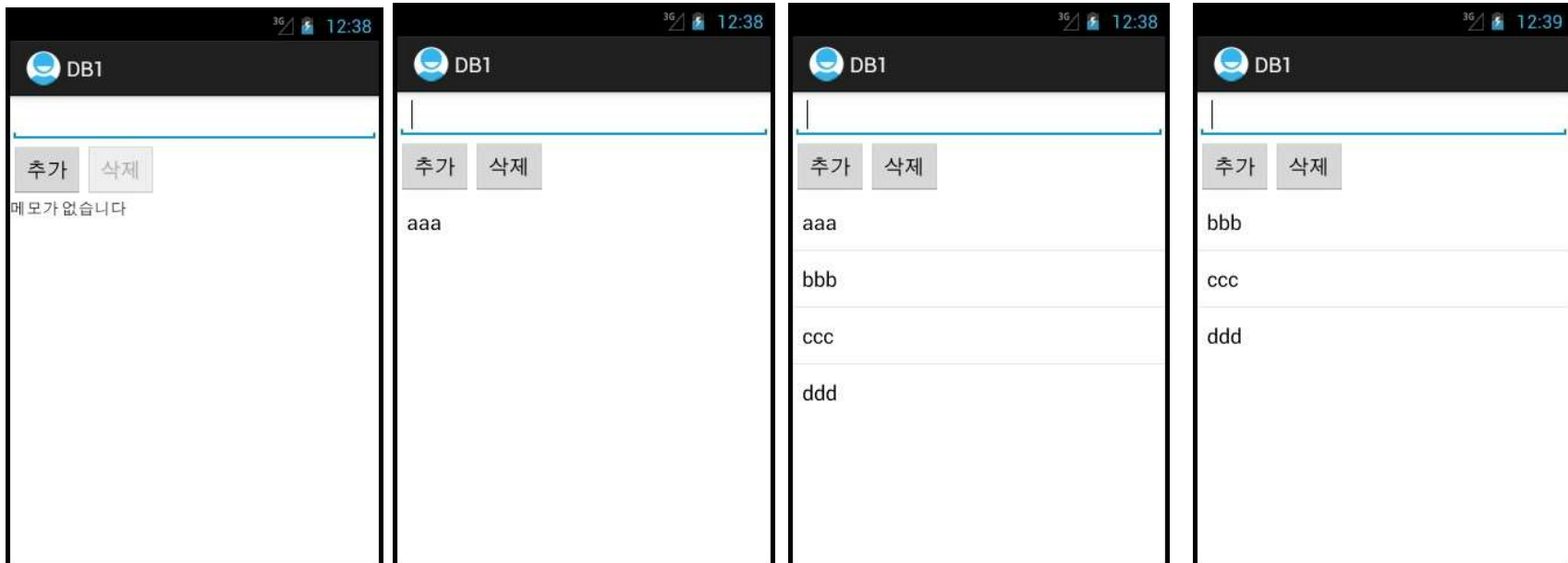
- DB0Demo : [activity](#), [helper](#), [layout](#)
- 과일의 이름과 가격으로 구성된 테이블을 포함하는 fruits.db라는 DB를 생성

DB0Demo	
사과	1500
바나나	1000
오렌지	2000

응용

■ 예제

- DB1Demo : [activity](#), [helper](#), [memo](#), [layout](#) (ArrayAdapter)
- 간단한 메모를 할 수 있는 memos라는 DB를 생성



- DB2Demo : [activity](#), [helper](#), [memo](#), [layout](#) (CursorAdapter)