**[연습문제]**

1. 다음은 각각 어떤 용어를 설명하는 것인지 써라.

(1) 열이 숫자형, 문자형, 날짜형 중 어떤 것을 입력할지는 지정

(2) 데이터베이스를 관리하는 시스템 또는 소프트웨어

(3) 테이블이 저장되는 저장소

2. 다음 각 메소드는 SQLiteOpenHelper, SQLiteDatabase, Cursor 중 어디에 속하는지 구분하라.

|  |
| --- |
| close(), query(), onCreate(), moveToLast(), getReadableDatabase(), rawQuery(), getWritableDatabase(), execSQL(), moveToFirst(), onUpgrade(), moveToNext() |

3. SQLite 명령과 그 의미를 짝지어라.

|  |  |
| --- | --- |
| (1) .table  (2) .scheam  (3) .header on  (4) .mode column  (5) .exit | a. SELECT문으로 출력할 때, 컬럼 모드로 출력해 준다.  b. 테이블의 열 및 데이터형식 등의 정보를 보여준다.  c. SQLite를 종료한다.  d. SELECT문으로 출력할 때, 헤더를 보여준다.  e. 현재 데이터베이스의 테이블 목록을 보여준다 |

4. 다음 데이터베이스 구축 절차를 차례로 나열하라.

a. 데이터 입력

b 데이터베이스 생성

c. 테이블 생성

d. 데이터 조회

5. <실습 12-1>을 참조해서 다음 조건에 맞춰서 테이블을 생성하라.

- 데이터베이스 이름은 ex12\_5DB 로 한다.

- 테이블이름은 prodTable로 열이름은 num, uName, product, count로 한다.

- Primary Key를 순번으로 한다.

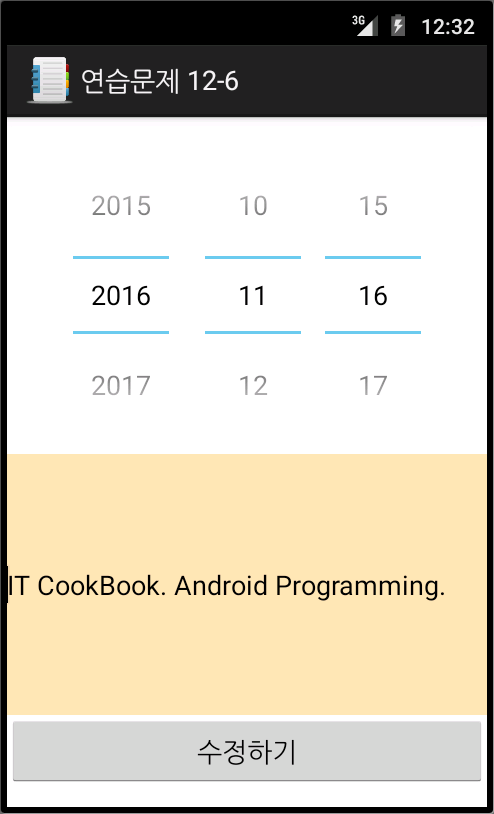
- 순번은 Insert문에서 별도로 입력하지 않고, 자동으로 1부터 시작해서 1씩 증가되어 입력되도록 한다.

|  |  |  |  |
| --- | --- | --- | --- |
| **순번** | **이름** | **물품명** | **수량** |
| 1 | 장동건 | 운동화 | 2 |
| 2 | 원빈 | 노트북 | 1 |
| 3 | 소지섭 | 모니터 | 1 |
| 4 | 김제동 | 모니터 | 5 |

6. <직접 풀어보기 8-1>를 일기장을 내장 메모리가 아닌 SQLite에 저장되도록 수정하라.

단, 데이터베이스는 myDB로, 테이블은 myDiary로 한다. 테이블의 열은 다음 표와 같다.

|  |  |
| --- | --- |
| 열이름 | 데이터형식 |
| diaryDate | char(10) |
| content | varchar(500) |



**[연습문제-답안]**

1. (1) 데이터 형식

(2) DBMS

(3) 테이블

2.

SQLiteOpenHelper : onCreate(), onUpgrade(), getReadableDatabase(), getWritableDatabase()

SQLiteDatabase : execSQL(), close(), query(), rawQuery()

Cursor : moveToFirst(), moveToLast(), moveToNext()

3.

(1) - e, (2) - b, (3) - d, (4) - a, (5) - c

4. b -> c -> a -> d

5.

C:\CookAndroid\sdk\platform-tools>adb shell

# cd /data/data/com.cookandroid.exercise12\_5

cd /data/data/com.cookandroid.exercise12\_5

# ls

ls

cache

lib

# mkdir databases

mkdir databases

# cd databases

cd databases

# pwd

pwd

/data/data/com.cookandroid.exercise12\_5/databases

# sqlite3 ex12\_5DB

sqlite3 ex12\_5DB

SQLite version 3.7.4

Enter ".help" for instructions

Enter SQL statements terminated with a ";"

sqlite> CREATE TABLE prodTable (num INTEGER PRIMARY KEY AUTOINCREMENT, uName CHA

R(10), product CHAR(10), count INT);

CREATE TABLE prodTable (num INTEGER PRIMARY KEY AUTOINCREMENT, uName CHAR(10),p

roduct CHAR(10), count INT);

sqlite> INSERT INTO prodTable VALUES(NULL, '장동건', '운동화', 2);

INSERT INTO prodTable VALUES(NULL, '장동건', '운동화', 2);

sqlite> INSERT INTO prodTable VALUES(NULL, '원빈', '노트북', 1);

INSERT INTO prodTable VALUES(NULL, '원빈', '노트북', 1);

sqlite> INSERT INTO prodTable VALUES(NULL, '소지섭', '모니터', 1);

INSERT INTO prodTable VALUES(NULL, '소지섭', '모니터', 1);

sqlite> INSERT INTO prodTable VALUES(NULL, '김제동', '모니터', 5);

INSERT INTO prodTable VALUES(NULL, '김제동', '모니터', 5);

sqlite> .header on

.header on

sqlite> .mode column

.mode column

sqlite> SELECT \* FROM prodTable;

SELECT \* FROM prodTable;

num uName product count

---------- ---------- ---------- ----------

1 장동건 운동화 2

2 원빈 노트북 1

3 소지섭 모니터 1

4 김제동 모니터 5

sqlite>

6.

\*\*\* activity\_main.xml \*\*\*

<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"

android:layout\_width="fill\_parent"

android:layout\_height="fill\_parent"

android:gravity="center"

android:orientation="vertical" >

<DatePicker

android:id="@+id/datePicker1"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:calendarViewShown="false" />

<EditText

android:id="@+id/edtDiary"

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:background="#ff00ff"

android:lines="8" >

</EditText>

<Button

android:id="@+id/btnWrite"

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:enabled="false"

android:text="Button" />

</LinearLayout>

\*\*\* Java 코드 \*\*\*

package com.cookandroid.exercise12\_6;

import java.io.File;

import java.util.Calendar;

import android.app.Activity;

import android.content.Context;

import android.database.Cursor;

import android.database.sqlite.SQLiteDatabase;

import android.database.sqlite.SQLiteOpenHelper;

import android.os.Bundle;

import android.view.View;

import android.widget.Button;

import android.widget.DatePicker;

import android.widget.EditText;

import android.widget.Toast;

public class MainActivity extends Activity {

DatePicker dp;

EditText edtDiary;

Button btnWrite;

String fileName;

File myDir;

String strSDpath;

myDBHelper myHelper;

SQLiteDatabase sqlDB;

@Override

public void onCreate(Bundle savedInstanceState) {

super.onCreate(savedInstanceState);

setContentView(R.layout.activity\_main);

setTitle("연습문제 12-6");

dp = (DatePicker) findViewById(R.id.datePicker1);

edtDiary = (EditText) findViewById(R.id.edtDiary);

btnWrite = (Button) findViewById(R.id.btnWrite);

int cYear = Calendar.getInstance().get(Calendar.YEAR);

int cMonth = Calendar.getInstance().get(Calendar.MONTH);

int cDay = Calendar.getInstance().get(Calendar.DAY\_OF\_MONTH);

// 데이터베이스 초기화

myHelper = new myDBHelper(this);

sqlDB = myHelper.getWritableDatabase();

myHelper.onUpgrade(sqlDB, 1, 2);

sqlDB.close();

// 처음의 실행한 날짜(오늘)를 체크하기

fileName = Integer.toString(cYear) + "\_" + Integer.toString(cMonth + 1)

+ "\_" + Integer.toString(cDay);

String str = readDiary(fileName);

edtDiary.setText(str);

btnWrite.setEnabled(true);

dp.init(cYear, cMonth, cDay, new DatePicker.OnDateChangedListener() {

public void onDateChanged(DatePicker view, int year,

int monthOfYear, int dayOfMonth) {

fileName = Integer.toString(year) + "\_"

+ Integer.toString(monthOfYear + 1) + "\_"

+ Integer.toString(dayOfMonth);

String str = readDiary(fileName);

edtDiary.setText(str);

btnWrite.setEnabled(true);

}

});

btnWrite.setOnClickListener(new View.OnClickListener() {

public void onClick(View v) {

if (btnWrite.getText().toString() == "새로저장") {

sqlDB = myHelper.getWritableDatabase();

sqlDB.execSQL("INSERT INTO myDiary VALUES ( '" + fileName

+ "' ,'" + edtDiary.getText().toString() + "');");

sqlDB.close();

Toast.makeText(getApplicationContext(), "입력됨",

Toast.LENGTH\_SHORT).show();

} else {

sqlDB = myHelper.getWritableDatabase();

sqlDB.execSQL("UPDATE myDiary SET content = '"

+ edtDiary.getText().toString()

+ "' WHERE diaryDate = '" + fileName + "' ;");

sqlDB.close();

Toast.makeText(getApplicationContext(), "수정됨",

Toast.LENGTH\_SHORT).show();

}

}

});

}

String readDiary(String fName) {

String diaryStr = null;

sqlDB = myHelper.getReadableDatabase();

Cursor cursor;

cursor = sqlDB.rawQuery("SELECT \* FROM myDiary WHERE diaryDate ='"

+ fName + "';", null);

if (cursor == null) {

edtDiary.setHint("일기없음");

btnWrite.setText("새로저장");

} else {

if (cursor.moveToFirst() == true) {

diaryStr = cursor.getString(1);

btnWrite.setText("수정하기");

} else {

edtDiary.setHint("일기없음");

btnWrite.setText("새로저장");

}

}

cursor.close();

sqlDB.close();

return diaryStr;

}

public class myDBHelper extends SQLiteOpenHelper {

public myDBHelper(Context context) {

super(context, "myDB", null, 1);

}

@Override

public void onCreate(SQLiteDatabase db) {

db.execSQL("CREATE TABLE if not exists myDiary ( diaryDate CHAR(10) PRIMARY KEY, content VARCHAR(500));");

}

@Override

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

onCreate(db);

}

}

}