

모바일 프로그래밍

12 SharedPreferences

2017 2학기

강승우

안드로이드에서 데이터 저장하기

- 애플리케이션에서 생성한 데이터를 저장하기 위한 방법
 - 파일 이용
 - 내부 저장: 내장 메모리
 - 외부 저장: 외장 메모리 (SD 카드)
 - 공유 프레퍼런스 (Shared Preferences) 이용
 - 애플리케이션 환경 설정 정보
 - 데이터베이스 이용
 - SQLite 데이터베이스
 - 서버 이용
 - 네트워크를 통한 데이터 전송

공유 프레퍼런스 (Shared Preferences)

공유 프레퍼런스

- 애플리케이션의 환경 설정
 - 게임 앱의 예
 - 사운드 on/off, 진동 on/off, 밝기
 - 애플리케이션이 종료되고 다시 실행될 때에도 설정 값이 유지되어야 함
- SharedPreferences
 - 환경 설정 정보를 저장하고 유지하는 용도로 사용
 - 기초 자료형을 키-값 쌍으로 저장하고 복원할 수 있음
 - boolean, float, int, long, String 등
 - 애플리케이션이 종료되더라도 저장된 값이 유지
 - 서로 다른 액티비티 간에 공유 가능

<http://developer.android.com/reference/android/content/SharedPreferences.html>

SharedPreferences 사용

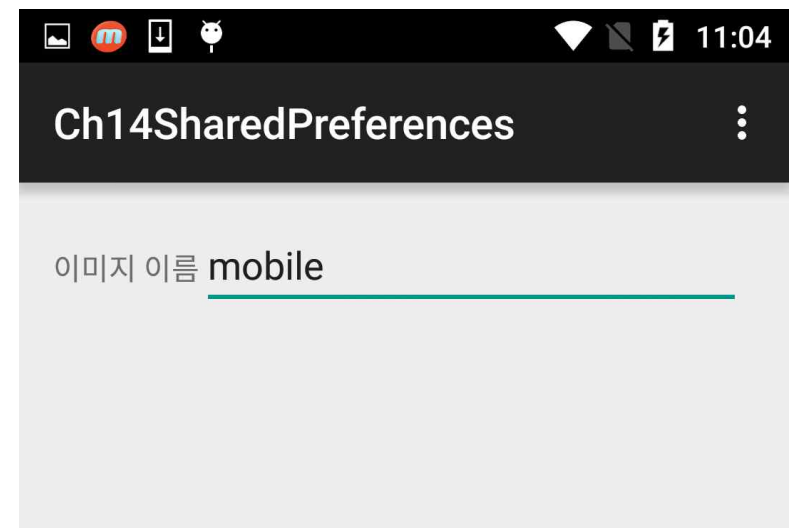
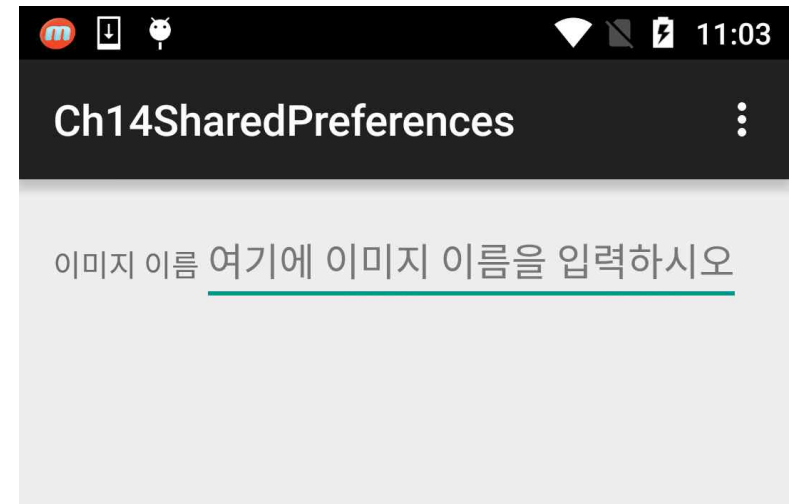
- SharedPreferences 객체 생성
 - `getSharedPreferences(String name, int mode)`
 - Context 클래스의 메소드
 - SharedPreferences 객체를 반환
 - 문자열 name을 이름으로 갖는 preference 파일을 생성
 - mode - 파일 열기 모드와 동일
 - `MODE_PRIVATE`: 정수 0에 해당 (기본적으로 이것 사용)
 - `MODE_WORLD_READABLE`, `MODE_WORLD_WRITEABLE`
- Preference 값 읽기
 - `boolean getBoolean(String key, boolean defaultValue)`
 - `int getInt(String key, int defaultValue)`
 - `String getString(String key, String defaultValue)`
 - `float getFloat(String key, float defaultValue)`
 - `long getLong(String key, long defaultValue)`

- Preference 값 쓰기

- 값을 쓸 때는 SharedPreferences 클래스의 메소드를 쓰는 게 아니라 SharedPreferences.Editor 클래스에 정의된 메소드를 사용
- SharedPreferences.Editor 객체는 SharedPreferences의 edit() 메소드를 이용하여 참조
- Editor putBoolean(String key, boolean value)
- Editor putInt(String key, int value)
- Editor putString(String key, String value)
- Editor putFloat(String key, float value)
- Editor putLong(String key, long value)
- boolean commit()
 - 변경된 내용을 즉시 파일로 저장
- void apply()
 - 우선 메모리에 변경사항을 적용한 후 비동기적으로 파일에 저장
 - 응답 시간이 빠르다 / 파일 저장 성공 여부를 확인할 수 없다

예제

- 문자열을 입력하고 백 버튼을 눌러서 앱을 종료한 후에 앱을 다시 실행하더라도 이전에 입력한 문자열이 그대로 유지되는 예
- 예제 프로젝트 이름
 - 11_SharedPreferences\
Ch14SharedPreferences



액티비티가 종료되기 전에 SharedPreferences에 데이터 저장

@Override

```
protected void onStop() {  
    super.onStop();
```

PREFS_NAME: String 값, 코드에서는 아래와 같이
String 상수로 정의되어 있음
public static final String PREFS_NAME = "MyPrefs";

```
    SharedPreferences settings = getSharedPreferences(PREFS_NAME, MODE_PRIVATE);  
    SharedPreferences.Editor editor = settings.edit();
```

```
    imageName = value.getText().toString();  
    editor.putString("imageName", imageName);  
    editor.apply();  
}
```


액티비티가 시작될 때 SharedPreferences에서 데이터 복원

`@Override`

```
protected void onStart() {  
    super.onStart();
```

```
    SharedPreferences settings = getSharedPreferences(PREFS_NAME, MODE_PRIVATE);  
    imageName = settings.getString("imageName", "");  
    value.setText(imageName);  
}
```