

CAMERA

2020년도 1학기 앱 프로그래밍 14주차 수업



목 차

1. Video Cam
2. 앱 프로그래밍 학습을 위한 복습

1~3교시

2020년도 1학기 앱 프로그래밍 – 동영상 캡처



I.VIDEO CAM

- 카메라 영상 녹화 앱의 구성 단계
 1. AndroidManifest.xml 파일에 권한 설정
 2. CheckPermissionActivity에 권한 요청 코드 작성
 3. Camera 객체 얻기
 4. 미리보기 화면 시작
 5. 동영상 녹화 시작
 6. 동영상 녹화 중지

2. ANDROIDMANIFEST

- 동영상 캡처를 위한 권한 설정

```
<uses-permission android:name="android.permission.CAMERA"/> <!-- 카메라 기능 설치시 허가권 확인 -->
<uses-feature android:name="android.hardware.camera"/> <!-- 카메라 기능 실행 시 접근 권한 -->
<uses-permission android:name="android.permission.WRITE_EXTERNAL_STORAGE"/> <!-- SD 저장 권한 확인 -->
<uses-permission android:name="android.permission.RECORD_AUDIO"/> <!-- 오디오 녹음 권한 -->
<uses-permission android:name="android.permission.ACCESS_FINE_LOCATION"/> <!-- 위치 태그를 위한 권한 -->
```


3. CHECKPERMISSIONACTIVITY

- 허가권 요청 대화상자

```
//사용자의 os 버전이 마시멜로우(23)이상인지 판별
if(Build.VERSION.SDK_INT >= Build.VERSION_CODES.M) {
    //현재 어플리케이션이 권한에 대해 거부되었는지 확인
    if (checkSelfPermission(Manifest.permission.CAMERA) == PackageManager.PERMISSION_DENIED ||
        checkSelfPermission(Manifest.permission.RECORD_AUDIO) == PackageManager.PERMISSION_DENIED ||
        checkSelfPermission(Manifest.permission.ACCESS_FINE_LOCATION) == PackageManager.PERMISSION_DENIED ||
        checkSelfPermission(Manifest.permission.WRITE_EXTERNAL_STORAGE) == PackageManager.PERMISSION_DENIED) {

        requestPermissions(new String[]{Manifest.permission.CAMERA,
            Manifest.permission.RECORD_AUDIO, Manifest.permission.ACCESS_FINE_LOCATION,
            Manifest.permission.WRITE_EXTERNAL_STORAGE}, requestCode: 1000);
    }
    else {
        Intent intent = new Intent(getApplicationContext(), MainActivity.class);
        startActivity(intent);
        finish();
    }
}
else {
    Intent intent = new Intent(getApplicationContext(), MainActivity.class);
    startActivity(intent);
    finish();
}
```

3. CHECKPERMISSIONACTIVITY(계속)

- 허가권 요청에 대한 응답 처리

```
//요청에 대한 응답을 처리하는 부분
//@param requestCode : 요청코드(1000)
//@param permissions : 사용자가 요청한 권한들(개발자) String배열
//@param grantResults : 권한에 대한 응답들(인덱스 별로 매칭)
@Override
public void onRequestPermissionsResult(int requestCode, @NonNull String[] permissions, @NonNull int[] grantResults) {
    if(requestCode==1000){
        if(grantResults.length>0 &&
            grantResults[0]== PackageManager.PERMISSION_GRANTED &&
            grantResults[1] == PackageManager.PERMISSION_GRANTED &&
            grantResults[2] == PackageManager.PERMISSION_GRANTED &&
            grantResults[3] == PackageManager.PERMISSION_GRANTED){

            if(ActivityCompat.checkSelfPermission( context: this, Manifest.permission.CAMERA) != PackageManager.PERMISSION_DENIED &&
                ActivityCompat.checkSelfPermission( context: this, Manifest.permission.RECORD_AUDIO) != PackageManager.PERMISSION_DENIED &&
                ActivityCompat.checkSelfPermission( context: this, Manifest.permission.ACCESS_FINE_LOCATION) != PackageManager.PERMISSION_DENIED &&
                ActivityCompat.checkSelfPermission( context: this, Manifest.permission.WRITE_EXTERNAL_STORAGE) != PackageManager.PERMISSION_DENIED){
                Intent intent=new Intent(getApplicationContext(),MainActivity.class);
                startActivity(intent);
                finish();
            }else{
                Toast.makeText( context: this, text: "권한 요청을 거부하였습니다", Toast.LENGTH_SHORT).show();
            }
        }
        else {
            Toast.makeText(getApplicationContext(), text: "권한 설정 실패!\n앱을 다시 설치하세요!\n앱 종료!", Toast.LENGTH_SHORT).show();
            finish();
        }
    }
}
```

4. CAMERA 객체 얻기

```
// 카메라 하드웨어 감지
private boolean checkCameraHardware(Context context){
    if(context.getPackageManager().hasSystemFeature(PackageManager.FEATURE_CAMERA))
        return true; // 카메라 있음!
    else
        return false; // 카메라 없음
}

public static Camera getCameraInstance(){
    Camera c = null;
    try {
        c = Camera.open();
    }
    catch (Exception e){}
    return c;
}
```


5. CAMERA PREVIEW

```
if(checkCameraHardware( context: this)) {  
    // 미리보기 뷰 생성 후 액티비티 내의 컨테츠로 설정한다.  
    mPreview = new CameraPreview( context: this, mCamera);  
    FrameLayout preview = (FrameLayout) findViewById(R.id.camera_preview);  
    preview.addView(mPreview);  
  
    hideSystemUI();  
}  
else{  
    Toast.makeText(getApplicationContext(), text: "장치에 카메라를 지원하지 않습니다.\n앱을 종료합니다!", Toast.LENGTH_SHORT).show();  
}
```

6. CAPTURE VIDEO

- 사용 클래스
 - Camera & MediaRecorder 클래스
- 사용 메서드
 - 카메라 객체 관리 메서드
 - Camera.open() & Camera.release()
 - MediaRecorder 의 카메라 하드웨어 접근 관리
 - Camera.unlock() → MediaRecorder에 카메라 하드웨어 접근 허용
 - Camera.lock() → MediaRecorder에 카메라 하드웨어 접근 미 허용
 - MediaRecorder.setCamera() → 카메라 객체를 통한 미디어 레코딩 기능 설정
 - MediaRecorder.setAudioSource() & MediaRecorder.setVideoSource() → 오디오, 비디오 소스 설정
 - MediaRecorder.setProfile() → 캠코더 프로파일 설정(동영상 출력 형식, 인코딩 설정)
 - MediaRecorder.setOutputFile() → 동영상 저장 파일 설정
 - MediaRecorder.setPreviewDisplay() → 미디어 레코더에 미리보기 화면 설정
 - MediaRecorder.prepare() → 미디어 레코드 준비
 - MediaRecorder.start() & MediaRecorder.stop() → 녹화 시작 및 중지

6. CAPTURE VIDEO(계속)

- 동영상 녹화 시작 과정
 1. 사용할 카메라 잠금 해제 → `Camera.unlock()`
 2. `MediaRecorder` 구성 → 카메라 설정, A/V 소스, 인코딩 형식, 출력 파일, 미리보기
 3. `MediaRecorder` 준비 → 미디어 레코더의 구성 설정으로 미디어 레코드 준비
 4. `MediaRecorder` 시작 → 동영상 녹화 시작
- 동영상 녹화 종료 과정
 1. `MediaRecorder` 정지 → 동영상 녹화 정지
 2. `MediaRecorder` 재설정 → 미디어 레코더 구성 설정 삭제
 3. `MediaRecorder` 해제 → `MediaRecorder` 객체 해제

6. CAPTURE VIDEO(계속)

- 준비 과정

```
private boolean prepareVideoRecorder(){
    mediaRecorder = new MediaRecorder();

    // Step 1: 카메라 언락후 미디어 레코더에 카메라 객체 설정
    mCamera.unlock();
    mediaRecorder.setCamera(mCamera);
    // Step 2: 오디오 비디오 소스 설정
    mediaRecorder.setAudioSource(MediaRecorder.AudioSource.CAMCORDER);
    mediaRecorder.setVideoSource(MediaRecorder.VideoSource.CAMERA);
    // Step 3: 캡코더프로파일을 통해 동영상 출력 형식과 인코딩 설정(API 레벨 8 이상 요구)
    mediaRecorder.setProfile(CamcorderProfile.get(CamcorderProfile.QUALITY_HIGH));
    // Step 4: 동영상 파일 설정
    mediaRecorder.setOutputFile(getOutputMediaFile(MEDIA_TYPE_VIDEO).toString());
    // Step 5: 미디어 레코더에 미리보기 화면을 설정한다.
    mediaRecorder.setPreviewDisplay(mPreview.getHolder().getSurface());
    // Step 6: 미디어 레코드 준비
    try {
        mediaRecorder.prepare();
    } catch (IllegalStateException e) {
        Log.d( tag: "CAMERA", msg: "IllegalStateException preparing MediaRecorder: " + e.getMessage());
        releaseMediaRecorder();
        return false;
    } catch (IOException e) {
        Log.d( tag: "CAMERA", msg: "IOException preparing MediaRecorder: " + e.getMessage());
        releaseMediaRecorder();
        return false;
    }
    return true;
}
```


6. CAPTURE VIDEO(계속)

- 녹화 버튼

```
btn_video = findViewById(R.id.btn_video);
btn_video.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        if (isRecording) {
            // 녹화 중지 및 카메라 해제
            mediaRecorder.stop(); // 녹화 중지
            releaseMediaRecorder(); // 미디어 레코드 객체 해제
            mCamera.lock(); // 미디어 레코더로 카메라 접근 잠금

            // 녹화 중지 상태 알림
            btn_video.setText("Video");
            releaseMediaRecorder();
            isRecording = false;
        } else {
            // 비디오 카메라 초기화
            if (prepareVideoRecorder()) {
                // 카메라가 사용가능하고 잠금 해제되면 미디어 레코더가 준비되며 녹화 시작이 가능하다.
                mediaRecorder.start();

                // 녹화 중 상태 알림
                btn_video.setText("Recording");
                isRecording = true;
            } else {
                // 미디어 레코드 준비가 안되면 카메라 해제
                releaseMediaRecorder();
            }
        }
    }
});
```


6. CAMERA & MEDIARECORDER RELEASE

- 레코더 해제
- 카메라 해제

```
private void releaseMediaRecorder(){
    if (mediaRecorder != null) {
        mediaRecorder.reset();    // 레코더 설정 삭제
        mediaRecorder.release(); // 레코더 객체 해제
        mediaRecorder = null;
        mCamera.lock();           // 다음 사용을 위해 카메라 접근 잠금 설정
    }
}

private void releaseCamera(){
    if (mCamera != null){
        mCamera.release();        // 다른 앱을 위해 카메라 해제
        mCamera = null;
    }
}
```

6. 앱 프로그래밍 학습을 위한 실습 (과제)

- 실습 문제
 - 동영상 녹화 기능을 완료하여 제출하세요!
- 제출 방법
 - <http://ctl.gtec.ac.kr> 의 과제 제출란에 제출
 - Java 파일과 xml 파일을 제출한다.
 - 압축파일은 자신의 “반_학번_이름.zip” 로 한다.
 - 제출 일자와 시간을 엄수하여 제출하세요!