Dangerous Permissions

- Dangerous permissions are permissions that may pose a risk to the user's privacy.

- Starting from Android 6.0 (API Level 23), an app that uses a dangerous permission must ask the user for approval at run time. After the user has granted permission, the app should remember and not ask again.

- If the developer did not include codes to ask for permission, the user will need to manually grant permission through the device's Application Manager. Otherwise, the app may not work and may crash.
Obviously, we want to write codes to show a screen prompt that first time the user runs the app.

To make use of the audio recorder for example, you need to do the following:

- Add the following permission statement in AndroidManifest.xml

```
<uses-permission
android:name="android.permission.RECORD_AUDIO" />
```

Example of using multiple dangerous permissions:

- Add the following permission statements in AndroidManifest.xml

```xml
<uses-permission
android:name="android.permission.RECORD_AUDIO" />


<uses-permission
android:name="android.permission.CAMERA" />

<uses-feature android:name="android.hardware.camera" />
<uses-feature
android:name="android.hardware.camera.autofocus" />
```

- In the activity that requires the permission, add the codes in the next 3 slides to request for permission during run time

```java
private static final int REQUEST_RECORD_AUDIO_PERMISSION = 200;

private static final int REQUEST_CAMERA_PERMISSION      = 300;

private boolean permissionToRecordAccepted = false;

private boolean permissionToUseCameraAccepted = false;

private String [] permissions = {Manifest.permission.RECORD_AUDIO,
Manifest.permission.CAMERA};
```

```java
@Override
public void onRequestPermissionsResult(int requestCode, @NonNull String[] permissions,
@NonNull int[] grantResults) {
    super.onRequestPermissionsResult(requestCode, permissions, grantResults);

    switch (requestCode) {
        case REQUEST_RECORD_AUDIO_PERMISSION:
            if (grantResults.length>0) {
                permissionToRecordAccepted = grantResults[0] ==
PackageManager.PERMISSION_GRANTED;
            }
            break;
        case REQUEST_CAMERA_PERMISSION:
            if (grantResults.length>1) {
                permissionToUseCameraAccepted = grantResults[1] ==
PackageManager.PERMISSION_GRANTED;
            }
            break;
    }

    // if (!permissionToRecordAccepted ) finish();
    if ((!permissionToRecordAccepted ) || (!permissionToUseCameraAccepted)) finish();

}
```

```java
@Override
protected void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    setContentView(R.layout.activity_main);

    if (ContextCompat.checkSelfPermission(this,
            Manifest.permission.RECORD_AUDIO)
            != PackageManager.PERMISSION_GRANTED) {

        ActivityCompat.requestPermissions(this,
                permissions,
                REQUEST_RECORD_AUDIO_PERMISSION);
    }

    if (ContextCompat.checkSelfPermission(this, Manifest.permission.CAMERA)
            == PackageManager.PERMISSION_DENIED) {

        ActivityCompat.requestPermissions(this, permissions, REQUEST_CAMERA_PERMISSION);

        //ActivityCompat.requestPermissions(this, new String[] {Manifest.permission.CAMERA},
        REQUEST_CAMERA_PERMISSION);
    }

}
```

```java
private static final int REQUEST_RECORD_AUDIO_PERMISSION = 200;

// Requesting permission to RECORD_AUDIO
private boolean permissionToRecordAccepted = false;
private String [] permissions = {Manifest.permission.RECORD_AUDIO};
```

```java
@Override
public void onRequestPermissionsResult(int requestCode, @NonNull
String[] permissions, @NonNull int[] grantResults) {
    super.onRequestPermissionsResult(requestCode, permissions,
grantResults);
    switch (requestCode){
        case REQUEST_RECORD_AUDIO_PERMISSION:
            permissionToRecordAccepted  = grantResults[0] ==
PackageManager.PERMISSION_GRANTED;
            break;
    }
    if (!permissionToRecordAccepted ) finish();

}
```

Add the following either in onCreate() or onResume():

```
if (ContextCompat.checkSelfPermission(this,
        Manifest.permission.RECORD_AUDIO)
        != PackageManager.PERMISSION_GRANTED) {

    ActivityCompat.requestPermissions(this,
        permissions,
        REQUEST_RECORD_AUDIO_PERMISSION);
}
```