**Acessando a câmera do dispositivo**

A proposta dessa aula bônus é que possamos acessar a câmera do nosso dispositivo, tirar uma foto e após isso, atualizar uma ImageView com a mesma.

Maiores informações sobre o uso da action “ACTION\_IMAGE\_CAPTURE”, responsável por acessar a câmera do dispositivo e obter um retorno, podem ser obtidas através do link <http://developer.android.com/reference/android/provider/MediaStore.html>.

**Preparando o Layout do nosso App**

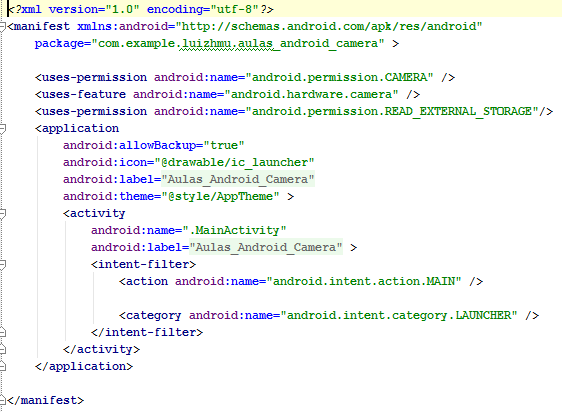
O layout utilizado para esse App será bem simples, composto por uma ImageView que será responsável por acessar a câmera, após ação do usuário e também por atualizar a foto tirada por ele. Além da ImageView, utilizaremos dois botões sendo um responsável para rotacionar a foto em 90º (caso necessário) e outro para voltar a foto à sua posição inicial.

Vocês perceberão que essa funcionalidade de rotacionar a foto será muito útil, pois diversos celulares atualizam a foto de forma invertida.



**Alterando o AndroidManifest.xml**

Será necessário adicionar as permissões para acesso à câmera, conforme ilustração abaixo.



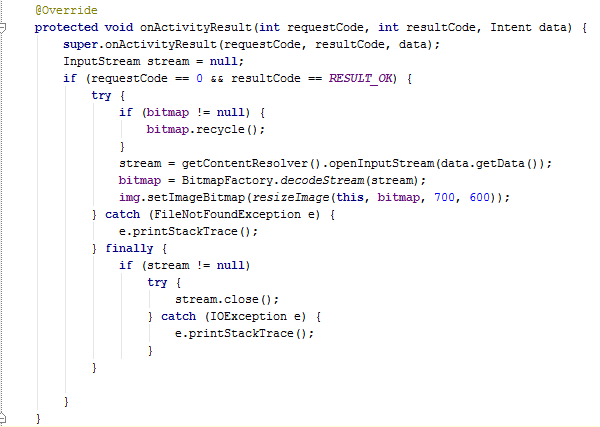
**Implementando nossa classe principal**

Primeiramente, foi criado o método “abrirCamera”, que foi associado à ImageView.

Neste método, consta a intent de chamada para acesso à câmera, enviando o requestCode=0.

“startActivityForResult(intent,0);”

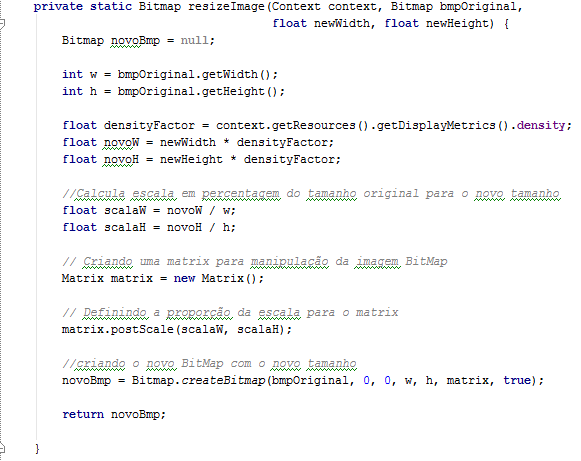
Após essa implementação inicial, deveremos implementar nossa onActivityResult, que será responsável por realizar as funcionalidades da câmera, retornar e redimensionar o bitmap, além de tratar todas as exceções.



Apenas com essas informações, o programa já funcionaria, porém vamos acrescentar ao nosso projeto, a funcionalidade responsável por rotacionar e redimensionar a foto, no momento em que for atualizada na ImageView.

Abaixo função responsável por retornar o Bitmap redimensionado, conforme altura e largura que definiremos no nosso onActivityResult.

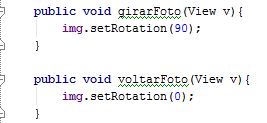
“img.setImageBitmap(resizeImage(this, bitmap, 700, 600));”



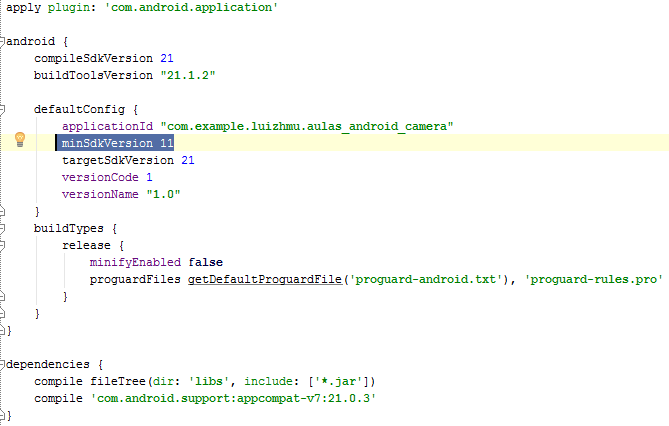
Por fim, criaremos dois métodos para rotacionar e voltar a foto na sua posição inicial. Os métodos estão atrelados aos botões, criados no nosso layout.

android:onClick="girarFoto"

android:onClick="voltarFoto"



É importante ressaltar que a funcionalidade “setRotation” está apta a partir da versão mínima 11 do SDK, sendo necessário fazer essa alteração no build.gradle (Module.app).



Pronto, podemos executar nosso aplicativo.

Estaremos à disposição para sanar qualquer dúvida.