



Cámara y Galería

# Permisos para usar la cámara

- En **AndroidManifest.xml**:

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

- **Required** es “**true**” si la cámara es esencial para el funcionamiento de la aplicación.

# Iniciar cámara

- **En Activity:**

```
final int CODIGO_PETICION = 1;
```

```
Intent intent = new Intent(MediaStore.ACTION_IMAGE_CAPTURE);
```

```
if (intent.resolveActivity(getPackageManager()) != null) {  
    startActivityForResult(intent, CODIGO_PETICION );  
}
```

# Obtener imagen de cámara

- **En Activity:**

@Override

```
protected void onActivityResult(int requestCode, int resultCode, Intent data) {  
    if (requestCode == CODIGO_PETICION && resultCode == RESULT_OK) {  
        Bundle extras = data.getExtras();  
        Bitmap imageBitmap = (Bitmap) extras.get("data");  
        imageView.setImageBitmap(imageBitmap);  
    }  
}
```

# Permisos para leer galería

- **En AndroidManifest.xml:**

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

# Iniciar galería

- **En Activity:**

```
final int CODIGO_PETICION = 1;
```

```
Intent photoPickerIntent = new Intent(Intent.ACTION_PICK);  
photoPickerIntent.setType("image/*");  
startActivityForResult(photoPickerIntent, CODIGO_PETICION);
```

# Obtener imagen de galería

- En Activity:

@Override

```
protected void onActivityResult(int requestCode, int resultCode, Intent data) {  
    if (requestCode == CODIGO_PETICION && resultCode == RESULT_OK) {  
        try {  
            Uri uri = data.getData();  
            InputStream stream = this.getContentResolver().openInputStream(uri);  
            Bitmap selectedImage = BitmapFactory.decodeStream(stream);  
            imageView.setImageBitmap(selectedImage);  
        } catch (FileNotFoundException e) {  
            // ... Informar al usuario ...  
        }  
    }  
}
```