

- Más información en:
 - ✓ <https://developer.android.com/guide/topics/resources/available-resources?hl=es-419>
 - ✓ <https://developer.android.com/guide/topics/resources/providing-resources?hl=es-419#Accessing>
 - ✓ [https://developer.android.com/reference/android/content/Context?hl=es-419#getResources\(\)](https://developer.android.com/reference/android/content/Context?hl=es-419#getResources())

- Normalmente empleamos la orden: `getResources().getZZZZZ`

siendo ZZZZZ el tipo de recurso que queremos recuperar.

Por ejemplo, para recuperar una cadena guardada en un archivo de recursos **/res/values/** y que tenga de nombre 'cadena' emplearíamos esta orden:

```
getResources().getString(R.string.cadena);
```

Por ejemplo, para recuperar una String-Array guardado en un archivo de recursos **/res/values/** y que tenga de nombre 'datos_array' emplearíamos esta orden:

```
getResources().getStringArray(R.string.datos_array);
```

- También podemos, a partir de un nombre, obtener el id del recurso que tiene ese nombre en alguno de las carpetas **/res/**, empleando el método `getIdentifier()` que espera recibir tres datos: el nombre del recurso, el tipo de recurso y el paquete.

```
ImageView foto = findViewById(R.id.imvFoto);  
  
foto.setImageResource(getResources().getIdentifier("foto1", "drawable", getPackageName()));
```

En el ejemplo estaríamos cargando una foto (que se encuentra en **/res/drawable/foto1.jpg**) de un `ImageView`.

En este caso tenemos que tener cuidado con varios aspectos:

- ✓ El nombre del identificador que va en el primer parámetro **no lleva la extensión de la imagen a cargar**.
- ✓ El tercer parámetro es el nombre del paquete de la Activity. Si estamos dentro de una clase que no represente la actividad, necesitaremos una referencia a la misma, pudiendo emplear la forma: `NombreActivity.this.getPackageName()`