

## 1. Problemas con el paso de datos entre Activities.

Cuando se pasan datos entre activities encapsulando estos dentro de un objeto Intent (o en un Bundle) tiene un tamaño máximo que no se puede superar ya que la aplicación fallara mediante la excepción `TransactionTooLargeException`.

Para más información se pueden consultar los siguientes enlaces:

- <https://www.neotechsoftware.com/blog/android-intent-size-limit>
- <https://developer.android.com/reference/android/os/TransactionTooLargeException.html>

Según este último enlace el tamaño máximo de datos que se pueden pasar es de 1Mb. El problema es que este tamaño es realmente menor ya que este tamaño es el total compartido por todas las aplicaciones que estén funcionando en el sistema, por lo que el tamaño real que se dispone no se puede conocer de antemano y este varía entre dispositivos.

Las posibles soluciones son:

- Minimizar los datos que se pasan entre activities.
- Si se pasan datos de gran tamaño intentar dividirlos y partes más pequeñas.
- Usar persistencia, por ejemplo ficheros, para almacenar los datos y pasar a la Activity únicamente la URI del fichero.
- Crear una única copia de los datos, no hace falta que sean todos solo los que tengan un tamaño grande, y que todas la Activities usen esta única copia. Para ello podemos usar un patrón Singleton: <https://es.wikipedia.org/wiki/Singleton>.

Un ejemplo sería la siguiente clase, con la se emula el paso de datos con Intents.

```
public class Datos {  
    private static Datos instance; // Al estar internamente autoreferenciada, en lenguajes como Java, el recolector de basura no actúa.  
    private final HashMap<String, Object> extras = new HashMap<>();  
    private Datos() {  
    }  
    public static Datos getInstance() {  
        if (instance == null) {  
            instance = new Datos();  
        }  
        return instance;  
    }  
    public void putExtra(String name, Object object) {  
        extras.put(name, object);  
    }  
    public Object getExtra(String name) {  
        return extras.get(name);  
    }  
    public boolean hasExtra(String name) {  
        return extras.containsKey(name);  
    }  
    public ArrayList<Película> getPelis(String name) {  
        return (ArrayList<Película>)extras.get(name);  
    }  
    public void clear() {  
        extras.clear();  
    }  
}
```

Por comodidad se ha generado un método `getPelis` que devuelve directamente un `ArrayList` con las películas.

Para su utilización generamos la lista de películas:

```
Datos datos=Datos.getInstance();  
datos.putExtra("pelis",rellenaPelículas());
```

Ahora en datos, dentro del HashMap extras tenemos las películas almacenadas con la clave pelis.

Ahora ya no es necesario pasar este arraylist, dentro de un Intent, a las actividades. Simplemente se usa directamente donde haga falta.

Por ejemplo en la activity del listado completo:

```
Datos datos=Datos.getInstance();  
ArrayList<Película> pelis=datos.getPelis("pelis");
```

O usarlo directamente. Imaginemos que el adaptador del RecyclerView se llama adaptador:

```
Adaptador adaptador=new Adaptador(Datos.getInstance().getPelis("pelis"));
```

Si nos fijamos en la declaración de Datos el atributo instance es estático y solo se crea una única vez en getInstance la primera vez que se ejecuta este método y instance tiene un valor nulo, es resto de veces se reusa el atributo sin volver a crearlo.