


Open Opened 3 weeks ago by  **Rubén Montero**

Nueva actividad

Resumen

- Crearemos una nueva actividad
- La iniciaremos si el usuario *clica* en el botón inferior

Descripción

Nuestra aplicación tiene sólo una actividad, `MainActivity`. Vamos a crear otra actividad llamada `MonstersActivity`. Recordemos que una actividad tiene *tres* componentes fundamentales:

- Clase Java
- Interfaz XML
- Declaración en `AndroidManifest.xml`

Vayamos añadiéndolos uno a uno.

La tarea

Clase Java

Añade una nueva clase Java. **Dale** el nombre `MonstersActivity`.

En su contenido, **añade** el siguiente código:

```
package com.afundacion.fp.library;

import androidx.appcompat.app.AppCompatActivity;

public class MonstersActivity extends AppCompatActivity {
}
```

(Como ves, `MonstersActivity` hereda de `AppCompatActivity`. La herencia se declara en Java con la palabra reservada `extends`)

Interfaz XML

En las carpetas de la izquierda, **expande** `res`. **Haz click** derecho en `res` > `layout`. Ahí, **selecciona** `New` > `Layout Resource File`.

Seguramente ya nos aparezca como **Root element** `androidx.constraintlayout.widget.ConstraintLayout`. Eso está bien.

Ahora, **escribe** `activity_monsters.xml` como **File Name** y dale a OK. Se generará un XML como este:

```
<?xml version="1.0" encoding="utf-8"?>
<androidx.constraintlayout.widget.ConstraintLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="match_parent">

</androidx.constraintlayout.widget.ConstraintLayout>
```

Declaración en `AndroidManifest.xml`

Ya hemos visto que en `manifest` > `AndroidManifest.xml` se añaden etiquetas `activity` para declarar nuevas actividades. ¡Hagámoslo!

Añade `<activity android:name=".MonstersActivity"/>` a `AndroidManifest.xml`:

```
<application
...>
    <activity
        android:name=".MainActivity"
        ... >
    </activity>
```

```
<!-- Añadimos una nueva actividad debajo de la otra -->
<activity android:name=".MonstersActivity"/>

</application>
```

¡Actividad creada y declarada!

Iniciando la actividad con `startActivity`

Vamos a hacer que la *app* navegue a esta nueva actividad cuando el usuario clique el botón "Iniciar nueva actividad."

En primer lugar, **crea** un `Intent` en `MainActivity.java`, como se indica a continuación:

```
// ...
// Este código pertenece a onCreate en MainActivity.java
otherButton.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View view) {
        Toast.makeText(context, "Iniciando otra actividad...", Toast.LENGTH_LONG).show();
        // Declaramos nuestra intención
        // Nótese que el constructor también recibe
        // un parámetro de tipo 'Context'
        Intent myIntent = new Intent(context, MonstersActivity.class);
    }
});
```

Este `Intent` es algo así como una *intención*. La *intención* de iniciar la `MonstersActivity`.

Para ponerlo en funcionamiento, **invoca** `startActivity` así:

```
Intent myIntent = new Intent(context, MonstersActivity.class);
context.startActivity(myIntent);
```

¡Enhorabuena! Has creado y arrancado tu primera actividad desde cero.

Prueba el comportamiento en el emulador. ¿Al hacer *click* ves una nueva pantalla vacía?

Por último

Sube tus cambios al repositorio en un nuevo *commit*.



Rubén Montero @ruben.montero changed milestone to [%Sprint 1](#) 3 weeks ago



Ania Blanco @ania.blanco mentioned in commit [2af98022](#) 1 week ago