

Bloque 3. Ejercicio Resuelto Parcelable

Descargar este ejercicio [pdf](#) o [html](#)

Almacenamiento de información en Android

Ejercicio que nos va a servir para practicar la parcelación de unos datos con un Intent, y también la transmisión de esos datos a una APP diferente en donde serán recibidos y mostrados.

Crea una aplicación **ResueltoParcelable**, con el siguiente aspecto.



The image shows a mobile application interface titled "6_EjercicioPropuestoParcelable". It features a white background with a purple header bar. Below the header, there is a form with three input fields: "Introduce DNI", "Nombre", and "Edad". The "Nombre" and "Edad" fields are on the same line. Below these fields is a purple button labeled "MANDAR".

Si no tienes claro el código a crear, puedes usar el siguiente archivo `activity_main.xml`

```

<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:paddingTop="100dp"
    android:orientation="vertical"
    tools:context=".MainActivity" >
    <EditText
        android:id="@+id/dni"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:hint="Introduce DNI"/>
    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:orientation="horizontal">

        <EditText
            android:id="@+id/nombre"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:layout_weight="0.5"
            android:hint="Nombre"/>
        <EditText
            android:id="@+id/edad"
            android:layout_weight="0.2"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:hint="Edad"/>
    </LinearLayout>
    <Button
        android:id="@+id/mandar"
        android:layout_gravity="center"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Mandar"/>
</LinearLayout>

```

Deberá tener una clase Pojo llamada **Persona** que extenderá de Parcelable, tendrá dos tipo String: dni y nombre y un entero edad. El código podría ser:

```

class Persona():Parcelable
{
    var nombre:String?
    var dni:String?
    var edad:Int
    constructor(parcel: Parcel) : this() {
        nombre = parcel.readString()
        dni = parcel.readString()
        edad = parcel.readInt()
    }
    init{
        this.nombre=""
        this.dni=""
        this.edad=0
    }
    constructor(nombre:String?, dni:String?, edad:Int) : this() {
        this.nombre = nombre;
        this.dni = dni;
        this.edad = edad;
    }
    override fun toString(): String {
        return "Persona{" +
            "nombre='" + nombre + '\'' +
            ", dni='" + dni + '\'' +
            ", edad=" + edad +
            '\''
        }
    }
    override fun writeToParcel(parcel: Parcel, flags: Int) {
        parcel.writeString(nombre)
        parcel.writeString(dni)
        parcel.writeInt(edad)
    }
    override fun describeContents(): Int {
        return 0
    }
    companion object CREATOR : Parcelable.Creator<Persona> {
        override fun createFromParcel(parcel: Parcel): Persona {
            return Persona(parcel)
        }
        override fun newArray(size: Int): Array<Persona?> {
            return arrayOfNulls(size)
        }
    }
}

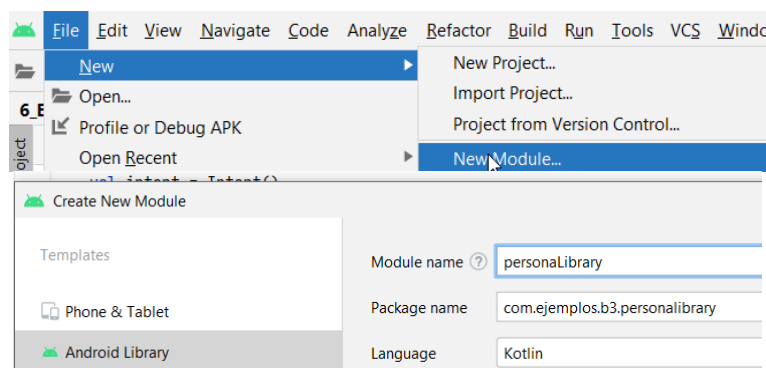
```

Usaremos un Intent al que le añadiremos el Bundle con el objeto parcelado. Al pulsar el botón **Mandar**, se deberá iniciar una aplicación diferente, **Parcelable2Activity**, que mostrará los datos que habremos metido en la aplicación anterior (puedes anular el método ToString en la clase Persona y mostrar la información con un TextView en la segunda actividad).

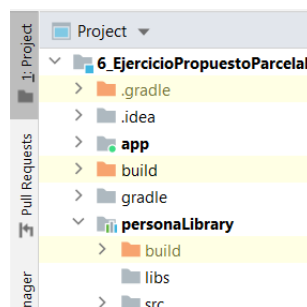
⚠ El único problema que nos encontraremos en este ejercicio, es que no nos va a permitir recuperar el objeto parcelado si no es exactamente en mismo que el de la clase de donde lo hemos parcelado (no vale con copiarlo y pegarlo). Para ello deberemos crear un módulo de librería **.arr**, con la clase **Persona** en la App **PropuestoParcelable** y tendremos que importar ese módulo a la clase **Parcelables2Activity**. La información con los pasos a seguir se puede ver en el siguiente [enlace](#), aunque se explica también más abajo.

Como crear un módulo de biblioteca

Para crear las clases que queremos que se incluyan en una biblioteca, se debe de crear un nuevo módulo y añadir todo el código en este. Para ello tendremos:

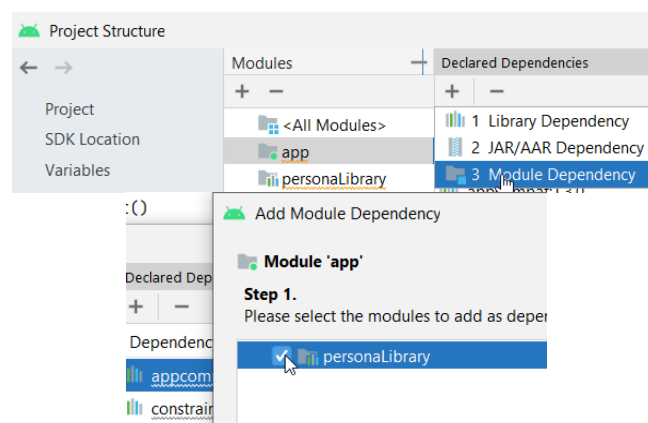


El resultado desde la vista proyecto mostrará los dos módulos, el del app y el de la biblioteca:

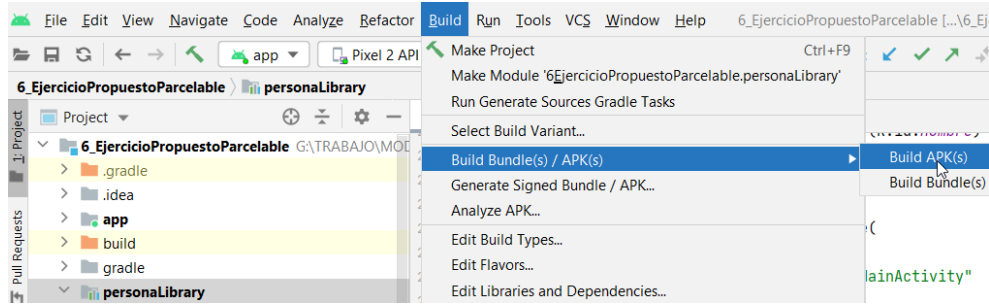


Persona debe estar en la biblioteca **personaLibrary**, y no en la app.

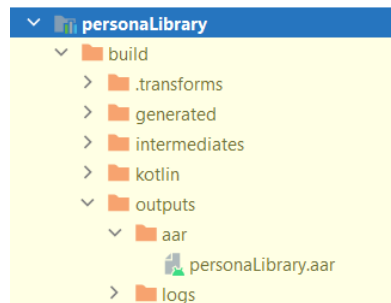
Para poder usar la biblioteca dentro del mismo proyecto, solamente tendremos que agregar una dependencia del módulo de la aplicación al módulo de biblioteca:



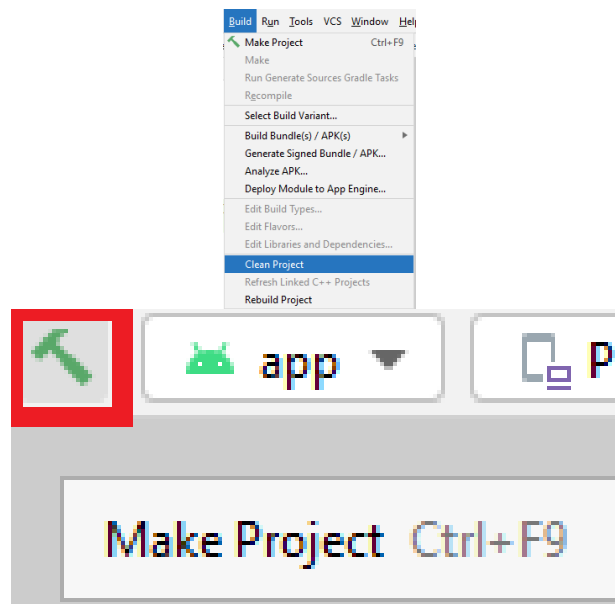
Para que se cree el archivo que luego exportaremos **.aar** , habrá que construir el módulo de biblioteca:



De esta forma se generará el archivo:



A veces no se genera el archivo y debemos forzarlo mediante, **Clean Project** y **Rebuild Project**, o mediante **Make Project**



Este archivo no es necesario para este proyecto, sino para el segundo proyecto, en el que debemos añadir esta biblioteca. Esto se explica en el siguiente punto.

Para lanzar la segunda aplicación con el parcelabre de Persona, podemos realizar el siguiente código en la MainActivity:

```

class MainActivity : AppCompatActivity(),View.OnClickListener {
    override fun onCreate(savedInstanceState: Bundle?) {
        super.onCreate(savedInstanceState)
        setContentView(R.layout.activity_main)
        findViewById<View>(R.id.mandar).setOnClickListener(this)
    }

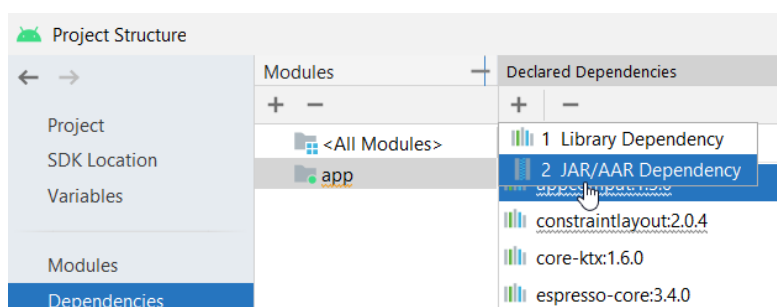
    override fun onClick(v: View) {
        if (v.getId() === R.id.mandar) {
            val nombre = (findViewById<View>(R.id.nombre) as EditText).text.toString()
            val dni = (findViewById<View>(R.id.dni) as EditText).text.toString()
            val edad = (findViewById<View>(R.id.edad) as EditText).text.toString().toInt()
            val intent = Intent()
            intent.component = ComponentName(
                "com.ejemplos.b3.activity2",
                "com.ejemplos.b3.activity2.MainActivity"
            )
            if (packageManager.resolveActivity(intent, 0) != null) {
                intent.putExtra("OBJETO", Persona(nombre, dni, edad))
                startActivity(intent)
            } else Toast.makeText(
                this,
                "No está instalada la aplicación en el Sistema",
                Toast.LENGTH_SHORT
            ).show()
        }
    }
}

```

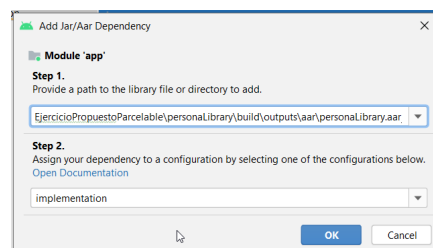
Como añadir una biblioteca a tu proyecto

Para añadir una biblioteca externa **jar o aar** a nuestro proyecto, tendremos que agregar la dependencia a esta como hacemos normalmente, pero esta vez seleccionando el tipo

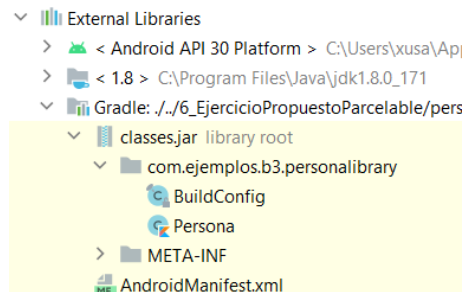
JAR/AAR Dependency



Indicaremos la ruta donde está guardado el archivo **.aar**, confirmaremos, y ya estará añadida.



Se podrá usar normalmente teniendo en cuenta importar correctamente el dominio. Si queremos comprobar que se ha añadido correctamente, se puede ver en **External Libraries** :



En la segunda aplicación podemos mostrar los datos de la manera que queramos, una vez añadida la librería y el import del dominio. En el siguiente código se han mostrado como cadena en el TextView del hola mundo.

```
import com.ejemplos.b3.personallibrary.Persona

class MainActivity : AppCompatActivity() {
    override fun onCreate(savedInstanceState: Bundle?) {
        super.onCreate(savedInstanceState)
        setContentView(R.layout.activity_main)
        val intent = intent

        if (intent.getParcelableExtra<Persona>("OBJETO") != null) (findViewById<TextView>(R.i
            intent.getParcelableExtra<Persona>("OBJETO").toString()
        }
    }
}
```