

Bloque 3. Exercici Proposat Parcelable

Descarregar aquests apunts [pdf](./EjercicioPropuestoParcelable_val.pdf) o [html](#)

Emmagatzematge d'informació en Android

Exercici que ens servirà per a practicar la parcel·lació d'unes dades amb un Intent, i també la transmissió d'aqueixes dades a una APP diferent on seran rebuts i mostrats.

Crea una aplicació **PropuestoParcelable**, amb el següent aspecte.



The mockup shows a mobile application interface with a purple header bar containing the text "6_EjercicioPropuestoParcelable". Below the header, there is a form with three input fields: "Introduce DNI" (with a small icon to its left), "Nombre", and "Edad". A purple button labeled "MANDAR" is positioned below the "Nombre" and "Edad" fields.

Si no tens clar el codi a crear, pots usar el següent arxiu 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>

```

Haurà de tindre una classe Pojo anomenada **Persona** que estendrà de Parcelable, tindrà dos tipus String: dni i nom i un sencer edat. El còdic podria 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)
        }
    }
}

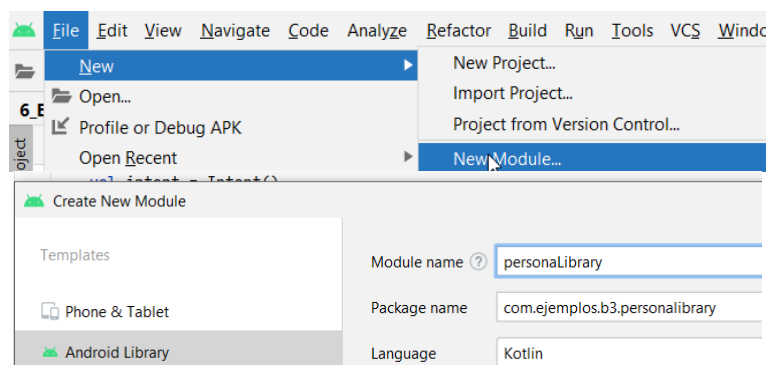
```

Usarem un Intent al qual li afegirem el Bundle amb l'objecte parcel·lat. En prémer el botó **Mandar**, s'haurà d'iniciar una aplicació diferent, **Parcelable2Activity**, que mostrarà les dades que haurem ficat en l'aplicació anterior (pots anul·lar el mètode ToString en la classe Persona i mostrar la informació amb un TextView en la segona activitat).

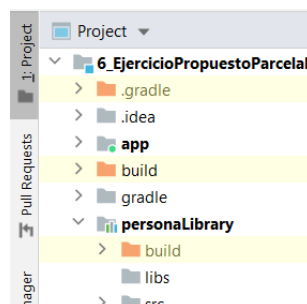
⚠️ Avís: L'únic problema que ens trobarem en aquest exercici, és que no ens permetrà recuperar l'objecte parcel·lat si no pertany exactament a la mateixa classe d'on l'hem parcel·lat. Per a això haurem de crear un mòdul de llibreria **.arr**, amb la classe Persona en l'App PropuestoParcelable i haurem d'importar aqueix mòdul a la classe Parcelables2Activity. La informació amb els passos a seguir es pot veure en el següent [enllaç](#), encara que s'explica també més a baix.

Com crear un mòdul de biblioteca

Per a crear les classes que volem que s'incloguen en una biblioteca, s'ha de crear un nou mòdul i afegir tot el codi en aquest. Per a això tindrem:

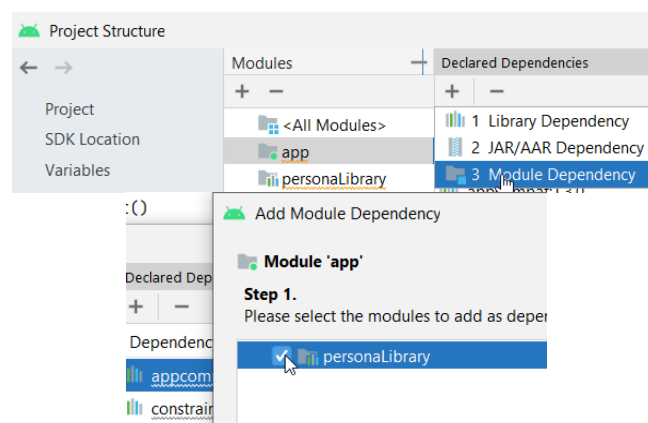


El resultat des de la vista projecte mostrarà els dos mòduls, el de l'app i el de la biblioteca:

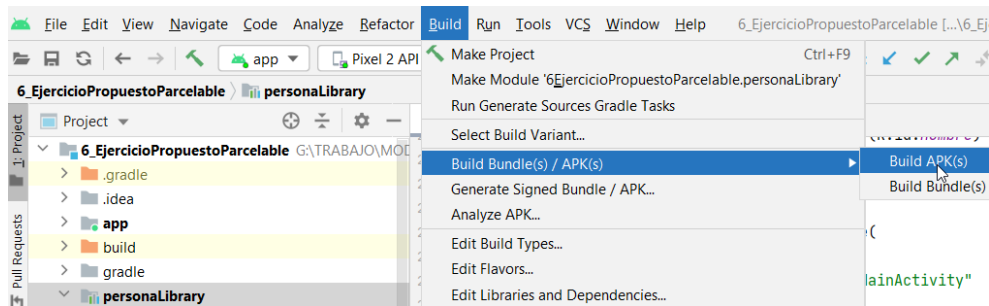


Persona ha d'estar a la biblioteca personaLibrary, i no en l'app.

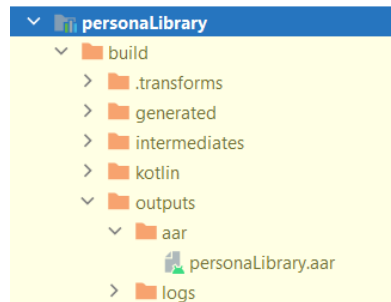
Per a poder usar la biblioteca dins del mateix projecte, solament haurem d'agregar una dependència del mòdul de l'aplicació al mòdul de biblioteca:



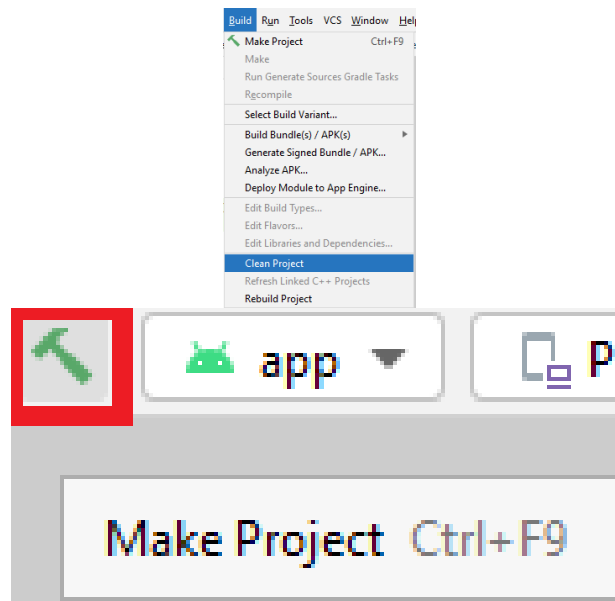
Perquè es cree l'arxiu que després exportarem **.aar** , caldrà construir el mòdul de biblioteca:



D'aquesta manera es generarà l'arxiu:



A vegades no es genera l'arxiu i hem de forçar-lo mitjançant, **Clean Project** i **Rebuild Project**, o mitjançant **Make roject*



Aquest arxiu no és necessari per a aquest projecte, sinó per al segon projecte, en el qual hem d'afegir aquest biblioteca. Això s'explica en el següent punt.

Per a llançar la segona aplicació amb el parcelabre de Persona, podem realitzar el següent codi en la MainActivity:"

```

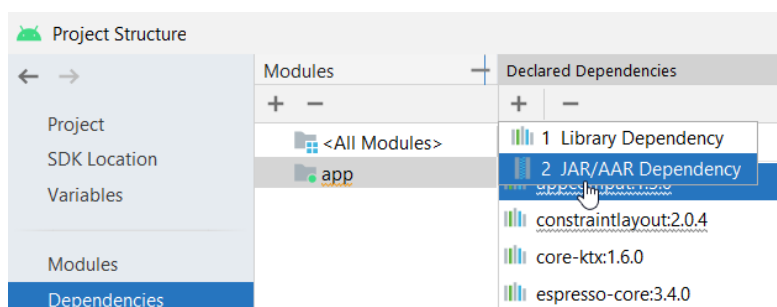
class MainActivity : AppCompatActivity(),View.OnClickListener {
    override fun onCreate(savedInstanceState: Bundle?) {
        super.onCreate(savedInstanceState)
        setContentView(R.layout.activity_main)
        findViewById<View>(R.id.mandar).setOnClickListener(this)
    }

    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()
        }
    }
}

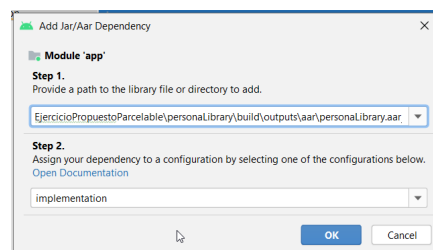
```

Com afegir una biblioteca al teu projecte

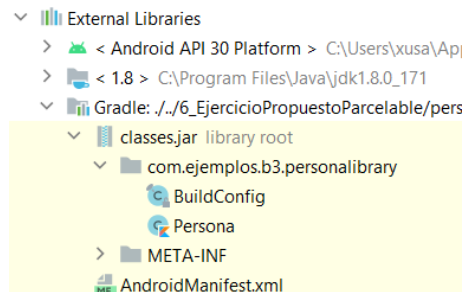
Per a afegir una biblioteca externa **jar** o **aar** al nostre projecte, haurem d'agregar la dependència a aquesta com fem normalment, però aquesta vegada seleccionant el tipus **JAR/AAR Dependency**



Indicarem la ruta on està guardat l'arxiu **.aar**, confirmarem, i ja estarà afegida.



Es podrà usar normalment tenint en compte importar correctament el domini. Si volem comprovar que s'ha afegit correctament, es pot veure en **External Libraries** :



En la segona aplicació podem mostrar les dades de la manera que vullguem, una vegada afegida la llibreria i l'import del domini. En el següent codi s'han mostrat com a cadena en el TextView de l'hola món.

```
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()
        }
    }
}
```