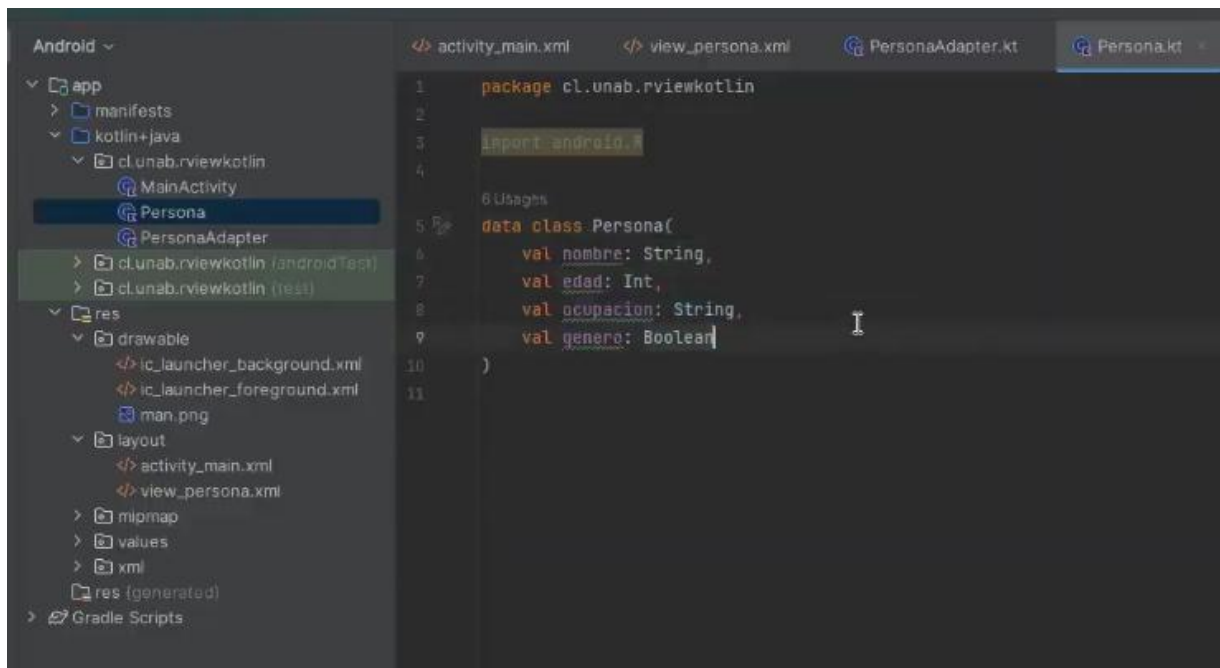
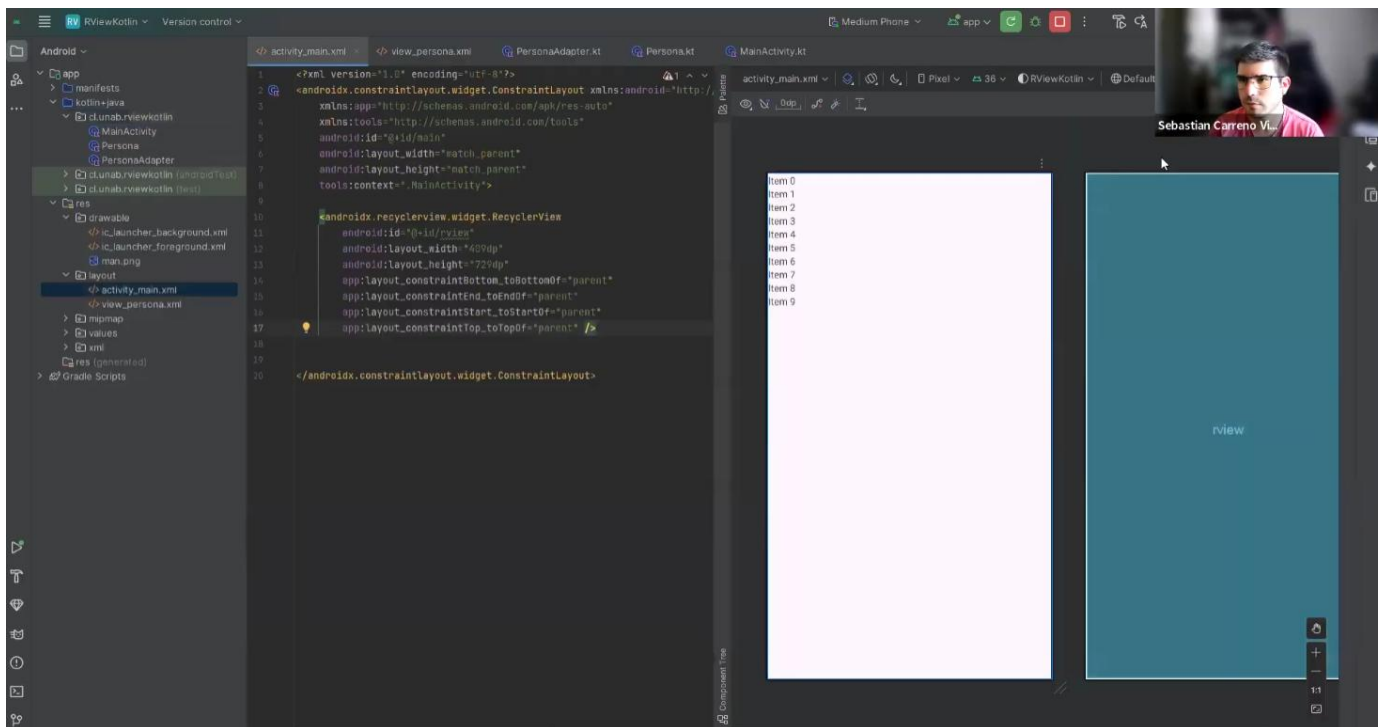


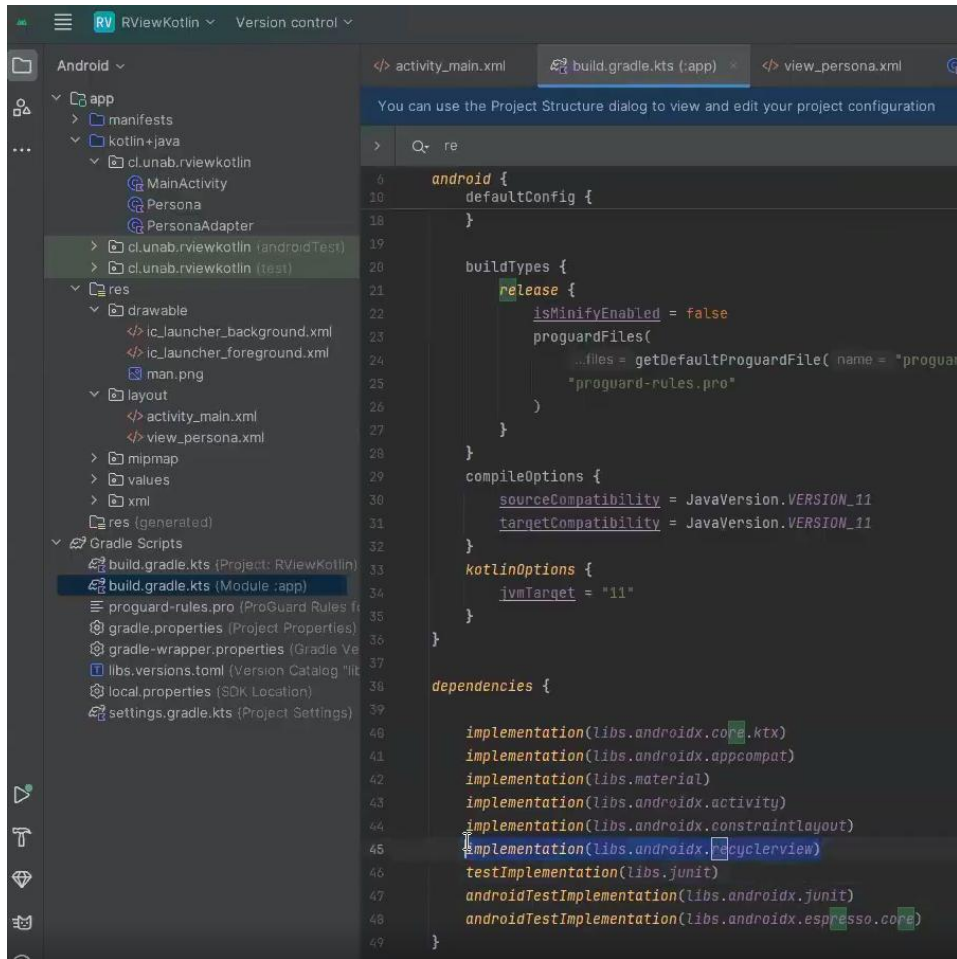
Ejemplo de creación de adaptadores, crear clase Data Class



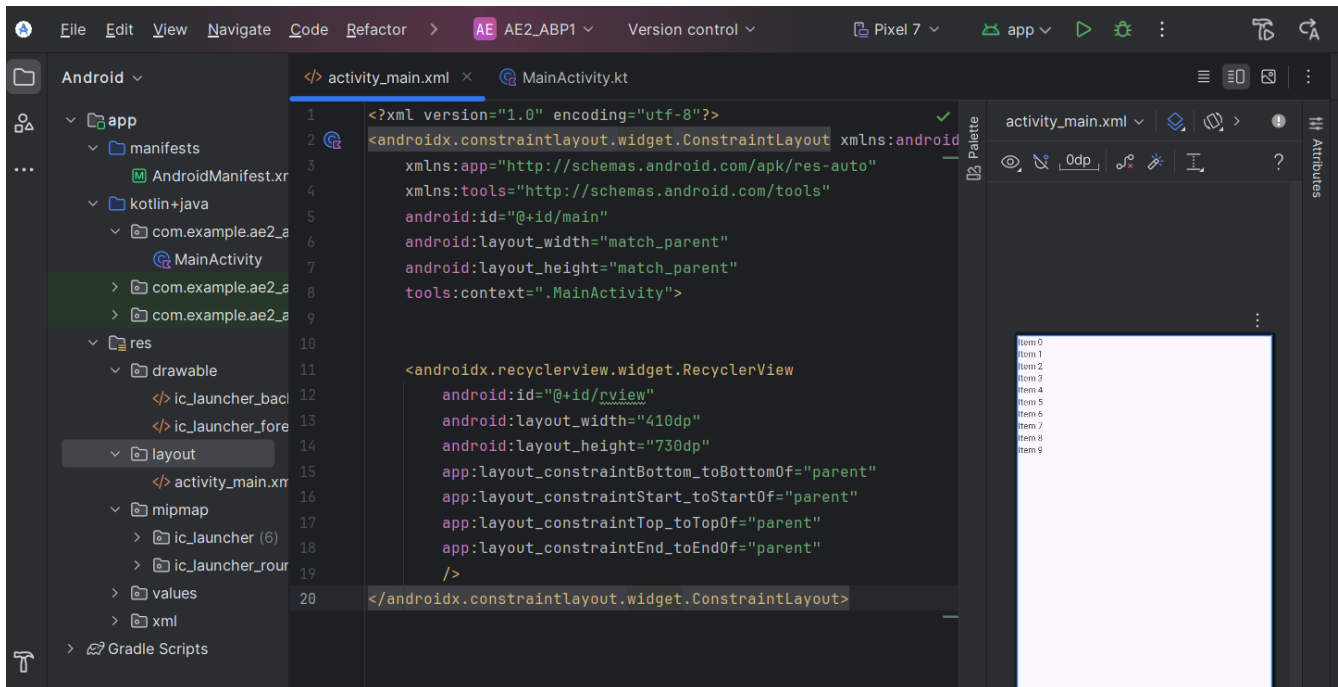
En el Main, crear un recycler view



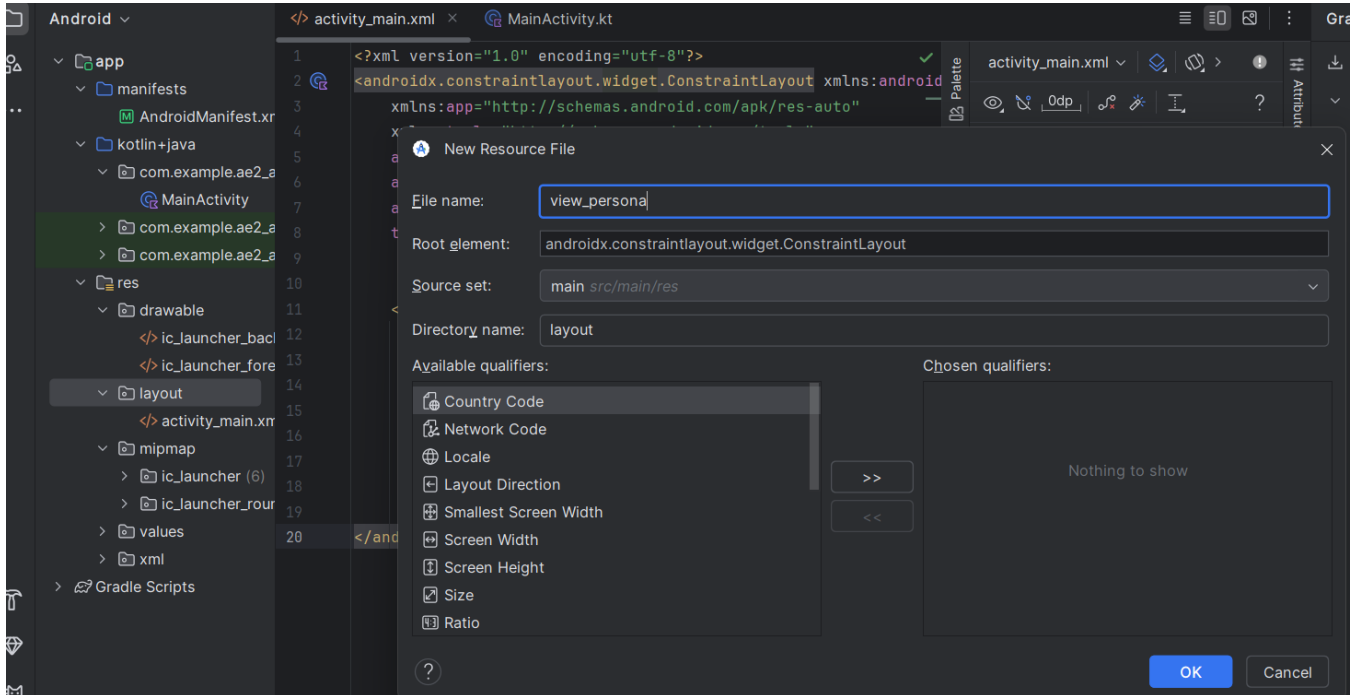
Se crea automáticamente en gradle app



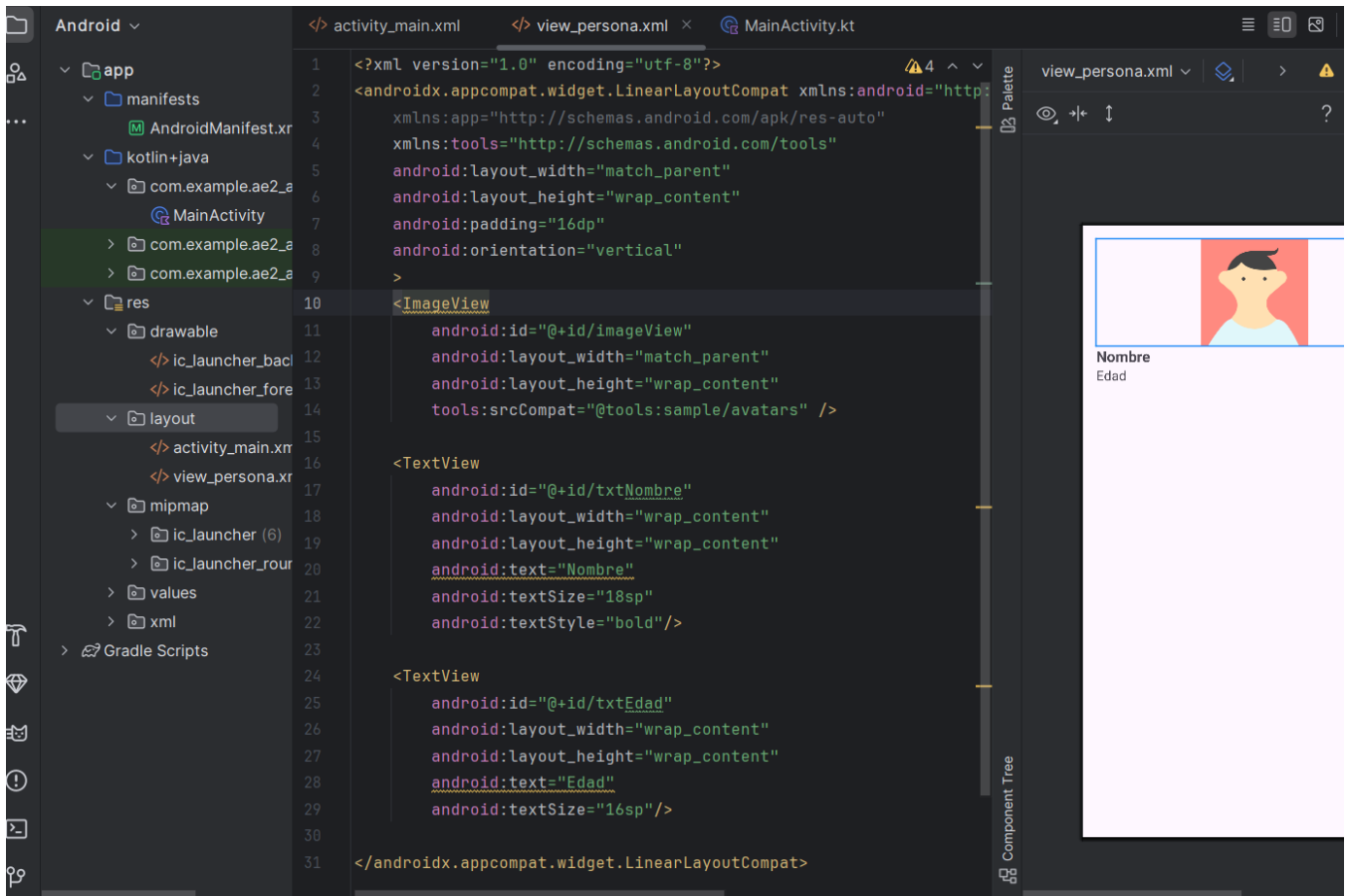
Se agregar los constraints y id



Se crea una nueva vista, que se llama view_persona



Se completa con una imagen y dos textview



Se crea personaAdapter

Parte de la clase principal.
Los 3 overrides son obligatorios o arrojará error el RecyclerView

Sin esta biblioteca, no resultará el RecyclerView

Lo primero es darle el parámetro de entrada. Le daremos una lista, una lista de personas

Y como adaptador va a devolver, un RecyclerView.Adapter de PersonaAdapter

Cada elemento que quiera que se interprete, debe ir aquí (en este caso: imgPersona, txtNombre, txtEdad) que va en la view_persona

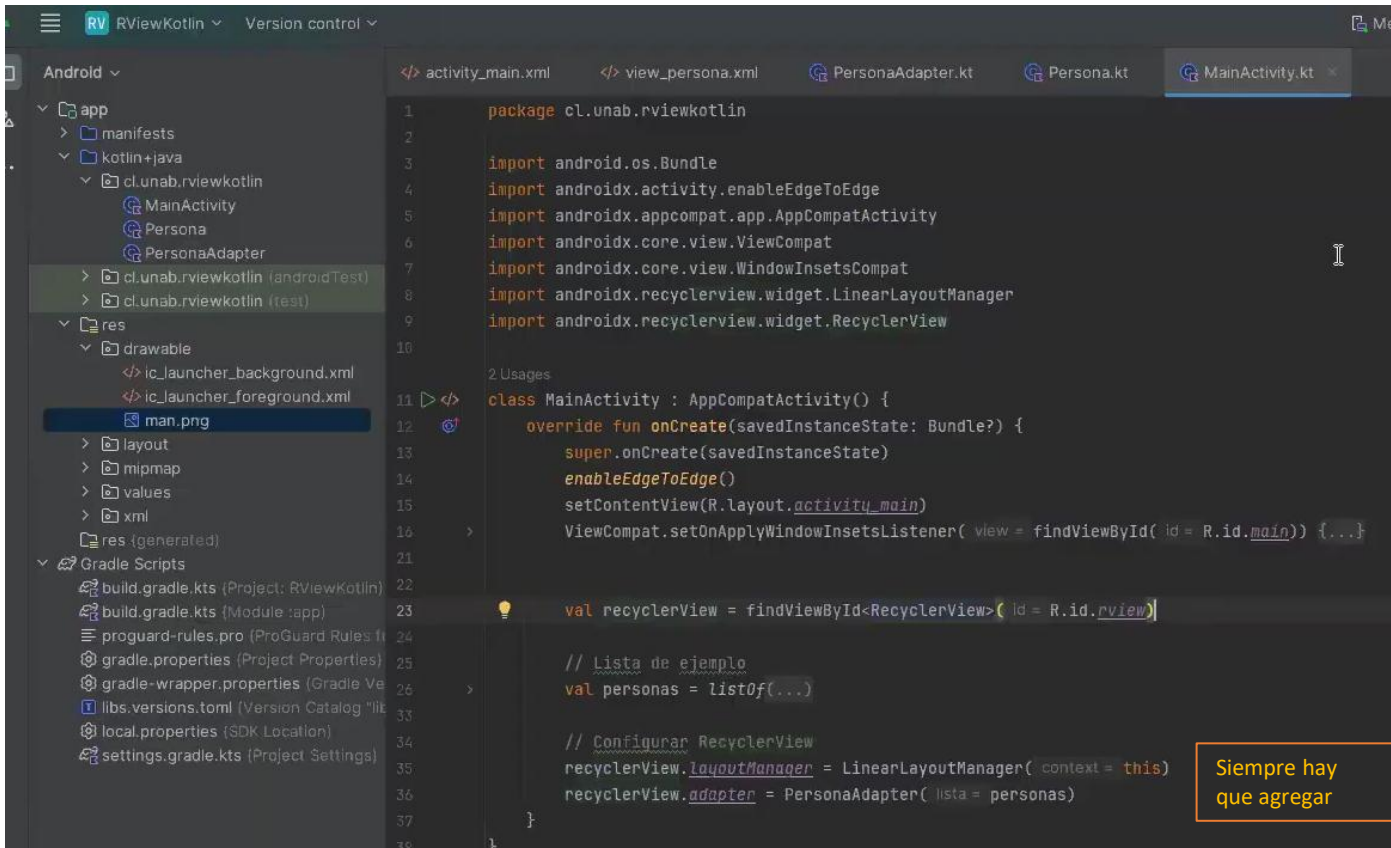
El ViewHolder, es como una caja que guarda las vistas de cada fila

Engancha los elementos a la vista que quiero utilizar.

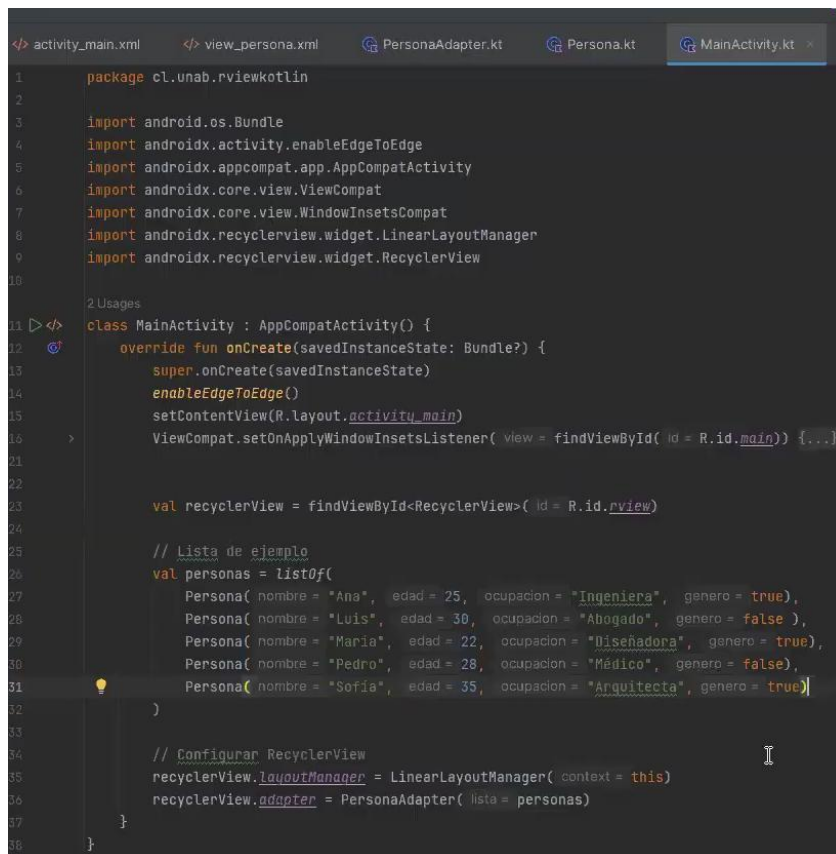
onBindViewHolder es como un for, es el ciclo, va a recorrer la vista, casi como un for each

```
1 package cl.unab.rviewkotlin
2
3 import android.view.LayoutInflater
4 import android.view.View
5 import android.view.ViewGroup
6 import android.widget.ImageView
7 import android.widget.TextView
8 import androidx.recyclerview.widget.RecyclerView
9
10 //conector entre tus datos y el RecyclerView.
11 class PersonaAdapter(private val lista: List<Persona>) :
12     RecyclerView.Adapter<PersonaAdapter.PersonaViewHolder>() {
13
14     //caja que guarda las vistas de cada fila.
15     class PersonaViewHolder(itemView: View) : RecyclerView.ViewHolder(itemView) {
16         1 Usage
17         val imgPersona: ImageView = itemView.findViewById<ImageView>(R.id.imgPersona)
18         1 Usage
19         val txtNombre: TextView = itemView.findViewById<TextView>(R.id.txtNombre)
20         1 Usage
21         val txtEdad: TextView = itemView.findViewById<TextView>(R.id.txtEdad)
22     }
23
24     //crea la vista de un item.
25     override fun onCreateViewHolder(parent: ViewGroup, viewType: Int): PersonaViewHolder {
26         val view = LayoutInflater.from(context).inflate(R.layout.view_persona, parent, false)
27         return PersonaViewHolder(itemView = view)
28     }
29
30     //llena la vista con datos.
31     override fun onBindViewHolder(holder: PersonaViewHolder, position: Int) {
32         val persona = lista[position]
33         holder.imgPersona.setImageResource(R.drawable.man)
34         holder.txtNombre.text = persona.nombre
35         holder.txtEdad.text = "Edad: ${persona.edad}"
36
37     //obligatorio debido a la abstraccion y nos permite ver el tamaño de la lista.
38     override fun getItemCount(): Int = lista.size
39 }
```

Para implementar en el Main:



Lista de ejemplo utilizada:



Resultado: listado hacia abajo

