Android Studio

" RecyclerView, what is it? "

INNEHÅLLSFÖRTECKNING

01

ÖVERSIKT

Recycler & List

Recycler

Tillämpa lösningar för större listor.

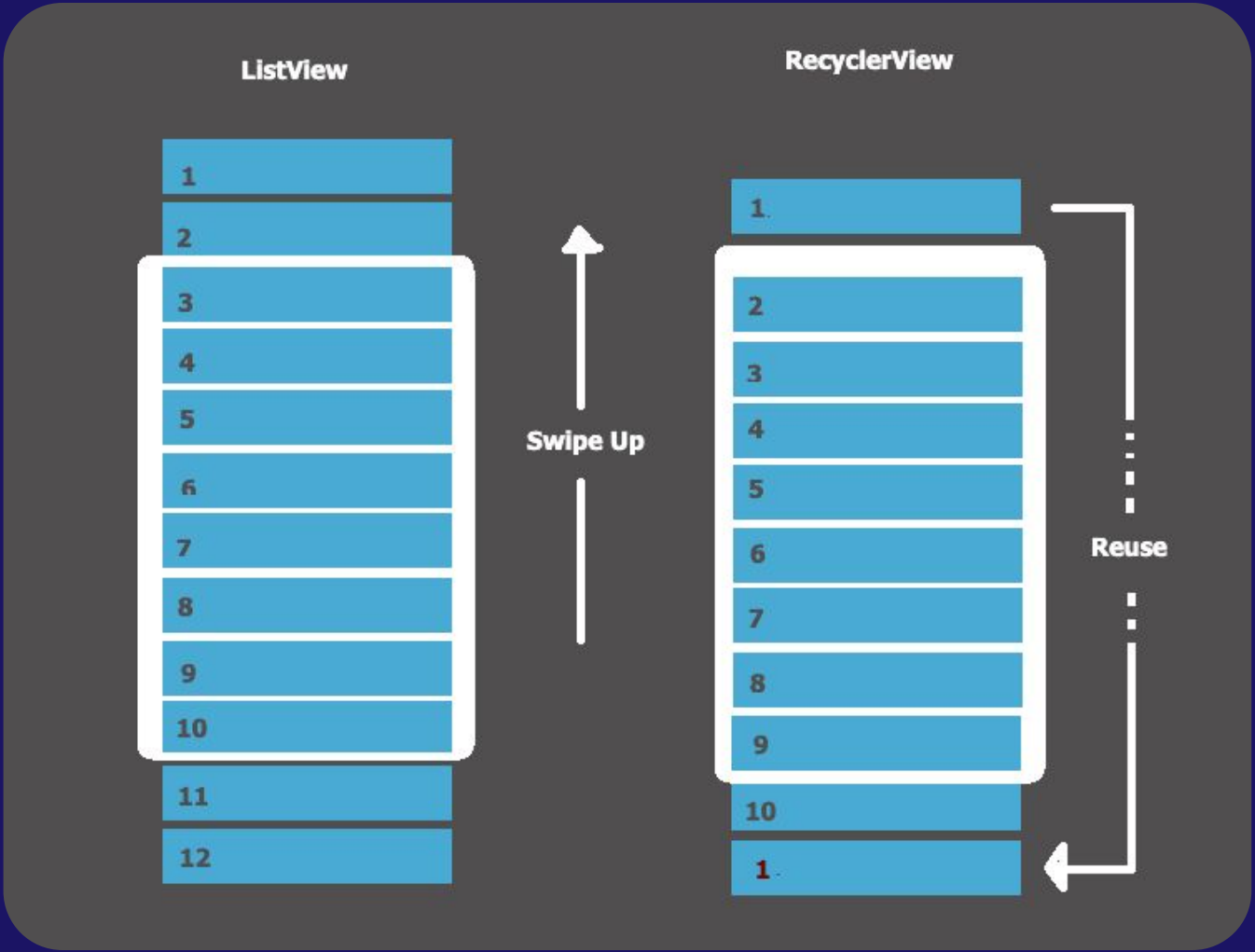
- List för mindre items (enklare uppsättning)
- RecyclerView för större listor (performant friendly)
- Compose alternative

Frågor?

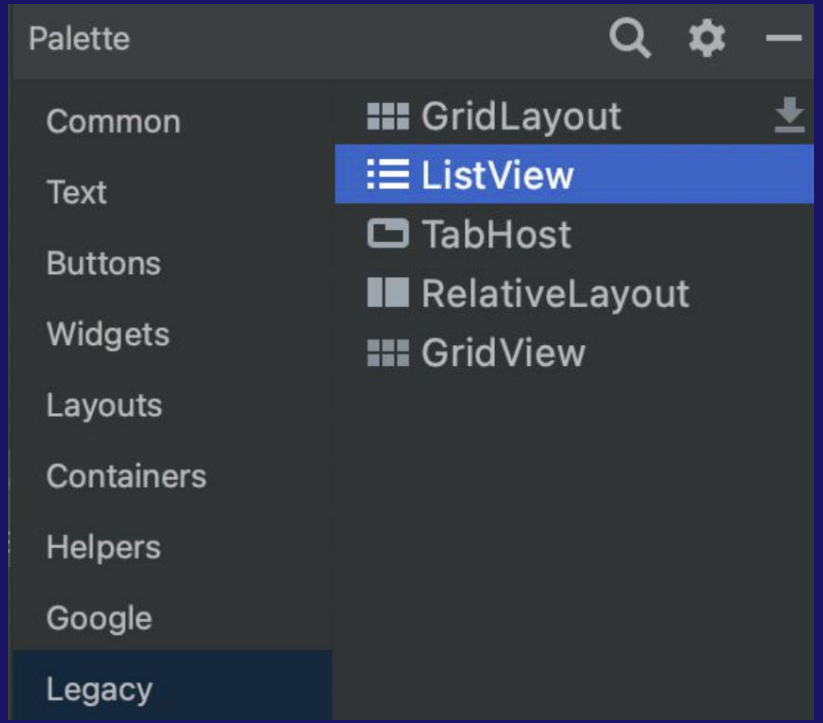
02

RecyclerView

What is a RecyclerView?


ListView
RecyclerView
1
2
3
4
5
6
7
8
9
10
11
12
Swipe Up
1
2
3
4
5
6
7
8
9
10
1
Reuse

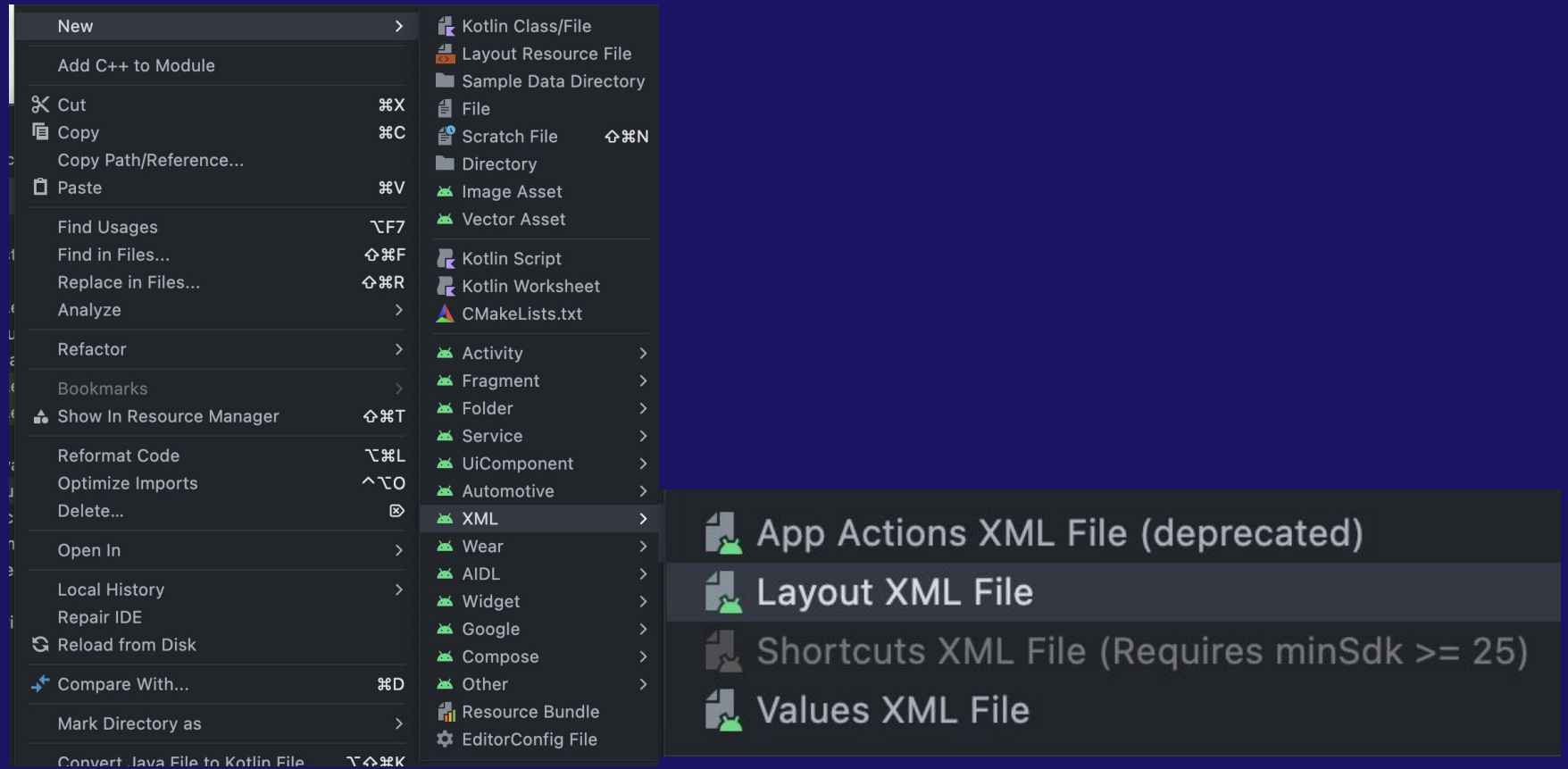
ListView?



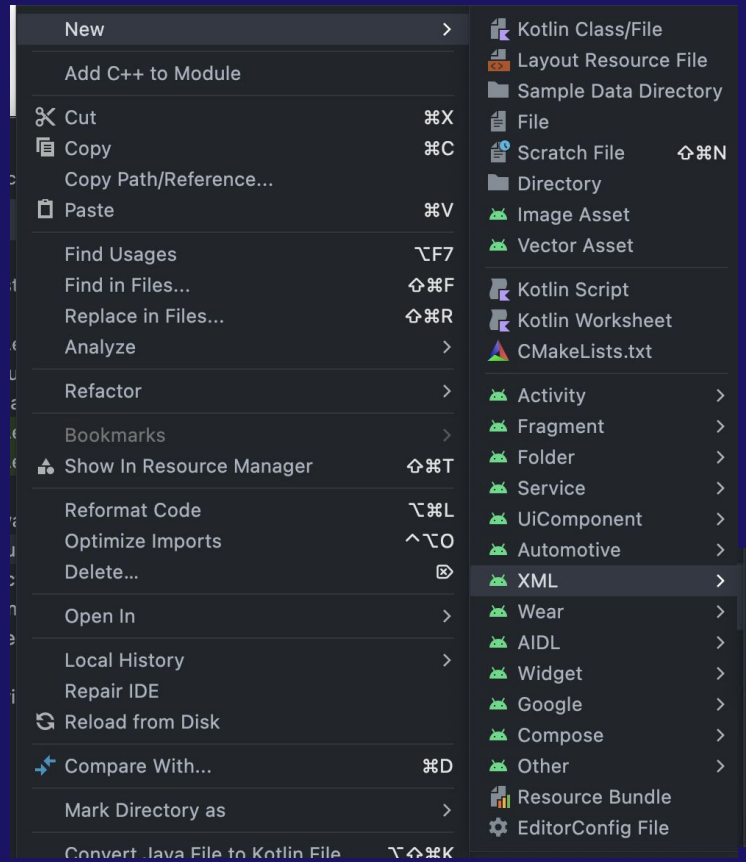
Numera anses ListView vara en del av ‘Legacy Code’ och används inte så mycket längre…

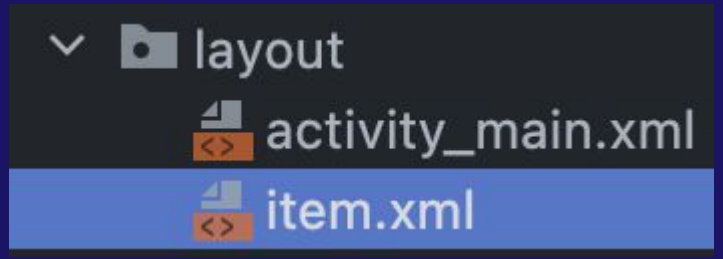
How to: RecyclerView

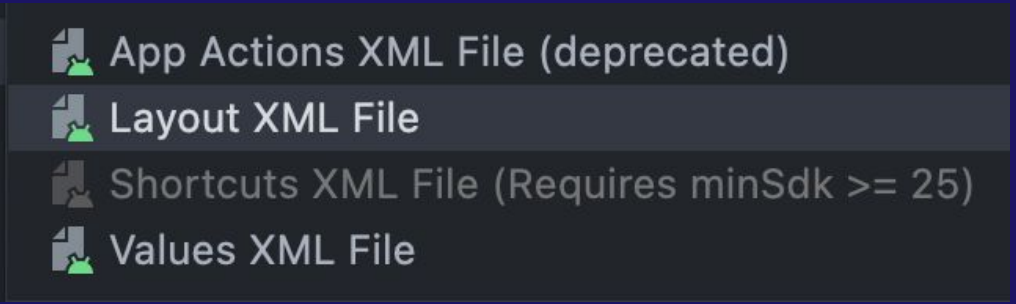
The classes


New
Add C++ to Module
Cut ⌘X
Copy ⌘C
Copy Path/Reference...
Paste ⌘V
Find Usages ⌥F7
Find in Files... ⇧⌘F
Replace in Files... ⇧⌘R
Analyze
Refactor
Bookmarks
Show In Resource Manager ⇧⌘T
Reformat Code ⌥⌘L
Optimize Imports ^⌥O
Delete...
Open In
Local History
Repair IDE
Reload from Disk
Compare With... ⌘D
Mark Directory as
Convert Java File to Kotlin File ⌥⇧⌘K
Kotlin Class/File
Layout Resource File
Sample Data Directory
File
Scratch File ⇧⌘N
Directory
Image Asset
Vector Asset
Kotlin Script
Kotlin Worksheet
CMakeLists.txt
Activity
Fragment
Folder
Service
UiComponent
Automotive
XML
Wear
AIDL
Widget
Google
Compose
Other
Resource Bundle
EditorConfig File
App Actions XML File (deprecated)
Layout XML File
Shortcuts XML File (Requires minSdk >= 25)
Values XML File

The classes


New
Add C++ to Module
Cut ⌘X
Copy ⌘C
Copy Path/Reference...
Paste ⌘V
Find Usages ⌥F7
Find in Files... ⇧⌘F
Replace in Files... ⇧⌘R
Analyze
Refactor
Bookmarks
Show In Resource Manager ⇧⌘T
Reformat Code ⌥⌘L
Optimize Imports ^⌥O
Delete...
Open In
Local History
Repair IDE
Reload from Disk
Compare With... ⌘D
Mark Directory as
Convert Java File to Kotlin File ⌥⇧⌘K
Kotlin Class/File
Layout Resource File
Sample Data Directory
File
Scratch File ⇧⌘N
Directory
Image Asset
Vector Asset
Kotlin Script
Kotlin Worksheet
CMakeLists.txt
Activity
Fragment
Folder
Service
UiComponent
Automotive
XML
Wear
AIDL
Widget
Google
Compose
Other
Resource Bundle
EditorConfig File


layout
activity_main.xml
item.xml


App Actions XML File (deprecated)
Layout XML File
Shortcuts XML File (Requires minSdk >= 25)
Values XML File

item.xml

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
   android:layout width="match parent"
   android:layout_height="match_parent">

   <TextView
       android:id="@+id/itemTextView"
       android:layout width="wrap content"
       android:layout height="wrap_content"
       android:text="SomeItem"

       android:textSize="22sp" />

</LinearLayout>
```

NOTERA: TextView har ett id “tv_item”

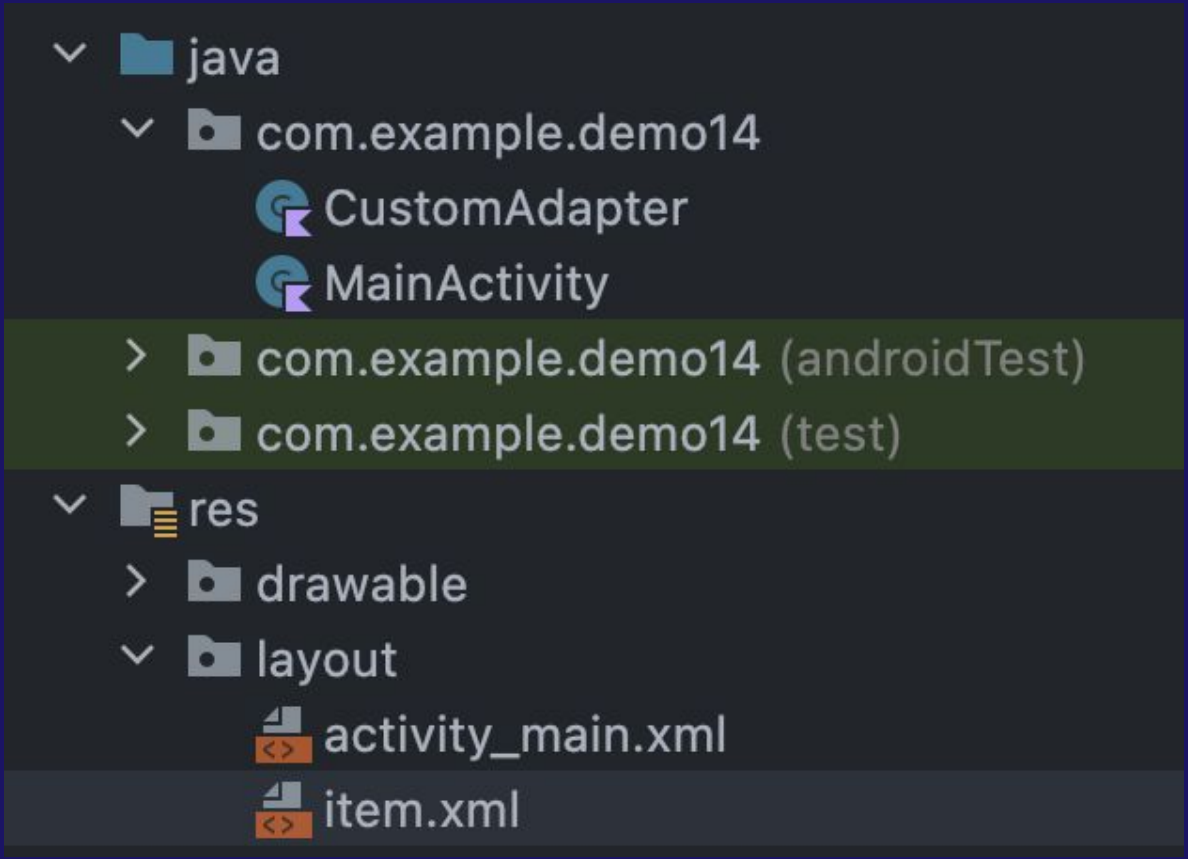
VARNING: Ändra match_parent till ‘wrap_content’ på LinearLayout!

item.xml

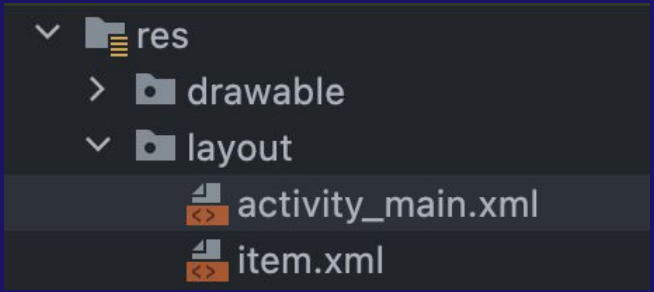
```xml
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
   android:layout_width="match_parent"
   android:layout_height="wrap_content"
   android:layout_marginBottom="10dp"
   android:clickable="true"
   android:focusable="true"
   android:background="?attr/selectableItemBackground"
   >


      <TextView
         android:id="@+id/tv_score"
         android:layout_width="250dp"
         android:layout_height="wrap_content"
         android:text=" Default Score "
         android:textSize="24sp"

         android:layout_marginStart="25dp"
         />

      <LinearLayout
         android:layout_width="match_parent"
         android:layout_height="match_parent"
         android:gravity="end"
         >

            <ImageView
               android:layout_width="wrap_content"
               android:layout_height="wrap_content"
               android:adjustViewBounds="true"
               android:maxWidth="45dp"
               android:maxHeight="45dp"
               android:scaleType="fitCenter"
               android:layout_marginEnd="25dp"

               android:layout_gravity="center"
               android:src="@drawable/baseline_cancel_24"

               />

      </LinearLayout>
</LinearLayout>
```

Copy paste me!

Structure


java
com.example.demo14
CustomAdapter
MainActivity
com.example.demo14 (androidTest)
com.example.demo14 (test)
res
drawable
layout
activity_main.xml
item.xml

Structure



ID = tv_test

```xml
<?xml version="1.0" encoding="utf-8"?>
<androidx.constraintlayout.widget.ConstraintLayout
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"
   tools:context=".MainActivity">

   <androidx.recyclerview.widget.RecyclerView
       android:id="@+id/rv_test"
       android:layout_width="409dp"
       android:layout_height="354dp"
       app:layout_constraintEnd_toEndOf="parent"
       app:layout_constraintStart_toStartOf="parent"
       app:layout_constraintTop_toTopOf="parent" />

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

Setup

MainActivity

```
class MainActivity : AppCompatActivity() {

    private lateinit var customAdapter: CustomAdapter
    private val itemsList = ArrayList<String>()

    override fun onCreate(savedInstanceState: Bundle?) {
        super.onCreate(savedInstanceState)
        setContentView(R.layout.activity_main)
    }
}
```

Adapter

CustomAdapter

```
package com.example.demo14

import android.view.View
import android.widget.TextView
import androidx.recyclerview.widget.RecyclerView

internal class CustomAdapter(private var itemsList: List<String>):
   RecyclerView.Adapter<CustomAdapter.MyViewHolder>()
{

   internal inner class MyViewHolder(view: View) : RecyclerView.ViewHolder(view)
{
       var itemTextView = view.findViewById<TextView>(R.id.tv_item)
   }

}
```

Scope

- `private` means that the member is visible inside this class only (including all its members).
- `protected` means that the member has the same visibility as one marked as `private`, but that it is also visible in subclasses.
- `internal` means that any client *inside this module* who sees the declaring class sees its `internal` members.
- `public` means that any client who sees the declaring class sees its `public` members.

In Kotlin, an outer class does not see private members of its inner classes.

Source: https://kotlinlang.org/docs/visibility-modifiers.html#class-members

Scope

- `private` means that the member is visible inside this class only (including all its members).
- `protected` means that the member has the same visibility as one marked as `private`, but that it is also visible in subclasses.
- `internal` means that any client *inside this module* who sees the declaring class sees its `internal` members.

A *module* in kotlin is a set of Kotlin files compiled together. *modules* can be: maven projects, gradle sets, files generated from an Ant task, or a IntelliJ IDEA module

Source: https://kotlinlang.org/docs/visibility-modifiers.html#class-members

Inheritance

CustomAdapter

Class 'CustomAdapter' is not abstract and does not implement abstract base class member
public abstract fun onCreateViewHolder(parent: ViewGroup, viewType: Int): CustomAdapter.MyViewHolder *defined in* androidx.recycle
widget.RecyclerView.Adapter

Implement members ⌥⇧⏎ More actions... ⌥⏎

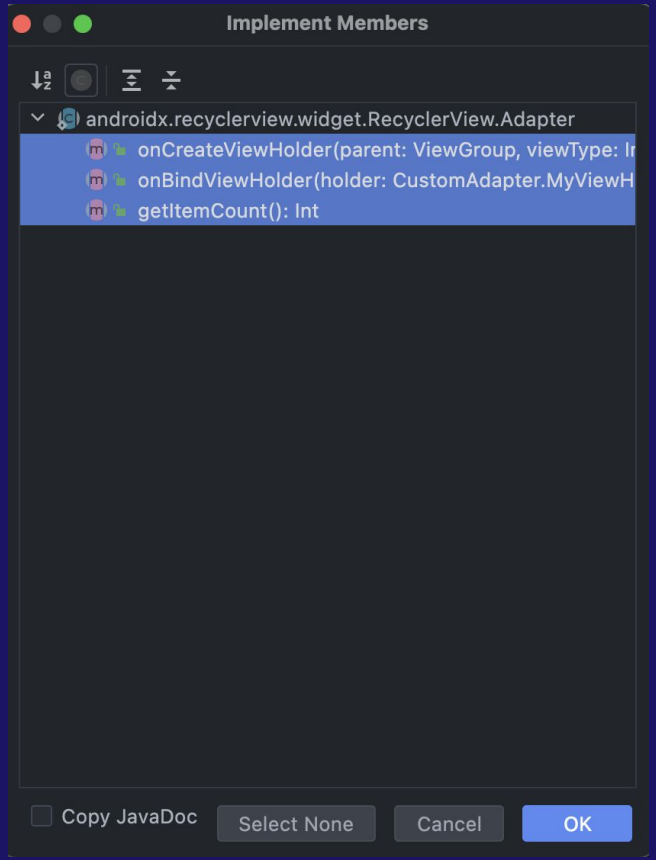
```
internal final class CustomAdapter
    : RecyclerView.Adapter<CustomAdapter.MyViewHolder>
```

com.example.demo14
CustomAdapter.kt

demo14.app.main

```
        var itemTextView = view.findViewById<TextView>(R.id.tv_item)
    }

}
```

Override!



Markera ALLT - tryck OK

Override

```
override fun onCreateViewHolder(parent: ViewGroup, viewType: Int): MyViewHolder {
    TODO("Not yet implemented")
}

override fun getItemCount(): Int {
    TODO("Not yet implemented")
}

override fun onBindViewHolder(holder: MyViewHolder, position: Int) {
    TODO("Not yet implemented")
}
```

Override

```
override fun onCreateViewHolder(parent: ViewGroup, viewType: Int): MyViewHolder {
    val itemView = LayoutInflater.from(parent.context)
        .inflate(R.layout.item, parent, false)

    return MyViewHolder(itemView)
}

override fun getItemCount(): Int {
    return itemsList.size
}

override fun onBindViewHolder(holder: MyViewHolder, position: Int) {
    val item = itemsList[position]
    holder.itemTextView.text = item
}
```

ActivityMain

MainActivity

```
// Setup RecyclerView
val recyclerView: RecyclerView = findViewById(R.id.rv_test)
customAdapter = CustomAdapter(itemsList)
val layoutManager = LinearLayoutManager(applicationContext)
recyclerView.layoutManager = layoutManager
recyclerView.adapter = customAdapter

prepareItems()  // We define this on the next slide
```

Override

```
private fun prepareItems() {
    for (i in 1..1000) {
        itemsList.add("Item $i")
    }
      customAdapter.notifyItemRangeInserted(1, 1000)
}
```

Det är viktigt att när ni är klara med att lägga till items, att notifiera förändringarna.

Gör vi inte det, så kan konstiga saker hända och det finns en risk att listan inte uppdateras korrekt.

OnClick!

CustomAdapter

```
var onItemClick: ((String) -> Unit)? = null      // BoilerPLate

internal inner class MyViewHolder(view: View) :
RecyclerView.ViewHolder(view) {
    var itemTextView: TextView = view.findViewById(R.id.tv_score)
    // Have more TextViews? Define them here!

    // Define OnCLick
    init {
        view.setOnClickListener {
            onItemClick?.invoke(itemsList[adapterPosition])
        }
    }

}
```

OnClick!

Main Activity

```
// onItemClick
adapter.onItemClick = {
   Toast.makeText(applicationContext, "Score was: $it",
Toast.LENGTH_LONG).show()
}
```

03

Compose Alternative

Lazy!

That’s all…

```
@Composable
fun MyScreen() {
    val itemsArray = arrayOf("Item 1", "Item 2", "Item 3", "Item 4", "Item 5")
    val selectedItemIndexState = remember { mutableStateOf(0) }

    LazyColumn {
        items(itemsArray) { item ->
            Text(
                text = item,
                modifier = Modifier
                    .fillMaxWidth()
                    .clickable {
                        selectedItemIndexState.value = itemsArray.indexOf(item)
                        println("Item $selectedItemIndexState")
                    }
                    .padding(16.dp)
            )
        }
    }
}
```

04

Övningar & Uppgifter

MINNS DU?

```
// Vad är en ListView?
// Hur jämför sig en RecyclerView mot
en ListView?
Vilket är att föredra?

// Hur fungerar en RecyclerView?
```

THANKS !

Do you have any questions?
kristoffer.johansson@sti.se

sti.learning.nu/

You can also contact me VIA Teams (quicker response)
Du kan också kontakta mig VIA Teams (Snabbare svar)