







The screenshot shows the Android Studio interface with the layout editor open. In the center, there is a circular progress bar with a blue outline and a light gray background. The XML code for the layout is displayed on the left, and the attributes and constraints for the circular progress bar are shown on the right.

viewholder_budget.xml

```
<?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">
    <com.mikhaelopez.circularprogressbar.CircularProgressBar
        android:id="@+id/circularProgressBar"
        android:layout_width="175dp"
        android:layout_height="175dp"

        android:layout_marginTop="16dp"
        app:cpb_background_progressbar_width="15dp"
        app:cpb_progress_index="100"
        app:cpb_progressbar_color="@color/blue"
        app:cpb_progressbar_width="15dp"
        app:cpb_round_border="true"
        app:cpb_background_progressbar_color="@color/lightGrey"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent" />
</androidx.constraintlayout.widget.ConstraintLayout>
```

Attributes

CircularProgressBar	circularProgressBar
id	circularProgressBar
Declared Attributes	
layout_width	175dp
layout_height	175dp
layout_marginTop	16dp
id	circularProgressBar
Layout	
Constraint Widget	

Component Tree

- ConstraintLayout
 - circularProgressBar

Constraints (0)

Transforms

The screenshot shows the Android Studio interface with the following details:

- Project Structure:** The left sidebar displays the project structure under the "Android" tab. It includes:
 - app:** Contains `manifests`, `kotlin+java`, `com.example.project1912` (which contains `Activity`, `Adapter`, `Domain`, `Repository`, and `ViewModel`), `res` (with `drawable`, `layout`, `values`, and `xml`), and `gradle Scripts` (with `build.gradle.kts`, `build.gradle.kts (Module: app)`, `proguard-rules.pro`, and `gradle.properties`).
 - `gradle.properties` is currently selected.
- Code Editor:** The main window shows the `ReportListAdapter.kt` file. The code is as follows:

```
1 package com.example.project1912.Adapter
2
3 import android.view.ViewGroup
4 import androidx.recyclerview.widget.RecyclerView
5 import com.example.project1912.databinding.ViewHolderBudgetBinding
6
7 class ReportListAdapter():RecyclerView.Adapter<ReportListAdapter.ViewHolder> {
8     class ViewHolder (val binding: ViewHolderBudgetBinding):RecyclerView.ViewHolder(binding)
9
10    override fun onCreateViewHolder(
11        parent: ViewGroup,
12        viewType: Int
13    ): ReportListAdapter.ViewHolder {
14        return ViewHolder(ViewHolderBudgetBinding.inflate(LayoutInflater.from(parent.context), parent, false))
15    }
16
17    override fun onBindViewHolder(holder: ReportListAdapter.ViewHolder, position: Int) {
18        holder.bind()
19    }
20
21    override fun getItemCount(): Int {
22        return 1000 // reason: "Not yet implemented"
23    }
24}
```

```
12  class ReportListAdapter(private val items: MutableList<BudgetDomain>) :  
13      RecyclerView.Adapter<ReportListAdapter.ViewHolder> {  
14          override fun onCreateViewHolder(parent: ViewGroup, viewType: Int): ViewHolder {  
15              context = parent.context  
16              formatter = DecimalFormat(pattern: "###,###,###,###")  
17              val binding = ViewholderBudgetBinding.inflate(LayoutInflater.from(context), parent, false)  
18              return ViewHolder(binding)  
19          }  
20          override fun onBindViewHolder(holder: ViewHolder, position: Int) {  
21              val item = items[position]  
22              holder.binding.titleTxt.text = item.title  
23              holder.binding.percentTxt.text = "% " + item.percent  
24              holder.binding.priceTxt.text = "$ " + formatter.format(item.price) + " /Month"  
25              holder.binding.circularProgressBar.apply {  
26                  progress = item.percent.toFloat()  
27                  if (position % 2 == 1) {  
28                      progressBarColor = context.resources.getColor(R.color.blue)  
29                      holder.binding.percentTxt.setTextColor(context.resources.getColor(R.color.black))  
30                  } else {  
31                      progressBarColor = context.resources.getColor(R.color.pink)  
32                      holder.binding.percentTxt.setTextColor(context.resources.getColor(R.color.white))  
33                  }  
34              }  
35          }  
36      }  
37  }  
38  
```

The screenshot shows the Android Studio interface with the following details:

- Project Structure:** The left sidebar displays the project structure under the "app" module. It includes sections for "manifests", "kotlin+java" (containing "com.example.project1912" package with "Activity" (IntroActivity, MainActivity, ReportActivity), "Adapter" (ExpenseListAdapter, ReportListAdapter), "Domain" (BudgetDomain, ExpenseDomain), "Repository" (MainRepository), and "ViewModel" (MainViewModel)), "res" (drawable, layout, mipmap, values), and "Gradle Scripts" (build.gradle.kts, build.gradle.kts (Module: app), proguard-rules.pro, gradle.properties).
- Code Editor:** The main editor window shows the "ReportActivity.kt" file. The code is as follows:

```
1 package com.example.project1912.Activity
2
3 import ...
4
5 class ReportActivity : AppCompatActivity() {
6     lateinit var binding: ActivityReportBinding
7     private val mainViewModel: MainViewModel by viewModels()
8
9     override fun onCreate(savedInstanceState: Bundle?) {
10         super.onCreate(savedInstanceState)
11         binding = ActivityReportBinding.inflate(layoutInflater)
12         setContentView(binding.root)
13
14     }
15 }
```

The code editor has several tabs at the top: BudgetDomain.kt, MainRepository.kt, MainViewModel.kt, ViewHolder_Budget.xml, ReportListAdapter.kt, ReportActivity.kt (which is the active tab), MainActivity.kt, and activity_report.xml.

The screenshot shows an Android application running on a Pixel 7 Pro API 33 device. The app interface is titled "Monthly Report".

Report Summary:

- Total: \$3,546.31
- income: \$3,245.21
- Expense: \$1,325.78

Total Balance: \$3,452.23 (+25%)

Income: \$165.21 (-10%)

Saving: \$1,542.22 (+16%)

My budget:

- Home Loan: %80.8, \$1,200 /Month
- Subscription: %10.0, \$1,200 /Month
- Car Loan: %30.0, \$800 /Month

MainActivity.kt (Code View):

```
17 class MainActivity : AppCompatActivity() {  
18     ...  
19     private fun setVariable() {  
20         binding.cardBtn.setOnClickListener {  
21             startActivity(  
22                 Intent(  
23                     packageContext: this,  
24                     ReportActivity::class.java  
25                 )  
26             )  
27         }  
28     }  
29     ...  
30     private fun setBlueEffect() {  
31         val radius = 10f  
32         val decorView = this.window.decorView  
33         val rootView = decorView.findViewById<View>(android.R.id.content) as ViewGroup  
34         val windowBackground = decorView.background  
35         binding.blueView.setupWith(  
36             rootView,  
37             RenderScriptBlue( context: this)  
38         )  
39         .setFrameClearDrawable(windowBackground)  
40         .setBlurRadius(radius)  
41         binding.blueView.setOutlineProvider(ViewOutlineProvider.BACKGROUND)  
42         binding.blueView.setClipToOutline(true)  
43     }  
44     ...  
45     private fun initRecyclerview() {  
46         ...  
47     }  
48 }
```

