



## Exercício

1- Para começar a implementação do exercício de RecyclerView adicione as dependências do componente no arquivo **build.gradle(module.app)**.

```
// Bibliotecas usadas para implementar o RecyclerView  
implementation "androidx.cardview:cardview:1.0.0"  
implementation "androidx.recyclerview:recyclerview:1.1.0"
```

2- Adicione o **RecyclerView** no layout do **DetailGastosFragment**.

```
<androidx.recyclerview.widget.RecyclerView...
```

**OBS.:** Use as seguintes configurações

```
app:layoutManager="androidx.recyclerview.widget.LinearLayoutManager"  
tools:listitem="@layout/item_entrada"
```

3- Crie o layout **item\_gasto** ele é a representação gráfica de cada item dentro de sua lista.

4- Crie a classe **DetailGastosAdapter** dentro do pacote de **ui**, esta classe é a responsável de fazer a lógica de implementação do RecyclerView.

**Dica.:**

```
class DetailGastosAdapter(private val listEntradas: List<Gasto>:  
RecyclerView.Adapter<DetailGastosAdapter.DetailGastosViewHolder>() {
```

5- Sobrescreva os métodos **onCreateViewHolder**, **onBindViewHolder** **getItemCount** da classe **DetailGastosAdapter**.

**Atenção.:** Estou colocando o código na íntegra mas não copie escreva

```
override fun onCreateViewHolder(parent: ViewGroup, viewType: Int):  
DetailGastosViewHolder {
```





```
        val itemView = LayoutInflater.from(parent.context).inflate(R.layout.item_gasto,  
parent, false)  
        return DetailGastosViewHolder(itemView)  
    }
```

```
    override fun onBindViewHolder(holder: DetailGastosViewHolder, position: Int) {  
        val currentItem = listEntradas[position]
```

```
        holder.tvDescricao.setText(currentItem.descricao)  
    }
```

```
    override fun getItemCount() = listEntradas.size
```

6- Crie a classe **DetailGastosViewHolder** e inicialize os componentes do layout

```
class DetailGastosViewHolder(itemView: View): RecyclerView.ViewHolder(itemView){
```

```
}
```

7- Criar uma fun para pegar uma lista estática de gastos

```
fun getListGastos(): ArrayList<Gasto> {
```

8- Instanciar o adapter e a lista de gastos

```
    private var listGastos = getListGastos()
```

```
    private val adapter = DetailGastosAdapter(listGastos)
```

9- Iniciar o recyclerView no DetailGastosFragments .kt

```
view.rcListaEntradas.adapter = adapter
```

```
view.rcListaEntradas.layoutManager = LinearLayoutManager(context)
```

```
view.rcListaEntradas.setHasFixedSize(true)
```



### Até aqui

10- Implemente os métodos de adicionar e remover gastos da lista.

```
fun insetGasto(){  
    val entrada3 = Gasto("Almoço", "fixo", "13/10/2020", 15.0)  
    listGastos.add(entrada3)  
    adapter.notifyItemInserted(listGastos.size)  
}
```

```
fun removeGasto(){  
    listGastos.removeAt(listGastos.size - 1)  
    adapter.notifyItemRemoved(listGastos.size - 1)  
}
```

### Extra

- Personalize o layout da lista de gastos.
- Adicione o Navigation na mudança de fragments