

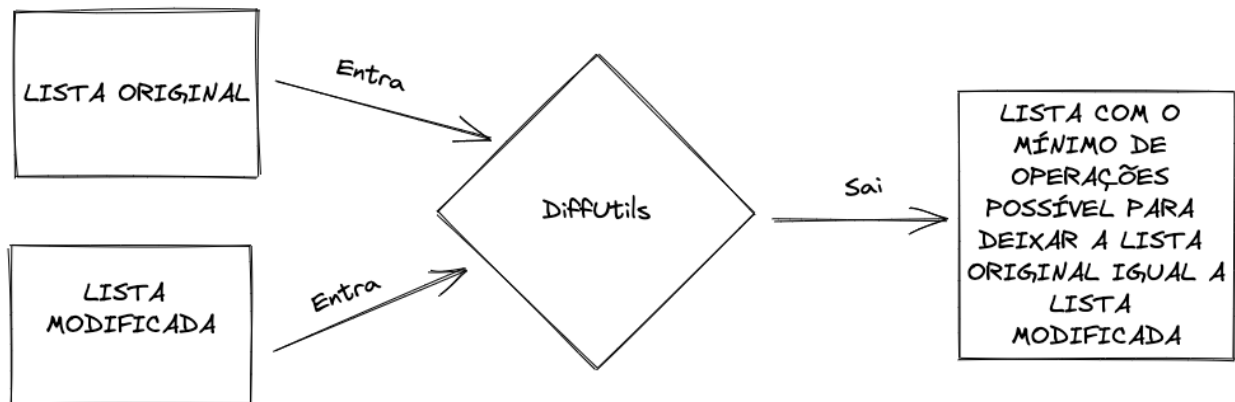


DiffUtils - Performance e Desempenho no seu RecyclerView

- Nas operações com **RecyclerViews**, nós geralmente usamos vários métodos para notificar as mudanças da nossa lista de dados para que essas mudanças aconteçam no **RecyclerView** também.
- Alguns desses métodos são o:
 - `notifyItemInserted()`
 - `notifyItemRemoved()`
 - e o famigerado `notifyDataSetChanged()`, que só deve ser usado em último caso, mas MUITOS iniciantes acabam usando para tudo, e isso DESTRÓI a performance do app.
- Em cenários simples, gerenciar essas mudanças é fácil, como **adição** ou a **remoção** de um item.
- Porém quando o **número de mudanças na lista cresce**, cresce também a **complexidade** de enviar todas essas mudanças para o **adapter** de forma **CERTA** e **OTIMIZADA**.
- Para isso, temos o **DiffUtil**!

Mas o que é o DiffUtil?

Basicamente, o **DiffUtil** é uma classe utilitária que **calcula a diferença entre duas listas** e gera **uma lista de operações** de atualização que converte a primeira lista na segunda.



Ele pode ser usado para calcular **atualizações** para um **Adaptador** RecyclerView.

Curiosidade:

Como o DiffUtils faz isso? Ele usa o **algoritmo de diferença** de **Eugene W. Myers** para calcular o número mínimo de atualizações para converter uma lista em outra.

Não entraremos a fundo nisso, mas no material do Android Definitivo ou na internet você consegue se aprofundar mais no assunto caso tenha interesse!

Só para você ter uma ideia, o algoritmo foi publicado originalmente em 1986.

Se as listas forem grandes, essa operação do **DiffUtil** pode **levar um tempo significativo**, portanto, é recomendável executá-la em um thread em segundo plano, obter **DiffUtil.DiffResult** e aplicá-lo no **RecyclerView** no thread principal.

Veremos isso na prática!

Mão na massa!

1. Vamos criar um **recyclerView** básico do zero, e carregá-lo com informações de exemplo.
2. Agora, criaremos nossa classe **DiffUtil**, **ClientDiffUtil**.

a. **Nomenclatura:**

- i. Geralmente seguimos o padrão do **Adapter**, usando o mesmo prefixo do **Adapter** com o final **DiffUtil**.

Exemplo:

Adapter: ClientAdapter

DiffUtil: ClientDiffUtil

- b. Implementar métodos da classe DiffUtil.

getOldListSize(): Int → Você deve implementar o retorno do tamanho da lista antiga.

getNewListSize(): Int → Você deve implementar o retorno do tamanho da nova lista.