Resumindo a aula:

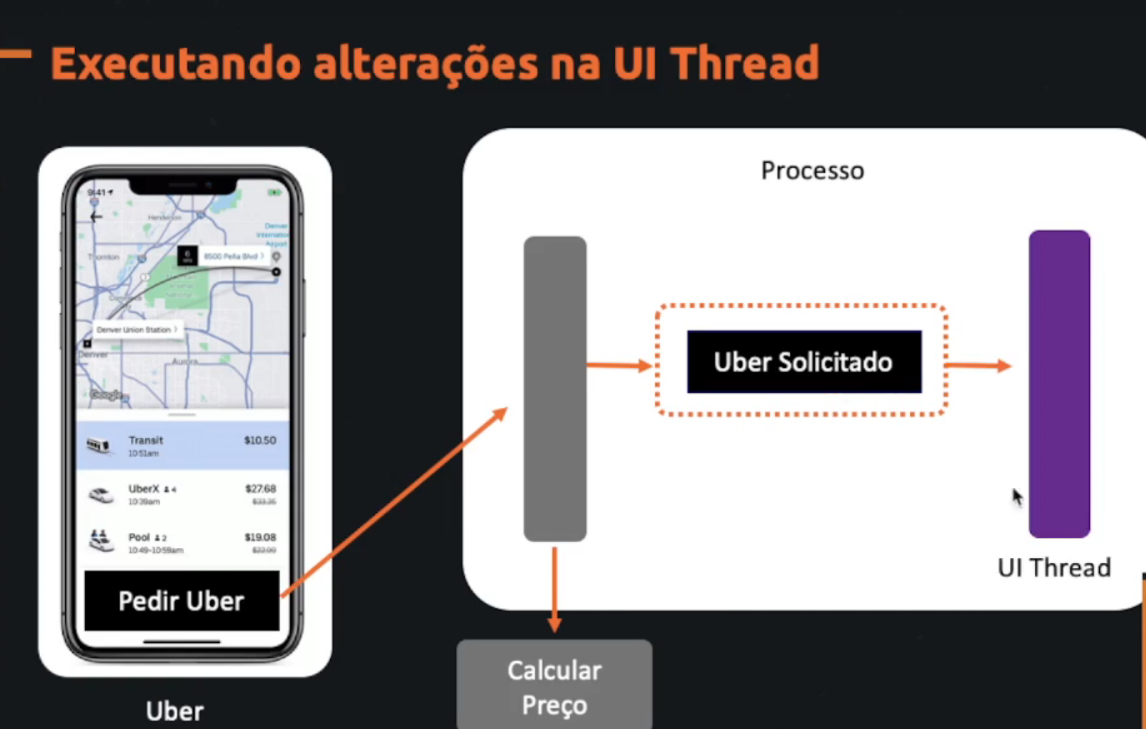
Vamos atualizar os dados que estamos processando na nossa Thread que criamos para a UI Thread com o método runOnUiThread()



Vamos imaginar uma thread para calcular o preço:

Essa thread vai calcular: Distancia, demanda, dinâmico, verificar cartão, motoristas próximos...

Por isso é ideal ter uma thread separada para calcular tudo isso e não travar a thread principal.



Resumindo: vamos criar um thread para calcular o preço, após essa thread calcular o preço vamos alterar o botão para uber solicitado com a thread principal

# Atualizando o botão para indicar que estamos usando uma thread

Pra isso vamos fazer o seguinte, imagina que o usuário clique em PEDIR UBER

Vamos usar uma thread para calcular todos os dados

Vamos usar a Thread principal para atualizar o botão para UBER SOLICITADO

Resumindo: Pediu uber, usou a thread para calcular, notifica a UI Thread para alterar o botão.

Aqui teríamos um erro, porque somente a thread UI pode alterar botões de visualizações

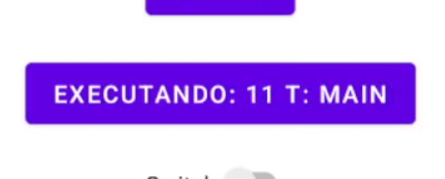
inner class MinhaThread : Thread() {  
 override fun run() {  
 super.run()  
  
 *repeat*(30)**{**indice **->** */\*\* Aqui estamos mandando ele repetir 30x e esperar 1 segundao antes de repetir a proxima  
 \* Metodo Thread.CurrentThread().nome = Cada Thread tem um nome e ai vamos saber qual thread esta processando esse bloco de codigo \*/* Log.i("info\_thread", "Executando: $indice - T: ${Thread.currentThread().*name*}")  
  
 */\*\* Atualizando o botao para indicar que estamosa usando uma thread  
 \*  
 \*/* binding.btnIniciar.*text* = "Executando"  
  
 sleep(1000)  
 **}**

Mas para corrigir isso podemos utilizar o:

Agora sim esta correto, vamos usar o RunUiThread

inner class MinhaThread : Thread() {  
 override fun run() {  
 super.run()  
  
 *repeat*(30)**{**indice **->** */\*\* Aqui estamos mandando ele repetir 30x e esperar 1 segundao antes de repetir a proxima  
 \* Metodo Thread.CurrentThread().nome = Cada Thread tem um nome e ai vamos saber qual thread esta processando esse bloco de codigo \*/* Log.i("info\_thread", "MinhaThread: $indice - T: ${Thread.currentThread().*name*}")  
  
 */\*\* Atualizando o botao para indicar que estamosa usando uma thread  
 // binding.btnIniciar.text = "Executando" ERRADO  
 \*/* runOnUiThread **{** */\*\* runOnUiThread = executar na Thread Principal  
 \* Aqui oque vai acontecer? somente essa linha de codigo que vai ser executada na thread principal  
 \*/* binding.btnIniciar.*text* = "Executando: $indice - T: ${Thread.currentThread().*name*}"  
 **}** sleep(1000)  
 **}** }

Assim vamos usar a thread para atualizar os dados e vamos usar esse método apenas para executar essa linha de código na UI principal



Aqui usamos o código para ver o índice e em qual thread ta funcionando se você reparar chamamos o binding dentro da classe minhaThread, mas mesmo assim ele esta executando na thread principal.

## Atençao:

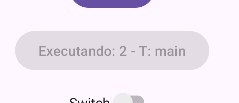


Quando chamar esse método, devemos colocar somente os tipos de dados que vai ser atualizado la na UI

Colocar somente atualizações de interface

## Habilitando e desabilitando o botão

Imagina que estamos executando a ação e não queremos que o usuário clique novamente no botão para executar tudo novamente.



Desabilitamos para o usuário não pode reiniciar a contagem.

Agora quando terminar e chegar no 29 vamos habilitar novamente

Quando chegar a 29 vamos mudar o texto do botão e vamos habilitar ele novamente

*repeat*(30)**{**indice **->** */\*\* Aqui estamos mandando ele repetir 30x e esperar 1 segundao antes de repetir a proxima  
 \* Metodo Thread.CurrentThread().nome = Cada Thread tem um nome e ai vamos saber qual thread esta processando esse bloco de codigo \*/* Log.i("info\_thread", "MinhaThread: $indice - T: ${Thread.currentThread().*name*}")  
  
 */\*\* Atualizando o botao para indicar que estamosa usando uma thread  
 // binding.btnIniciar.text = "Executando" ERRADO  
 \*/* runOnUiThread **{** */\*\* runOnUiThread = executar na Thread Principal  
 \* Aqui oque vai acontecer? somente essa linha de codigo que vai ser executada na thread principal  
 \*/* binding.btnIniciar.*text* = "Executando: $indice - T: ${Thread.currentThread().*name*}"  
 */\*\*Habilitando e desabilitando o botao para o usuario nao clicar e reiniciar  
 \*  
 \*/* binding.btnIniciar.*isEnabled* = false  
  
 if (indice == 29) {  
 */\*\* Agora quando a contagem terminar, vamos deixar o usuario clicar no botao novamente  
 \*  
 \*/* binding.btnIniciar.*text* = "Reiniciar a execução"  
 binding.btnIniciar.*isEnabled* = true  
 }  
 **}**

