Atualizando e removendo os registros.

Primeiro criamos o método para quando o usuário clicar no botão chamar esse método.

btnAtualizar.setOnClickListener **{** atualizar()  
**}**

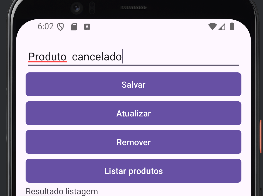


Nessa instrução a cima estamos mudando a senha para 56789 onde o usuário for igual a marcos.almeida

Ficando dessa forma

private fun atualizar() {  
 */\*\* Basicamente vamos atualizar o nome do produto onde o id seja igual a 1  
 \* Poderiamos deixar o usuario definir também qual vai ser o id, mas isso é mais para frente.  
 \*/* val titulo = binding.editProduto.*text*.toString() // Capturando o nome do produto.  
  
 val codigoSQL = "UPDATE produtos SET titulo = '$titulo' WHERE id\_produto = 1 ;"  
  
 try {  
 bancoDeDados.*writableDatabase*.execSQL(codigoSQL)  
 Log.i("info\_db", "Sucesso ao atualizar")  
 }catch (e: Exception) {  
 e.printStackTrace()  
 Log.i("info\_db", "Erro ao atualizar")  
 }  
}

Vai atualizar para o nome que a gente digitar na caixa de texto



# Removendo um produto

btnRemover.setOnClickListener **{** removerProduto()  
 **}  
  
  
 }**}  
  
private fun removerProduto() {  
  
 */\*\* Remover um produto onde o ID seja igual a 01   
 \* Futuramente vamos fazer um recyclerView para poder remover um produto. \*\*/* val codigoSQL = "DELETE FROM produtos WHERE id\_produto = 1"  
  
 try {  
 bancoDeDados.*writableDatabase*.execSQL(codigoSQL)  
 Log.i("info\_db", "Sucesso ao remover")  
 }catch (e: Exception) {  
 e.printStackTrace()  
 Log.i("info\_db", "Erro ao remover")  
 }  
  
}