



Faça como eu fiz

Apresentando mensagens de erro no Login

Apresente a mensagem de erro na tela de login de acordo com a Exception lançada. Utilize a mesma técnica feita na tela de cadastro.

Em vídeo, utilizamos a mesma mensagem para ambas Exceptions esperadas na autenticação.

Validação e refatoração de código

Em seguida, evite o problema de campos vazios colocando um try catch ao autenticar, e validando o formulário de login com as mesmas técnicas usadas na tela de cadastro.

Então, refatore o código aplicando a extração de métodos. Lembre-se também de remover referências que não são mais utilizadas.

Adicionando a tela Minha conta

Adicione a tela pra representar a conta do usuário. Para isso, crie:

- Um novo destino no grafo de navegação com um Fragment e layout pra representar a nova tela;

O layout precisa apenas de um `TextView` para apresentar o e-mail do usuário

- Adicione um novo menu com o mesmo nome da label do destino;
- Adicione um novo asset para representar o ícone do menu;

Com o destino e menu configurados, teste o App e veja se acessa o novo destino pela `BottomNavigationBar`. Se tudo der certo, crie:

- Um `ViewModel` para a tela Minha Conta e adicione uma property com um e-mail;
- Dentro do `onViewCreated` do Fragment, adicione o e-mail no `TextView`.

Confira se o e-mail é apresentado ao entrar na tela **Minha conta**.

Apresentando o e-mail do usuário logado

Apresente o e-mail do usuário logado. Para isso, crie:

- Um método no repositório para devolver um usuário em um `LiveData`;
- Devolva o usuário apenas com o e-mail, para isso, faça com que a property `senha` tenha um valor padrão vazio;
- No `ViewModel`, crie um método que devolva o usuário que vem do repositório;
- Observe o `LiveData` e insira o e-mail do usuário no `TextView`.

Teste o App e confira se apresenta o e-mail do usuário logado. Para o teste, utilize pelo menos dois usuários diferentes.

