

NETWORKING EM FLUTTER - RESUMO SIMPLIFICADO

* Introdução:

- Networking é o processo de enviar e receber dados da internet em apps Flutter.
- Usado para consumir APIs REST (GET, POST, PUT, DELETE).

* Pacote http:

- Adicionado com: `flutter pub add http`
- Importação: `import 'package:http/http.dart' as http;`

* Permissões:

- Android: `AndroidManifest.xml` -> `<uses-permission android:name="android.permission.INTERNET"/>`
- macOS/iOS: ajustes no Info.plist

* Requisições:

- GET: busca dados
- POST: envia dados
- PUT: atualiza
- DELETE: remove

* Parsing JSON:

- Use `jsonDecode()` para converter `response.body`
- Crie classes com `factory fromJson(Map<String, dynamic> json)`

* Exibir dados:

- Use FutureBuilder ou setState após chamadas assíncronas

* Ferramentas:

- DevTools -> Network View para debug de requisições

- Bibliotecas úteis: dio, retrofit, json_serializable

QUESTIONÁRIO - NETWORKING EM FLUTTER

1. Qual pacote básico usado para HTTP em Flutter?

- a) dio b) http c) retrofit d) ws

2. Qual comando adiciona o http ao projeto?

- a) flutter install http
- b) flutter pub add http
- c) pub add http
- d) flutter get http

3. Onde declarar <uses-permission> no Android?

- a) MainActivity.java
- b) AndroidManifest.xml
- c) build.gradle
- d) strings.xml

4. .get() retorna que tipo?

- a) String
- b) Future<http.Response>
- c) Album
- d) http.Response

5. Status HTTP bem-sucedido é:

- a) 100 b) 301 c) 200 d) 500

... (total de 20 perguntas)

GABARITO:

1:b 2:b 3:b 4:b 5:c

... até 20