

Questionário Flutter e Dart

1. Qual ferramenta é recomendada para configurar um novo projeto Flutter e gerenciar a interface do usuário, conforme mencionado nas fontes? a) VS Code b) Android Studio c) Xcode d) IntelliJ IDEA
2. Para que serve o arquivo pubspec.yaml em um projeto Flutter? a) Definir a lógica de negócios do aplicativo. b) Configurar a navegação entre telas. c) Incorporar dependências, recursos personalizados e fontes. d) Gerenciar o estado dos widgets.
3. Qual é a principal diferença entre um StatelessWidget e um StatefulWidget? a) StatelessWidget pode ter estado mutável, StatefulWidget não. b) StatelessWidget é usado para layouts, StatefulWidget para dados. c) StatefulWidget pode ter estado mutável e ser atualizado, enquanto StatelessWidget não pode. d) StatefulWidget é para iOS, StatelessWidget é para Android.
4. O que significa o estilo de programação de UI "declarativo" em Flutter? a) Onde a UI é construída por uma sequência de instruções que modificam o estado. b) Onde os widgets da UI reagem automaticamente às mudanças de estado. c) Onde o desenvolvedor precisa controlar manualmente cada atualização da tela. d) Onde a UI é gerada dinamicamente com base em entradas do usuário.
5. Qual o propósito do método build() em um Widget Flutter? a) Inicializar o estado do widget. b) Definir o comportamento de interações do usuário. c) Descrever a parte da interface do usuário que este widget representa. d) Lidar com requisições de rede.
6. No Dart, para que servem os modificadores final e const? a) Ambos são usados para declarar variáveis mutáveis. b) final é para constantes de tempo de execução, const para constantes de tempo de compilação. c) final é para constantes de tempo de compilação, const para constantes de tempo de execução. d) Ambos tornam uma variável imutável, mas final é mais restritivo.
7. Como o Flutter favorece a customização de widgets? a) Através de herança, estendendo classes de widgets existentes. b) Através de composição, combinando widgets menores para criar novos. c) Usando mixins para adicionar funcionalidade. d) Modificando diretamente o código-fonte dos widgets do Flutter SDK.
8. Qual pacote é recomendado pelo Google para gerenciamento de estado em Flutter, conforme as fontes? a) BLoC b) Redux c) Provider d) GetX
9. Como se constroem aplicativos Flutter multi-telas? a) Usando if/else statements para mostrar diferentes UIs. b) Aprendendo sobre rotas e o widget Navigator. c) Criando um novo MaterialApp para cada tela. d) Utilizando apenas o PageView para todas as transições.
10. Qual é o propósito do pacote http em Flutter? a) Gerenciar o estado global do aplicativo. b) Realizar requisições HTTP para obter dados da internet. c) Criar animações complexas. d) Controlar a câmera do dispositivo.

11. O que `async` e `await` permitem fazer em Dart? a) Sincronizar threads para evitar deadlocks. b) Executar código de forma síncrona em uma Future. c) Realizar programação assíncrona, esperando a conclusão de operações sem bloquear a UI. d) Definir funções que retornam void.

12. Para que serve a biblioteca `dart:convert` ao trabalhar com APIs? a) Encriptar e decriptar dados. b) Converter tipos de dados primitivos. c) Analisar (parsear) dados JSON. d) Gerenciar conexões de banco de dados.

13. Qual o código de status HTTP que geralmente indica que uma requisição foi bem-sucedida? a) 404 b) 500 c) 200 d) 301

14. Como o pacote `http` obtém o corpo da resposta de uma requisição? a) Através do campo `response.headers`. b) Através do campo `response.statusCode`. c) Através do campo `response.body`. d) Através de um callback separado.

15. Que tipo de retorno uma função que realiza uma requisição de rede e processa uma lista de objetos pode ter, considerando que a operação é assíncrona? a) void b) `List<String>` c) `Future<List<Person>>` (assumindo um modelo `Person`) d) `Map<String, dynamic>`

16. Qual é a finalidade do construtor nomeado `from JSON` em uma classe de modelo de dados (como `Person`)? a) Criar uma nova instância da classe a partir de um objeto Dart existente. b) Converter um objeto Dart em uma string JSON. c) Criar uma instância da classe a partir de um mapa JSON (key-value pair). d) Gerar uma representação em string do objeto para depuração.

17. Qual serviço do Firebase pode ser usado para implementar autenticação em aplicativos Flutter?

a) Firebase Cloud Firestore b) Firebase Storage c) Firebase Auth d) Firebase Functions

18. Qual permissão deve ser declarada no `AndroidManifest.xml` para que um aplicativo Flutter no Android possa acessar a internet?

a) `android.permission.READ_EXTERNAL_STORAGE`
b) `android.permission.ACCESS_FINE_LOCATION`
c) `android.permission.CAMERA`
d) `android.permission.INTERNET`

19. O que o pacote `Cupertino` permite construir em Flutter?

a) Widgets com estilo Material Design.
b) Widgets com estilo UIKit para iOS
c) Widgets de layout para Android.
d) Widgets de animação personalizados.

20. Como o Flutter Explained utilizou o `FutureBuilder` em conjunto com a requisição de rede?

a) Para criar um novo thread para a requisição.

- b) Para mostrar uma lista de usuários assim que os dados da rede fossem recebidos.
- c) Para armazenar os dados da rede localmente.
- d) Para realizar a requisição de rede em segundo plano sem impactar a UI.

Respostas:

- Q1: b
- Q2: c
- Q3: c
- Q4: b
- Q5: c
- Q6: b
- Q7: b
- Q8: c
- Q9: b
- Q10: b
- Q11: c
- Q12: c
- Q13: c
- Q14: c
- Q15: c
- Q16: c
- Q17: c
- Q18: d
- Q19: b
- Q20: b