

## Quiz : APIs integration

### Question 1

1. True/False: Flutter provides built-in support for API integration.
2. True/False: You can make API requests in Flutter using the `http` package.
3. True/False: Flutter can only communicate with RESTful APIs.
4. True/False: API integration in Flutter requires the use of external libraries and packages.
5. True/False: Flutter widgets like `FutureBuilder` are useful for handling asynchronous API calls.
6. True/False: Flutter supports both GET and POST requests for API integration.
7. True/False: Error handling is not necessary when making API requests in Flutter.
8. True/False: You can use the `async` and `await` keywords in Flutter to work with asynchronous API calls.
9. True/False: Flutter's `Future` class is commonly used to represent asynchronous operations when dealing with APIs.
10. True/False: You must always use third-party state management libraries like Provider or Riverpod for API integration in Flutter.
11. True/False: You can cache API responses in Flutter to improve performance.
12. True/False: Flutter provides built-in tools for automatic API response parsing.
13. True/False: Flutter allows you to mock API responses for testing purposes.
14. True/False: Flutter's `http` package automatically handles pagination for API endpoints.
15. True/False: You should always store API keys directly in your Flutter source code.
16. True/False: Flutter's `dio` package is a more feature-rich alternative to `http` for API integration.
17. True/False: Flutter's `SharedPreferences` class can be used to store API tokens securely.
18. True/False: Flutter widgets can automatically update when API data changes.
19. True/False: Flutter has a built-in mechanism for rate-limiting API requests.
20. True/False: Flutter can handle real-time data updates from APIs using WebSocket connections.

### Question 2

1. What does API stand for in the context of Flutter?

- a) Application Programming Interface
- b) Android Programming Interface
- c) Application Protocol Interface
- d) Application Program Interface

2. Which Flutter package is commonly used for making HTTP requests to APIs?

- a) http
- b) api\_requester
- c) network\_manager
- d) data\_connector

3. In Flutter, which method is commonly used to perform asynchronous API requests?

- a) Future.delayed()
- b) asyncRequest()
- c) FutureBuilder()
- d) await

4. Which HTTP method is typically used for retrieving data from an API?

- a) POST
- b) PUT
- c) GET
- d) DELETE

5. What is the primary purpose of the `FutureBuilder` widget in Flutter when dealing with API integration?

- a) To define API endpoints
- b) To parse JSON responses
- c) To handle asynchronous tasks and update the UI

d) To perform HTTP requests

6. Which HTTP status code indicates that a resource was successfully created on the server?

a) 200 OK

b) 201 Created

c) 204 No Content

d) 400 Bad Request

7. When parsing JSON data from an API response, what is a common package used to convert JSON strings into Dart objects?

a) json\_serializable

b) json\_parser

c) dart\_json

d) json\_converter

8. What is the purpose of an API key when making requests to some APIs?

a) To specify the HTTP method

b) To authenticate and authorize access to the API

c) To define API endpoints

d) To store API response data

9. Which widget is used to display a loading indicator while waiting for data from an API request?

a) CircularProgressIndicator

b) ProgressIndicator

c) LoadingWidget

d) LoaderIndicator

10. In Flutter, what is the purpose of the `Future` class when working with APIs?

a) To create a new HTTP request

- b) To handle asynchronous operations and represent a potential value or error
- c) To parse JSON data
- d) To define API endpoints

11. Which HTTP status code indicates that the requested resource could not be found on the server?

- a) 200 OK
- b) 401 Unauthorized
- c) 404 Not Found
- d) 500 Internal Server Error

12. To make an authenticated API request, which header is commonly used to send the authentication token?

- a) Authorization
- b) Authentication-Token
- c) Auth-Token
- d) API-Token

13. When should you use the `http` package in Flutter for API requests instead of the `dio` package?

- a) When you need advanced features like caching and interceptors
- b) When you need to perform complex data transformations
- c) When you only need to make simple HTTP requests
- d) When you need WebSocket support

14. Which method is used to handle errors when making API requests using the `http` package in Flutter?

- a) `onError()`
- b) `catchError()`
- c) `onFailure()`
- d) `handleErrors()`

15. What is the purpose of a RESTful API?

- a) To perform remote procedure calls
- b) To create user interfaces
- c) To represent resources as URLs and use HTTP methods for CRUD operations
- d) To manage database connections

16. Which HTTP method is used to update an existing resource on the server in a RESTful API?

- a) POST
- b) PUT
- c) PATCH
- d) DELETE

17. What is the primary benefit of using asynchronous API requests in Flutter?

- a) Improved security
- b) Reduced code complexity
- c) Faster network speed
- d) Preventing UI freezes while waiting for responses

18. In Flutter, how can you pass data from an API request to another screen or widget?

- a) Using the Navigator class
- b) By using the global variable
- c) By embedding the data in the URL
- d) By using the SharedPreferences package

19. Which package is commonly used for managing and storing API response data in Flutter?

- a) redux
- b) provider
- c) hive

d) sqflite

20. What is the purpose of the `http.Response` object in Flutter when making API requests?

- a) To define API endpoints
- b) To represent the HTTP request
- c) To store API request data
- d) To represent the HTTP response and its properties