**Flutter and React Native: Interview Test**

Flutter:

**1. What is Flutter?**

- Explain what Flutter is and its primary use.

Answer => Flutter is a framework built with dart. And it is used for building multi platform apps for operating systems like mac os,android, ios, linux,windows os, and even the website. It is fast, smooth, relatively easier. And you just need minor adjustment for each platform apps.

- Discuss its advantages and disadvantages.

Answer => Advantages of flutter are:

1. You can write cross-platform apps with one code base.
2. Flutter is one of the fastest hybrid technologies for building apps.
3. It is relatively easier.
4. Its syntax is easier to learn.
5. It has a huge community and it is funded.
6. File size of finished apps is extremely light

Disadvantages of flutter are:

1. It is not as fast as native technologies like swift and java
2. If you want to write intricate platform apps, you will need to write native plugins

**2. What are widgets in Flutter?**

- Define widgets and their role in Flutter.

Answer => Widgets are basically classes that have a visible form or are used to arrange visible widgets. They have the characteristics of classes which allows you to build even more complex widgets from the default ones. Their roles include:

1. providing visible parts of the app
2. Accepting user inputs and sending it to the backend for processing
3. Some affect the layout of the visible ones

- Differentiate between Stateful and Stateless widgets.

Answer => Widgets in flutter are basically non rewritable (immutable) and sometimes, some widgets need to respond to frequently changing data. This is where state comes in. State is a changeable property of a widget.

* Stateless widgets are widgets with no state characteristics. Ie they are static, immutable.
* While stateful widget have state characteristics ie they have frequently changing data

**3. Explain Flutter's architecture.**

- Describe how Flutter renders UI and manages its state.

Answer => Flutter builds UIs with a tree of lightweight, immutable widgets. These widgets describe the UI and its data. When data changes, the relevant widget is rebuilt, and a diffing process identifies the minimal changes needed on the screen. This reactive approach ensures the UI reflects the latest state efficiently. State can be local to widgets or managed externally using providers or libraries, depending on the complexity of your app.

**4. What is the significance of the `setState` method?**

- Discuss how state management works in Flutter.

Flutter offers a flexible approach to state management. By default, widgets can hold their own state for simple scenarios. For more complex needs, the state can be externalized using providers or third-party libraries. These external state holders act like central sources of truth, accessible from various widgets in the app. When the state updates with the use of the setState((){}) method, widgets that depend on it are notified, triggering a rebuild and ensuring the UI reflects the latest information. This reactive approach keeps your UI in sync with the underlying data throughout your application.

**Coding Tasks**

Flutter:

**Task:** Create a todo list app with the following features:

* Users can add, remove, and mark tasks as completed.
* Users can filter tasks by all, completed, and incomplete.
* Persist tasks using local storage.

***\*\*Candidates can submit the code and question into github and share to us, good luck***