

# React Native Advance interview Questions

- [React Native Advance interview Questions](#)
  - [React native Advance Interview Questions](#)

# React Native Advance interview Questions

# React native Advance Interview Questions

Explain the concept of 'flexbox' and its role in React Native layout.

'Flexbox' is a layout model that allows you to distribute space and align-items within a container. It simplifies complex layouts and ensures that components adjust dynamically to different screen sizes. In React Native, 'flexbox' is the primary layout system and is used to position and organize UI components.

Describe the purpose of 'AsyncStorage' in React Native.

'AsyncStorage' is an API in React Native for asynchronous, unencrypted, and persistent storage of small amounts of data. It's often used to store settings, preferences, or authentication tokens locally on the user's device.

How can you integrate Redux with a React Native app?

In React Native, Redux can be integrated by creating a store to hold the global state and using components like 'Provider' to make the state available to the app. Actions, reducers, and middleware are used to update and manage the state.

# How can you integrate third-party libraries in a React Native app?

Third-party libraries can be integrated using package managers like npm or yarn. You install the library, import it into your code, and use its components or APIs. Native modules can also be linked using the 'react-native link' command.

# How do you handle form validation in React Native?

Form validation in React Native is often done using state and event handling. You capture user input using components like 'TextInput', validate the input based on your criteria, and display error messages accordingly.

# What is the role of 'navigator' in React Navigation?

'Navigator' is a core component in React Navigation that manages the navigation flow between screens in a React Native app. It defines the app's navigation structure, handles transitions, and maintains the navigation stack.

# Explain the role of managed and bare workflows in Expo.

Expo's managed workflow offers simplicity and speed but limited access to native modules. The bare workflow provides full control but requires more setup and native code management.

# Can you describe the difference between user interface (UI) design and UX design?

User interface (UI) design and user experience (UX) design are two distinct but interconnected disciplines.

UI design focuses on the visual elements and interactions within a digital product. It involves creating aesthetically pleasing and functional interfaces that are visually appealing and intuitive for users.

UI designers are responsible for designing elements such as buttons, icons, color schemes, typography, and layout.

On the other hand, UX design encompasses the entire user journey and aims to create a seamless and meaningful experience for users.

## What is a user persona and why is it important in UX design?

A user persona is a fictional representation of a target user based on research and data. It helps designers understand user needs, behaviors, and motivations.

Personas bring focus and empathy to the design process by humanizing the target audience.

By referring to user personas, designers can make informed decisions about functionality, features, and design elements that resonate with their target users.

## What are some common UX design principles?

Some of the most common UX design principles include:

- a) User-centered design: Prioritize the needs and goals of the target users throughout the design process.
- b) Consistency: Maintain a consistent visual and interactive experience to reduce cognitive load.
- c) Accessibility: Ensure that the design is inclusive and usable by people with disabilities.
- d) Learnability: Make it easy for users to understand and navigate the interface without extensive instructions.
- e) Feedback and responsiveness: Provide clear and timely feedback to user actions to enhance the sense of control and understanding.
- f) Simplicity: Strive for simplicity in design, minimizing complexity and unnecessary elements.
- g) Visual hierarchy: Use visual cues to guide users' attention and prioritize important information.

## What is MVC and MVVM pattern?

MVC(Model View Controller) MVVM(Model View ViewModel) Oldest android app architecture. Industry-recognized architecture pattern for applications. User Inputs are handled by the Controller.

## What is the role of an Adapter in Android UI?

An Adapter in Android UI serves as a bridge between the data source and the AdapterView (ListView or RecyclerView) to efficiently display large datasets. It handles data retrieval, view creation, and recycling of views for optimal performance.

## What are Pure Components?

Pure components are the components which render the same output for the same state and props. In function components, you can achieve these pure components through memoized `React.memo()` API wrapping around the component. This API prevents unnecessary re-renders by comparing the previous props and new props using shallow comparison. So it will be helpful for performance optimizations.

## What is state in React?

State of a component is an object that holds some information that may change over the lifetime of the component. The important point is whenever the state object changes, the component re-renders. It is always recommended to make our state as simple as possible and minimize the number of stateful components.

## What are props in React?

Props are inputs to components. They are single values or objects containing a set of values that are passed to components on creation similar to HTML-tag attributes. Here, the data is passed down from a parent component to a child component.

The primary purpose of props in React is to provide following component functionality:

1. Pass custom data to your component.
2. Trigger state changes.
3. Use via `this.props.reactProp` inside component's `render()` method.

# What is the difference between state and props?

In React, both state and props are plain JavaScript objects and used to manage the data of a component, but they are used in different ways and have different characteristics. state is managed by the component itself and can be updated using the `setState()` function. Unlike props, state can be modified by the component and is used to manage the internal state of the component. Changes in the state trigger a re-render of the component and its children. props (short for "properties") are passed to a component by its parent component and are read-only, meaning that they cannot be modified by the component itself. props can be used to configure the behavior of a component and to pass data between components.

# What is the purpose of callback function as an argument of `setState()`?

The callback function is invoked when `setState` finished and the component gets rendered. Since `setState()` is asynchronous the callback function is used for any post action.

Note: It is recommended to use lifecycle method rather than this callback function.

```
setState({ name: "John" }, () =>

  console.log("The name has updated and component re-rendered")

);
```

# What is "key" prop and what is the benefit of using it in arrays of elements?

A key is a special attribute you should include when mapping over arrays to render data. Key prop helps React identify which items have changed, are added, or are removed.

Keys should be unique among its siblings. Most often we use ID from our data as key:

```
const todoItems = todos.map((todo) => <li key={todo.id}>{todo.text}</li>);
```

# What is the use of refs?

The ref is used to return a reference to the element. They should be avoided in most cases, however, they can be useful when you need a direct access to the DOM element or an instance of a component.

## What is the lifecycle of a component in React hooks?

A React component undergoes three phases in its lifecycle: mounting, updating, and unmounting. The mounting phase is when a new component is created and inserted into the DOM or, in other words, when the life of a component begins

## What are hooks in React interview questions?

Answer 1: React Hooks are a feature in React that allows developers to use state and other React features without writing a class component. Hooks are functions that let developers "hook into" the React state and lifecycle methods from a functional component

## What is the difference between hooks and functional components?

Key Differences between React Hooks and Functions

React Hooks are specifically designed to interact with the React component lifecycle and state management. React Hooks can only be used within functional components, whereas JavaScript functions can be used anywhere in your code

## What is Axios?

Axios is a promise-based HTTP client for making asynchronous HTTP requests. It works in the browser and in Node.js

## What is an API?

An API (Application Programming Interface) is a software intermediary that enables two applications to communicate with each other. It comprises a number of subroutine definitions, logs,



and tools for creating application software.

In an API testing interview, you could be asked to give some [API examples](#), here are the well-known ones: Google Maps API, Amazon Advertising API, Twitter API, YouTube API, etc.

## What are the common API testing types?

While there are certainly specialty tests, and no list can be asked to be comprehensive in this realm, most tests fit broadly into the following nine categories that you should remember before attending an API testing interview.

1. Validation Testing
2. [Functional Testing](#)
3. [UI testing](#)
4. Load testing
5. Runtime/ Error Detection
6. Security testing
7. Penetration testing
8. Fuzz testing
9. Interoperability and WS Compliance testing

## Can you explain your process for finding and resolving bugs in your code?

Software developers often create their own debugging processes, as they regularly complete this task in their roles. A [recruiter](#) might ask you this question to learn about the process you use to debug software and determine whether it is efficient and effective. In your response, share the steps you take chronologically and explain why this process works for you.

## What do 'stepping over' and 'stepping into' mean in a debugging context?

There are different debugging techniques you can use as a software developer. Recruiters might ask you this question if they want to evaluate your understanding of these primary techniques for [identifying and resolving bugs](#). Respond thoroughly by defining each of these concepts and explaining when they might be useful.

# What is a stack overflow?

Stack overflows can cause significant challenges in software development. An interviewer might ask this question if they want to make sure you understand these and can manage them efficiently in the role. In your answer, explain what a stack overflow is, the issues it may cause and how you might resolve it.

# Explain what is SQLite?

SQLite is a mostly ACID compliant relational database management system contained in a relatively small C programming library.

# Mention what are the SQLite storage classes?

SQLite storage classes include

- Null: The value is a NULL value
- Integer: The value is a signed integer (1,2,3, etc.)
- Real: The value is a floating point value, stored as an 8 byte IEEE floating point number
- Text: The value is a text string, stored using the database encoding ( UTF-8, UTF-16BE)
- BLOB (Binary Large Object): The value is a blob of data, exactly stored as it was input

# What is Deep link in iOS?

Ans :

Deep linking is the idea of not only having a clickable link to open up your app but a smart one that will also navigate to the desired resource

What is deeplinking with context?

Contextual deep linking is a type of targeted advertising that uses the context of a user's current location, search query, or other factors to provide them with a more relevant link to a specific piece of content. It improves UX by providing relevant content, and it tracks user behavior.

# What is the difference between context API and Redux?

Context API is perfect for sharing small bits of information between components. On the other hand, Redux is more powerful and comes with helpful features that Context API doesn't have. It's great for managing big chunks of data and handling requests to APIs

## What is the context API in React interview questions?

What is Context API? The React Context API is a way for a React app to effectively produce global variables that can be passed around. This is the alternative to "prop drilling" or moving props from grandparent to child to parent, and so on.

## What is a ClassLoader?

A classloader in Java is a subsystem of Java Virtual Machine, dedicated to loading class files when a program is executed; ClassLoader is the first to load the executable file.

Java has Bootstrap, Extension, and Application classloaders.

## What is a Promise in JavaScript and how does it work?

The interviewer may start with basic questions before asking more advanced ones. This question assesses your understanding of what Promises are and how you work with them. In your response, define the object and talk about its functions and states to demonstrate your strong subject knowledge.

1. What is the difference between a Promise and an Observable in JavaScript?

## What are logical operators in JavaScript?

Logical operators allow developers to compare variables and perform tasks based on the result of the comparison. As a hiring manager, you'd ask this question to gauge the candidate's familiarity with the language and its fundamental features. Your candidate should be able to explain each logical operator and their behavior – stepping through each operand and computing its output.

There are four logical operators in JavaScript:

- || – OR

- && – AND
- ! – NOT
- ?? – Nullish Coalescing (see next question)

## Explain loops in JavaScript.

We often require repeat actions. Loops are a way to execute the same code multiple times. The candidate should be able to explain how to use loops in JavaScript. An ideal answer should include the pros and cons of each looping method and its respective applications.

There are two main ways to create loops in JavaScript – while and for. Both methods consist of a condition to stop the loop and a “loop body”, the code that will be executed multiple times.

## What are the differences between call, apply, and bind?

JavaScript has multiple ways to indirectly invoke a function. Your candidate needs to understand the differences between each and their use cases. You, as the candidate, should be able to explain not only their differences conceptually but also their use case and the reason behind it.

call, apply, and bind are different methods to tie a function to an object and call the function within the specified context.

## What are the features of Android architecture?

Android architecture refers to the various layers in the Android stack. It consists of operating systems, middleware, and applications. Each layer in the Android architecture gives different services to the layer just above it.

The five layers present in the Android stack are:

- Linux Kernel - It is responsible for device drivers, device management, memory management, power management, and resource access.
- Libraries - There are a set of libraries having open-source Web browser engine WebKit, well-known library libc, libraries to play and record audio and video, SQLite database for sharing of application data and storage, SSL libraries for internet security, etc.
- Android Runtime - There are core libraries along with DVM (Dalvik Virtual Machine) or ART(Android RunTime) as runtime which is helpful for running an Android application. DVM is optimized for mobile devices. DVM provides fast performance and consumes less

memory. Replacing DVM, ART(Android RunTime) virtual machine was introduced to execute android apps from Android lollipop 5.0 version (API level 21).

- Android Framework - It consists of Android APIs like UI (User Interface), resources, content providers (data), locations, telephony, and package managers. It provides interfaces and classes for the development of Android applications.
- Android Applications - Applications like home, games, contacts, settings, browsers, etc. uses the Android framework that will make use of Android runtime and libraries.

## What is an activity?

Activity in java is a single screen that represents GUI(Graphical User Interface) with which users can interact in order to do something like dial the phone, view email, etc.

For example, the Facebook start page where you enter your email/phone number and password to log in acts as an activity.

## What is a service in Android?

Service is an application component that facilitates an application to run in the background in order to perform long-running operations without user interaction. A service can run continuously in the background even if the application is closed or even after the user switches to another application.

## How do I integrate third party payment gateway?

1. Step 1: Provider onboarding. In this step, the provider connects their Sharetribe account with the payment gateway.
2. Step 2: Customer checkout. Customer checkout happens when the customer initiates a transaction.
3. Step 3: Provider accept.
4. Step 4: Customer refund.
5. Step 5: Provider payout.