## Question 1: What do you mean by RESTful Web Services?

RESTful Web Services are web services based on the REST (Representational State Transfer) architecture. REST uses HTTP methods like GET, POST, PUT, DELETE to perform operations on resources.  
  
- Resources are represented in formats like JSON or XML.  
- RESTful services are stateless, meaning each request from the client contains all the information needed.  
- They are widely used in modern web applications for data communication between the frontend (like React) and backend (server/database).

## Question 2: What is JSON-Server? How do we use it in React?

JSON-Server is a simple tool that allows developers to create a mock REST API using a JSON file as a database.  
  
- It is mainly used for frontend development and testing before connecting to a real backend.  
- With JSON-Server, developers can simulate CRUD operations without building a backend.  
  
Usage in React:  
1. Install JSON-Server globally:  
 npm install -g json-server  
2. Create a db.json file with sample data.  
3. Start server:  
 json-server --watch db.json --port 3001  
4. Use the API endpoint (e.g., http://localhost:3001/users) inside React for fetching data.

## Question 3: How do you fetch data from a JSON-Server API in React? Explain the role of fetch() or axios().

In React, we use fetch() or axios() to make API requests to the JSON-Server.  
  
Using fetch():  
useEffect(() => {  
 fetch("http://localhost:3001/users")  
 .then(response => response.json())  
 .then(data => setUsers(data))  
 .catch(error => console.error("Error:", error));  
}, []);  
  
Using axios():  
import axios from "axios";  
  
useEffect(() => {  
 axios.get("http://localhost:3001/users")  
 .then(response => setUsers(response.data))  
 .catch(error => console.error("Error:", error));  
}, []);  
  
- fetch() → Native JavaScript function to make API calls.  
- axios() → A library that provides easier syntax, automatic JSON conversion, and better error handling.

## Question 4: What is Firebase? What features does Firebase offer?

Firebase is a Backend-as-a-Service (BaaS) platform developed by Google. It provides ready-to-use backend services so developers can focus on building frontend applications.  
  
Features of Firebase:  
1. Firebase Realtime Database – Store and sync data in real-time.  
2. Cloud Firestore – Scalable NoSQL database.  
3. Authentication – Supports Google, Facebook, Email/Password logins.  
4. Hosting – Deploy web apps with free SSL.  
5. Cloud Storage – Store user-generated content (images, videos, etc.).  
6. Firebase Cloud Messaging (FCM) – Send push notifications.  
7. Analytics – Track user behavior in apps.

## Question 5: Importance of Handling Errors and Loading States in React APIs

When working with APIs in React, it is essential to handle:  
  
- Loading State → To show the user that data is being fetched. Example: "Loading…".  
- Error State → To notify the user if something goes wrong. Example: "Failed to load data".  
  
Example:  
const [loading, setLoading] = useState(true);  
const [error, setError] = useState(null);  
  
useEffect(() => {  
 fetch("http://localhost:3001/users")  
 .then(res => res.json())  
 .then(data => {  
 setUsers(data);  
 setLoading(false);  
 })  
 .catch(err => {  
 setError(err.message);  
 setLoading(false);  
 });  
}, []);  
  
This ensures a better user experience, prevents app crashes, and makes debugging easier.