

Q1. What is an API? Give an example, where an API is used in real life.

APIs are mechanisms that enable two software components to communicate with each other using a set of definitions and protocols.

For example, the weather bureau's software system contains daily weather data. The weather app on your phone "talks" to this system via APIs and shows you daily weather updates on your phone.

Q2. Give advantages and disadvantages of using API.

Advantage : 1) Increased visibility and traffic 2) Seamless integration 3) increased efficiency 4) easier maintenance 5) Reduced Costs 6) improve customer experience

Disadvantage : 1) Increased complexity 2) Limited Functionality 3) Dependency on third party service 4) Security and privacy Risks

Q3. What is a Web API? Differentiate between API and Web API.

Web api : WEB API is the subset of API, The normal api works on TCP/IP protocol while web api works on HTTP protocol, in layman language the transfer and receiving of data on the web by following HTTP protocols is called web api

Implementation: APIs can be implemented using any programming language or technology, while Web APIs are typically implemented using web-based languages and frameworks, such as JavaScript, XML, or JSON.

Platform: APIs can be used on any platform, including desktop applications, mobile devices, and embedded systems, while Web APIs are specifically designed for use on web-based applications.

Scope: API is a general term that can refer to any type of interface for accessing software functionality or data, while Web API specifically refers to APIs that use web-based protocols.

Access: APIs can be accessed through a variety of different methods, such as local function calls or network sockets, while Web APIs are typically accessed over the internet using HTTP or other web protocols.

Standards: Web APIs often follow specific standards, such as REST or SOAP (Simple Object Access Protocol), while APIs may or may not follow any specific standards or protocols.

Q4. Explain REST and SOAP Architecture. Mention shortcomings of SOAP.

REST (Representational State Transfer) is a lightweight, web-based architecture that uses the HTTP protocol to exchange data between clients and servers. The API client can then interact with these resources using the HTTP methods (GET, POST, PUT, DELETE) to retrieve, create, update, or delete data.

SOAP (Simple Object Access Protocol) is a more complex architecture that uses XML to transfer data between clients and servers. SOAP is more heavyweight than REST and is designed to work with more complex distributed systems. SOAP defines a set of rules for how messages should be formatted and sent between applications, including a message envelope, header, and body.

Some Shortcomings of SOAP :

Limited Support: SOAP has limited support for caching, which can lead to slower performance and increased network traffic.

Complexity: SOAP is more complex and has a steeper learning curve than REST, which can make it more difficult for developers to use.

Overhead: SOAP messages are typically larger and more verbose than REST messages, which can lead to increased network traffic and slower performance.

Q5. Differentiate between REST and SOAP.

REST(Representational State Transfer) :

Security : It is less secure than SOAP. **Protocol :** It uses HTTP protocol only for Data Transferring. **Weight :** It is very light weight. **Language :** It send data in JSON(Java Script Object Notation). **Speed :** REST API's are Faster than SOAP API's.

SOAP(Simple Object Access Protocol) :

Security : It is more secure than REST. **Protocol :** It uses SMTP, TCP/IP , HTTP protocol for Data Transferring. **Weight :** It is heavy weight. **Language :** It send data in XML(Extensible Markup Language). **Speed :** SOAP API's are Slower than REST API's.