Web API and RESTful Services in Detail

1. What is a Web API?

A Web API (Application Programming Interface) is a way for applications to communicate with each other over the web (usually using HTTP/HTTPS). It exposes endpoints (URLs) that clients can call and returns data (JSON, XML, etc.) instead of full HTML pages.

2. REST and RESTful Services

REST (Representational State Transfer) is an architectural style for designing networked applications. A RESTful service is a Web API that follows REST principles.

REST Principles

- Statelessness: Each API call is independent.
- Client-Server Architecture: Separation between client and server.
- Uniform Interface: Resources identified with URIs.
- Representation of Resources: JSON, XML, etc.
- Cacheable: Responses can be cached.
- Layered System: Multi-layered architecture possible.

3. HTTP Methods in REST

GET: Retrieve resource(s)
POST: Create new resource
PUT: Update/replace existing resource
PATCH: Partially update resource
DELETE: Delete resource

4. Example of RESTful Web API (JSON)

Request: GET /api/products/101

Response:
{
"id": 101,
"name": "Laptop",
"price": 75000,
"inStock": true
}

5. Web API vs RESTful API

- Web API: General term for exposing functionality/data over the web (can use SOAP, REST, GraphQL, gRPC, etc.)
- RESTful API: A type of Web API that strictly follows REST principles.

6. Advantages of RESTful Services

- Lightweight (JSON, HTTP)
- Platform-independent
- Scalable and stateless
- Easy to cache
- Widely used and supported

7. Real-World Examples of RESTful APIs

- GitHub API → /users/{username}/repos
- Twitter API → /tweets
- Google Maps API → /geocode