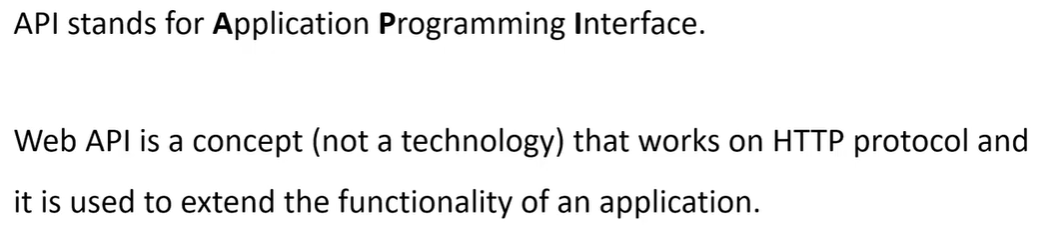
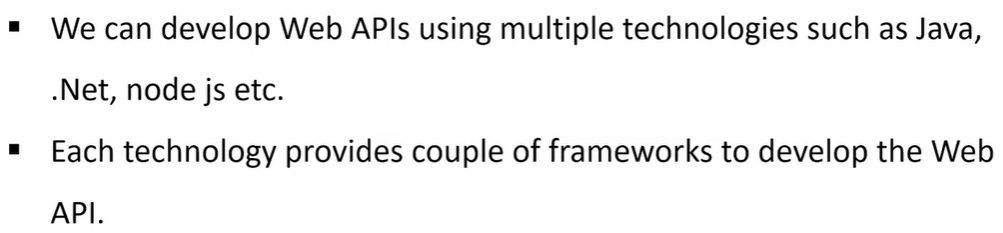
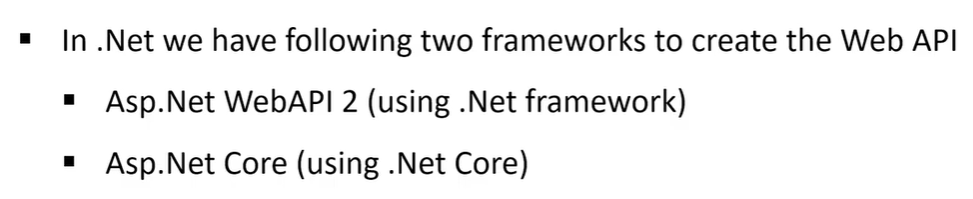
**What is Web API**



**How to Create Web API**



**Web API using .NET**



*What is REST?*

**RE**presentational **S**tate **T**ransfer

Principles for REST:

1. Uniform Interface
2. Client-Server
3. Stateless
4. Cacheable
5. Layered System
6. Code on Demand

*What is Resource?*

The key abstraction of information in REST is a resource. Any information that we can name can be a resource. For example, a REST resource can be a document or image, a temporal service, a collection of other resources, or a non-virtual object.

The resource representations are consisted of:

* the data
* the metadata describing the data
* and the hypermedia links that can help the clients in transition to the next desired state.

*Self-Descriptive:*

**Resource representations** shall be self-descriptive: the client does not need to know if a resource is an employee or a device. It should act based on the media type associated with the resource.

*Resource Methods:*

These resource methods are used to perform the desired transition between two states of any resource.

**Summary:**

* REST architectural style, data and functionality are considered resources and are accessed using Uniform Resource Identifiers (URIs).
* The resources are acted upon by using a set of simple, well-defined operations. Also, the resources have to be decoupled from their representation so that clients can access the content in various formats, such as HTML, XML, plain text, PDF, JPEG, JSON, and others.
* The clients and servers exchange representations of resources by using a standardized interface and protocol. Typically, HTTP is the most used protocol, but REST does not mandate it.
* Metadata about the resource is made available and used to control caching, detect transmission errors, negotiate the appropriate representation format, and perform authentication or access control.
* And most importantly, every interaction with the server must be stateless.
* All these principles help RESTful applications to be simple, lightweight, and fast.