Q: What is ASP.NET Web API?

ASP.NET Web API is a framework that simplifies building HTTP services for broader range of clients (including browsers as well as mobile devices) on top of .NET Framework.
Using ASP.NET Web API we can create non-SOAP based services like plain XML or JSON strings etc. with many other advantages including:
Create resource-oriented services using the full features of HTTP.
Exposing services to a variety of clients easily like browsers or mobile devices etc.

Q: What are the advantages of using ASP.NET Web API?

Using ASP.NET Web API has a number of advantages, but core of the advantages are:

It works the HTTP way using standard HTTP verbs like GET, POST, PUT, DELETE etc for all CRUD operations.
Complete support for routing.
Response generated in JSON or XML format using MediaTypeFormatter.
It has the ability to be hosted in IIS as well as self-host outside of IIS.
Supports Model binding and Validation.
Support for OData.
and more....
For implementation on performing all CRUD operations using ASP.NET Web API, click here.

Q: What new features are introduced in ASP.NET Web API 2.0?

More new features introduced in ASP.NET Web API framework v2.0 are as follows:
Attribute Routing
External Authentication
CORS (Cross-Origin Resource Sharing)
OWIN (Open Web Interface for .NET) Self Hosting
IHttpActionResult
Web API OData
You can follow a good Web API new feature details on Top 5 New Features in ASP.NET Web API 2 here.

Q: WCF Vs ASP.NET Web API?

Actually, Windows Communication Foundation is designed to exchange standard SOAP-based messages using variety of transport protocols like HTTP, TCP, NamedPipes or MSMQ etc.
On the other hand, ASP.NET API is a framework for building non-SOAP based services over HTTP only.
For more details, please follow here.

Q: Is it true that ASP.NET Web API has replaced WCF?

It's a misconception that ASP.NET Web API has replaced WCF. It's another way of building non-SOAP based services, for example, plain XML or JSON string etc.
Yes, it has some added advantages like utilizing full features of HTTP and reaching more clients such as mobile devices etc.
But WCF is still a good choice for following scenarios:
If we intended to use transport other than HTTP e.g. TCP, UDP or Named Pipes.
Messag Queuing scenario using MSMQ.
One-way communication or Duplex communication
A good understanding for WCF(Windows Communication Foundation), please follow WCF Tutorial.

Q: MVC Vs ASP.NET Web API?

As in previous ASP.NET Web API Interview Questions, we discussed that purpose of Web API framework is to generate HTTP services that reaches more clients by generating data in raw format, for example, plain XML or JSON string. So, ASP.NET Web API creates simple HTTP services that renders raw data.

On the other hand, ASP.NET MVC framework is used to develop web applications that generates Views as well as data. ASP.NET MVC facilitates in rendering HTML easy.

Q: MVC Vs Web API

As in previous ASP.NET Web API Interview Questions, we discussed that purpose of Web API framework is to generate HTTP services that reaches more clients by generating data in raw format, for example, plain XML or JSON string. So, ASP.NET Web API creates simple HTTP services that renders raw data.
On the other hand, ASP.NET MVC framework is used to develop web applications that generates Views as well as data. ASP.NET MVC facilitates in rendering HTML easy

Q: How to return View from ASP.NET Web API method?

(A tricky Interview Question) No, we can't return view from ASP.NET Web API Method. As we discussed in earlier interview question about difference between ASP.NET MVC and Web API that ASP.NET Web API creates HTTP services that renders raw data. Although, it's quite possible in ASP.NET MVC application.

Q: How to restrict access to Web API method to specific HTTP Verb?

Attribute programming plays it's role here. We can easily restrict access to an ASP.NET Web API method to be called using a specific HTTP method. For example, we may required in a scenario to restrict access to a Web API method through HTTP POST only as follows:

```
 [HttpPost]
 public void UpdateStudent(Student aStudent)
 {
      StudentRepository.AddStudent(aStudent);
 }
```

Can we use Web API with ASP.NET Web Form?

Yes, ASP.NET Web API is bundled with ASP.NET MVC framework but still it can be used with ASP.NET Web Form.
It can be done in three simple steps as follows:
Create a Web API Controller.
Add a routing table to Application_Start method of Global.asax.
Make a jQuery AJAX Call to Web API method and get data.
jQuery call to Web API for all CRUD (Create, Retrieve, Update, Delete) operations can be found here.

Q: How we can provide an alias name for ASP.NET Web API action?

We can provide an alias name for ASP.NET Web API action same as in case of ASP.NET MVC by using "ActionName" attribute as follows:

```
 [HttpPost]
[ActionName("SaveStudentInfo")]
 public void UpdateStudent(Student aStudent)
 {
      StudentRepository.AddStudent(aStudent);
 }
```

Q: What is Web API?

It is a framework which helps us to build/develop HTTP services. So there will a client server communication using HTTP protocol.

Q: What is Representational state transfer or REST?
REST is architectural style, which has defined guidelines for creating services which are scalable. REST used with HTTP protocol using its verbs GET, POST, PUT and DELETE.

Q: Explain Web API Routing?

Routing is the mechanism of pattern matching as we have in MVC. These routes will get registered in Route Tables. Below is the sample route in Web API –

```
Routes.MapHttpRoute(
 Name: "MyFirstWebAPIRoute",
 routeTemplate: "api/{controller}/{id}
 defaults: new { id = RouteParameter.Optional}
};
```

Q: List out the differences between WCF and Web API?

WCF

It is framework build for building or developing service oriented applications.
WCF can be consumed by clients which can understand XML.
WCF supports protocols like – HTTP, TCP, Named Pipes etc.

Web API

It is a framework which helps us to build/develop HTTP services
Web API is an open source platform.
It supports most of the MVC features which keep Web API over WCF.

Q: What are the advantages of using REST in Web API?

REST always used to make less data transfers between client and server which makes REST an ideal for using it in mobile apps. Web API supports HTTP protocol thereby it reintroduces the old way of HTTP verbs for communication.

Q: Difference between WCF Rest and Web API?

WCF Rest

"WebHttpBinding" to be enabled for WCF Rest.
For each method there has to be attributes like – "WebGet" and "WebInvoke"
For GET and POST verbs respectively.
Web API

Unlike WCF Rest we can use full features of HTTP in Web API.
Web API can be hosted in IIS or in application.

Q: List out differences between MVC and Web API?

Below are some of the differences between MVC and Web API

MVC

MVC is used to create a web app, in which we can build web pages.

For JSON it will return JSONResult from action method.
All requests are mapped to the respective action methods.
Web API

This is used to create a service using HTTP verbs.
This returns XML or JSON to client.
All requests are mapped to actions using HTTP verbs.

Q: What are the advantages of Web API?

Below are the list of support given by Web API –

OData
Filters
Content Negotiation
Self Hosting
Routing
Model Bindings

Q: Can we unit test Web API?

Yes we can unit test Web API.

Q: How to unit test Web API?

We can unit test the Web API using Fiddler tool. Below are the settings to be done in Fiddler –

Compose Tab -> Enter Request Headers -> Enter the Request Body and execute

Q: Can we return view from Web API?

No. We cannot return view from Web API.

Q: How we can restrict access to methods with specific HTTP verbs in Web API?

Attribute programming is used for this functionality. Web API will support to restrict access of calling methods with specific HTTP verbs. We can define HTTP verbs as attribute over method as shown below

```
[HttpPost]
 public void UpdateTestCustomer(Customer c)
 {
 TestCustomerRepository.AddCustomer(c);
 }
```

Q: Can we use Web API with ASP.NET Web Forms?

Yes. We can use Web API with ASP.NET Webforms.

Q: List out the steps to be made for Web API to work in Web Forms?

Below are the steps to be followed –

Creating new controller for Web API.
Adding routing table to "Application_Start" method in Global.asax
Make a AJAX call to Web API actions.

Q: Explain how to give alias name for action methods in Web API?

Using attribute "ActionName" we can give alias name for Web API actions. Eg:

```
[HttpPost]
[ActionName("AliasTestAction")]
 public void UpdateTestCustomer(Customer c)
 {
 TestCustomerRepository.AddCustomer(c);
 }
```

Q: With WCF also you can implement REST, So why WebAPI?

WCF was brought in to implement SOA, never the intention was to implement REST.
WebAPI is built from Scratch and the only goal is to create HTTP services using REST.

Q: What is the difference between ASP.NET MVC Vs ASP.NET WebAPI?

MVC vs ASP.NET WebAPI

ASP.NET MVC	ASP.NET WebAPI
used to create web applications that returns both view and data	used to create HTTP services, It returns only data.
return data in json format using jsonResult	return data in JSON, XML
Requests are mapped to actions name.	Requests are mapped to the actions based on HTTP verbs

Q: WebAPI where is the proxy?

Web API doesn't make it easy for consumers to generate a service client like a SOAP WSDL does.
If you're only ever going to have .NET clients it's not a big deal because they can share the contract objects you
implement, but other language clients will need to manually create their client objects if you don't use SOAP

Q: What are the Advantages using WebAPI?

OData support (via Queryable attribute)
Content Negotiation

Filters
Model binding and validation
Ability to self host outside of IIS
Link generation to related resources that incorporates routing rules
Full support for routes/routing
Ability to create custom help and test pages using IApiExplorer

Q: What are the Http methods used to implement a RESTful API

RESTful API HTTP methods

Resource	
Collection URI, such as http://example.com/resources	Create a new entry in the collection. The new entry's URI is assigned automatically and is usually returned by the operation.
Element URI, such as http://example.com/resources/ item17	Not generally used. Treat the addressed member as a collection in its own right and create a new entry in it.
GET	DELETE
List the URIs and perhaps other details of the collection's members.	Delete the entire collection.
Retrieve a representation of the addressed member of the collection, expressed in an appropriate Internet	Delete the addressed member of the collection.
PUT	The PUT and DELETE methods are idempotent methods. The GET method is a safe method (or nullipotent), meaning that calling it produces no side-effects.
Replace the entire collection with another collection.	
Replace the addressed member of the collection, or if it doesn't exist, create it.	
POST	