1. **What is Web API Routing? Can we have multiple API Routing Configuration files in single application?**

Routing is the mechanism of pattern matching as we have in MVC. These routes will get registered in Route Tables.

Yes we can have multiple API Routing Configuration files. All Routing configurations that should be used in application should be registered in Global.asax file.

1. **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);  
}

1. **What is the difference between MVC Routing and Web API Routing?**

There should be atleast one route defined for MVC and Web API to run MVC and Web API application respectively. In Web API pattern we can find “api/” at the beginning which makes it distinct from MVC routing. In Web API routing “action” parameter is not mandatory but it can be a part of routing.

1. **Explain Exception Filters?**

Exception filters will be executed whenever controller methods (actions) throws an exception which is unhandled. Exception filters will implement “IExceptionFilter” interface.

1. **Explain Authentication in Web API?**

Web API authentication will happen in host. In case of IIS it uses Http Modules for authentication or we can write custom Http Modules. When host is used for authentication it used to create principal, which represent security context of the application.

1. **When to use “FromUri” and “FromBody” in Web API?**

In Web API to read complex types from URL we will use “FromUri” attribute to the parameter in action method. Eg:

public MyValuesController : ApiController  
{  
public HttpResponseMessage Get([FromUri] MyCustomer c) { … }  
}

This attribute is used to force Web API to read the simple type from message body. “FromBody” attribute is along with parameter. Eg:

public HttpResponseMessage Post([FromBody] int customerid, [FromBody] string customername) { … }

1. **What are Authentication Filters in Web API?**

Authentication Filter will let you set the authentication scheme for actions or controllers. So this way our application can support various authentication mechanisms.

1. **What is “Under-Posting” and “Over-Posting” in Web API?**

* “Under-Posting” - When client leaves out some of the properties while binding then it’s called under – posting.
* “Over-Posting” – If the client sends more data than expected in binding then it’s called over-posting.

**9. How to resolve the Cross Origin Resource Sharing?**

Cross Origin Resource Sharing (CORS) allows the server to share the resources being requested by the client application by overcoming the problem of same-origin policy. It is a very common requirement while developing the Asp.net WebAPI applications at the time of deployment where an external client application will try to access the resources by sitting at a different domain where as the resources are available at some other domain and that causes the root issue. CORS overcomes this issue by providing the needed permission for accessing the different domains.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| ***Q7:* What is the difference between ApiController and Controller?** | |  |  |  |  |  | | --- | --- | --- | --- | --- | |  |  |  |  | **Junior** | |

* Use **Controller** to render your normal views.
* **ApiController** action only return data that is serialized and sent to the client.

Consider:

public class TweetsController : Controller {

// GET: /Tweets/

[HttpGet]

public ActionResult Index() {

return Json(Twitter.GetTweets(), JsonRequestBehavior.AllowGet);

}

}

or

public class TweetsController : ApiController {

// GET: /Api/Tweets/

public List<Tweet> Get() {

return Twitter.GetTweets();

}

}

**What exactly is OAuth (Open Authorization)?**

**OAuth** (Open Authorization) is an open standard for access granting/deligation protocol. It used as a way for Internet users to grant websites or applications access to their information on other websites but without giving them the passwords. It does not deal with authentication.

Basically there are three parties involved: oAuth Provider, oAuth Client and Owner.

* oAuth Client (Application Which wants to access your credential)
* oAuth Provider (eg. facebook, twitter...)
* Owner (the person with facebook,twitter.. account )

**Why should I use IHttpActionResult instead of HttpResponseMessage?**

***Q2:* Explain the usage of HttpResponseMessage?**

HttpResponseMessage works with HTTP protocol to return the data with status/error.