1. **WCF:**
   1. **ABC in WCF:**

* Address: This lets you know the location on server. Different bindings support different address types.
* Binding: Defines which protocol is being used.
* Contract: This defines each method exposed from service.
  + 1. **Contract:**
* ServiceContract
* OperationContract
* DataContract
* MessageContract
  + 1. **BINDING:**
* **HTTP-based:** If we want our service to be accessed across multiple OS or multiple programming architectures, HTTP based bindings are our obvious choice. Let’s see the bindings supported.
  + BasicHttpBinding: It is used in case of web service, offers backward compatibility, the message encoding used is Text/XML, supports WS-basic profile.
  + WsHttpBinding: It gives all functionality which BasicHttpBinding offers, apart from that it offers transaction support, reliable message and WS-Addressing.
  + WSDualHttpBinding: Offers all functionality offered by WsHttpBinding, but the main purpose is to be used with duplex communication.
  + WsFederationHttpBinding: This is used when security within the organization is the most important aspect.
* **TCP-based:** If we want to share the data in compact binary format, these bindings are of best use.
  + NetNamedPipeBinding: This is best binding to be used if our service and client both are hosted on the same machine, use TCP protocol to exchange data. It supports transaction, reliable session and secure communication.
  + NetPeerTcpBinding: This binding provided secure binding for P2P network, offers all functionality fro NetNamedPipeBinding.
  + NetTcpBinding: Used for secure and optimized binding for cross-machine communication between .NET application.
* **MSMQ-based:** If we want to use MSMQ server to exchange data, we can use these bindings:
  + MsmqIntegrationBinding: We can use this binding to send and receive data from existing MSMQ applications that use COM, C++.
  + NetMsmqBinding: This is used to communicate between cross-machine using queue. This is preferred binding when using MSMQ.

1. **Exception handling:**

Wcf exception handling – fault exception

Wep api exception handling – custom exception