# Esercitazione di Fine Settimana – Week 4

|  |  |  |  |
| --- | --- | --- | --- |
|  |  | Nome | Carlotta |
|  |  | Cognome | Colla |
|  |  | Data | 16/04/2021 |

Leggete attentamente ogni domanda e argomentare quanto più possibile fornendo anche degli esempi.

1. Quali sono i componenti che caratterizzano un Endpoint?  
   I componenti di un Endpoint sono gli ABC:  
   -Address: è l’indirizzo del servizio, sono composti da un identificativo (come http o https) e un url (come localhost se il servizio è in locale)  
   -Binding: come viene pubblicato, indica il tipo di protocollo che si usa, solitamente è basicHttp  
   -contract: cosa rende disponibile il servizio, nel servizi WCF solitamente è IService.
2. Cos’è un Web Service?  
   Un Web service è un software che permette la comunicazion tra diversi dispositivi attraverso la rete internet. Si compone di server che mettono a disposizione dei servizi e dei client che effettuano delle richieste.
3. Quali parti costituiscono una HTTP Response?  
   Le rispotte Http si compongono di:  
   - Header: qui vengono inserite delle informazioni riguardo al server e alla risposta  
   - Codice: il codice della risposta (200 = risposta affermativa, 400 = c’è stato un errore lato cliente, 500 c’è stato un errore lato server  
   -Body: se la risposta deve riportare dei dati all’utente questi vengono inseriti nel body utilizzando il formato JSON.
4. Descrivere l’utilizzo degli attributi [DataContract] e [OperationContract]  
   [DataContract] viene utilizzato all’inizio delle entità per specificare che saranno utilizzate per la comunicazione tra client e server quindi dovranno essere serializzate per utilizzare il formato JSON.  
   L’attributo [OperationContract] viene utilizzati per i servizi WCF nell’interfaccia IService. L’attributo [ServiceContract] viene messo prima della dichiarazione dell’interfaccia per indicare che verranno inseriti lì i servizi. All’inizio dei servizi si utilizza l’attributo [OperationContract] per indicare che que servizio sarà visibile al client. I servizi verranno implementati nella classe Service che implementerà l’interfaccia IService.
5. Come vengono utilizzati gli HTTP Methods in un servizio REST?  
   Un servizio rest ha bisogno di un controller per indicare quali sono i servizi disponibili al client. Questi servizi vengono preceduti da degli attributi per indicare che tipo di metodo HTTP sono:  
   - HttpGet: viene utilizzato per richiedere dei dati al server, può prendere dei parametri in input (come per il GetByID)  
   - HttpPut: viene utilizzato per effettuare l’update dei dati  
   - HttpPost: viene utilizzato per indicare un servizio di creazione di un oggetto, i dati dell’oggetto vengono passati nel body della chiamata  
   - HttpDelete: serve per le operazioni di delete, può prendere in input dei dati (come l’id dell’oggetto da eliminare)
6. Come viene configurato un servizio realizzato con ASP.NET Core WebAPI?  
   Per configurare un servizio ASP .NET Core WebAPI serve un controller in cui vengono indicati i servizi che sono visibili dal client (operazioni get, put ..), queste operazioni dovranno essere collegate ai dati (ad esempio utilizzando un business layer che richiami le operazioni crud delle entità o una classe che implementa una lista di oggetti da utilizzare), nello startup project dovrà essere indicato in che modo collegarsi al database (inserire la stringa di connessione con AddDbContext e i repository che servono con AddTransient<IRepository, Repository>();

**Esercitazione Pratica**

* Realizzare un database per la Gestione degli Ordini e dei Clienti.
  + ***Cliente***
    - *ID* (int, PK), CodiceCliente (*string*), Nome (*string*), Cognome (*string*)
  + ***Ordine***
    - *ID* (int, PK), *DataOrdine* (date), *CodiceOrdine* (string), *CodiceProdotto* (string), *Importo* (decimal)
  + La realizzazione dello strato di accesso al dato deve essere realizzata con EF Code-first.
* Realizzare un servizio WCF per la gestione di una Anagrafica Clienti (CRUD)
* Realizzare un servizio REST per la gestione di una Anagrafica Ordini (CRUD)