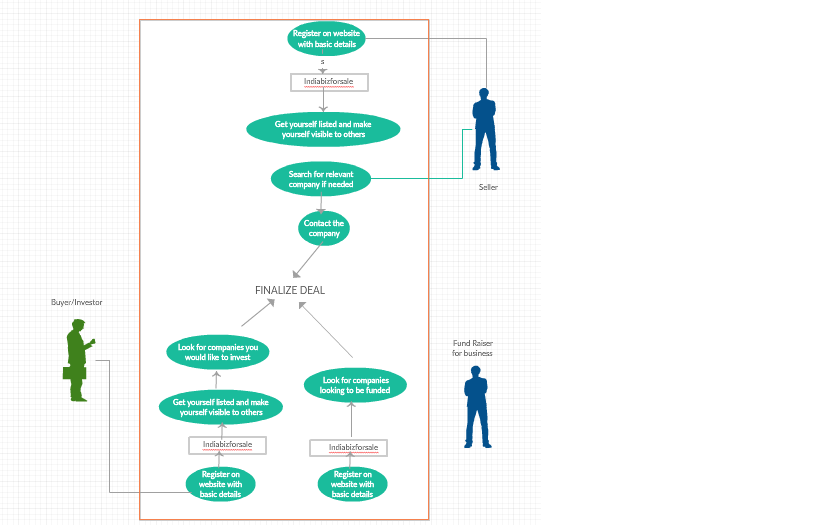
TASK 1

How would you store the information in a database (suggest a schema or data model)?



The above mentioned UML Use case diagram is the depiction of the process that takes place as the interaction of the buyer and seller happens before finalising the deal. Thus it is important to keep in mind the attributes to be considered for the deal to take place. These set of inputs can be stored in a Database Management System like MySql.

|  |
| --- |
| SELLER |

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Company\_number | Company\_name | Size | Franchise | Fran\_no | Sales | Deals\_with | Profit |
|  |  |  |  |  |  |  |  |

|  |  |  |  |
| --- | --- | --- | --- |
| Fran\_name | Fran\_no | Lic\_free | Reg\_fee |
|  |  |  |  |

|  |  |  |
| --- | --- | --- |
| Reputation | Legal\_info | No\_of\_directors |
|  |  |  |

|  |  |  |
| --- | --- | --- |
| Profit | EBITDA | Fin\_to |
|  |  |  |

|  |
| --- |
| BUYER |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Buyer\_id | Buyer\_name | Buyer\_company | Buyer\_company\_id | Buyer\_company\_history |
|  |  |  |  |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Company\_id | Company\_size | Sales | Value\_sale | Profit |
|  |  |  |  |  |

Seller

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Company\_number | Company\_name | Size | Franchise | Fran\_no | Sales | Deals\_with | Profit |
|  |  |  |  |  |  |  |  |

Franchisee

|  |  |  |  |
| --- | --- | --- | --- |
| Fran\_name | Fran\_no | Lic\_free | Reg\_fee |
|  |  |  |  |

Sale

|  |  |  |
| --- | --- | --- |
| Reputation | Legal\_info | No\_of\_directors |
|  |  |  |

Profit

|  |  |  |
| --- | --- | --- |
| Profit | EBITDA | Fin\_to |
|  |  |  |

Buyer

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Buyer\_id | Buyer\_name | Buyer\_company | Buyer\_company\_id | Buyer\_company\_history |
|  |  |  |  |  |

Company

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Company\_id | Company\_size | Sales | Value\_sale | Profit |
|  |  |  |  |  |

What kind of database would you choose for implementing the said schema/data model Relational or non-relational? And why?

Ans: I would choose to use a Relational database as

* It is easy to link one or more columns of a table based on the attributes
* Due to the ACID property (Atomicity,  Consistency, Isolation, Durability )
* Since we need to select depending upon the criteria of fulfillment and satisfy the needs of the seller and buyer it is important to consider every aspect.

TASK 2

Implement a simple CRUD application with any modern MVC framework, using language of your choice.

CRUD refers to Create Read Update and Delete.   
The MVC framework consists of ASP.Net or C# language

I have chosen to do it on Visual Studio by creating a DB for creation of tables. On creation, using Linq to SQL class using C#.  
  
This is done by creating the SQL Table and adding a new item. Once created, using get and set methods the MVC framework can be compiled.  
  
Since am not aware about the C# and MVC framework I could come up with only the DBMS (SQL Commands).

CREATE TABLE seller

(

company\_number INT NOT NULL PRIMARY KEY,

company\_name CHAR(50) NOT NULL ,

size INT NOT NULL ,

franchise CHAR(40) NOT NULL ,

fran\_no INT ,

sales CHAR (30),

deals\_with CHAR(65),

profit INT,

)

CREATE TABLE franchisee

( fran\_name CHAR(40) NOT NULL,

fran\_no INT NOT NULL PRIMARY KEY,

lic\_fee INT NOT NULL,

reg\_fee INT NOT NULL,

);

CREATE TABLE sale

(

reputation INT NOT NULL ,

legal\_info char(100) NOT NULL,

no\_of\_diretors INT NOT NULL,

)

CREATE TABLE profit

(

profit INT NOT NULL,

EBITDA INT NOT NULL ,

fin\_to INT NOT NULL,

)

CREATE TABLE buyer

(

buyer\_id INT NOT NULL PRIMARY KEY,

buyer\_name CHAR(30) NOT NULL ,

buyer\_company CHAR (50),

buyer\_company\_history VARCHAR(150),

)

CREATE TABLE company

(

company\_id INT NOT NULL PRIMARY KEY,

company\_size INT ,

sales INT,

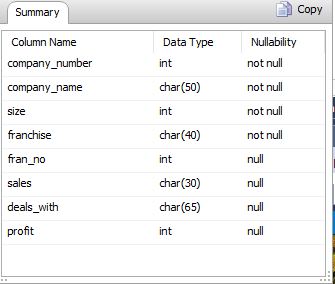
value\_sale INT,

profit INT ,

)

UPDATE TABLE seller SET company\_number=2411 WHERE company\_name='X Enterprises'

SELECT seller;



DELETE seller;

Also provide a reason for choosing a specific language and framework to build this

Since I had no prior knowledge in MVC framework, I have taken the help of Internet and using Visual Studio. This can be done by using **LINQtoSQL class wherein a connection can be initiated. On connecting, the database will be displayed in the server explorer, then the required table can be dragged and dropped. Here the table name is converted to a C# class and attributes to properties. The table data is retrieved, buttons are formed and then the connection must be checked. The entry point is checked and any required changed as applied.**