*Van Abeshi*

# Data Modeling Questions

* Design a database to model a cookbook with meal types, recipes, ingredients.
  1. You can use paper to document your data model.

*See Attached File : cookbook.jpg*

# SQL Questions

* Given the following SQL SELECT statement what are the highlighted text represent?

SELECT ***CUST***.CUSTOMER AS ***NAME***, ***CUST***.CUSTOMER\_ADDRESS

FROM CUSTOMER\_V ***CUST***

WHERE ***NAME*** = ‘Home Furnishings’;

*The AS operator is used in order to temporarily rename the results of the query, so they can further be worked with in terms of representing the result of said query, rather than permanently changing the column heading to the new name.*

* Given the following Orders and Customers tables:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **OrderID** | | **CustomerID** | **OrderDate** | |
| 10308 | | 2 | 1996-09-18 | |
| 10309 | | 37 | 1996-09-19 | |
| 10310 | | 77 | 1996-09-20 | |
| **CustomerID** | **CustomerName** | | | **ContactName** | | **Country** |
| 1 | Alfreds Futterkiste | | | Maria Anders | | Germany |
| 2 | Ana Trujillo Emparedados y helados | | | Ana Trujillo | | Mexico |
| 3 | Antonio Moreno Taquería | | | Antonio Moreno | | Mexico |

What would the result set be given the following SQL SELECT statement?

SELECT Orders.OrderID, Customers.CustomerName, Orders.OrderDate  
FROM Orders  
INNER JOIN Customers  
ON Orders.CustomerID=Customers.CustomerID;

*The result would be a table showing CustomerName and OrderID, left to right.*

What would the result set be given the following SQL SELECT statement?

SELECT Orders.OrderID, Customers.CustomerName, Orders.OrderDate  
FROM Orders  
LEFT JOIN Customers  
ON Orders.CustomerID=Customers.CustomerID;

*The result would be a table showing OrderID, CustomerName and OrderDate, left to right.*

# Entity Framework Questions

* What do the following Entity Framework Object Services provide for your application in regards to data from a database?
  1. Materialization

*Produces objects based on records.*

* 1. Change Tracking

*Makes it so that relationships are updated dynamically based on entity changes.*

* 1. Object identities

*Makes it so objects have an identifiable “name”, for lack of a better word, meaning that they are more easily called, referenced, or modified*.

* Design a code first data model which has a Project class that can contain a bunch of tasks.

*public  class Person*

*{*

*public int ID { get; set; }*

*[StringLength(50)]*

*[Display(Name = "Last Name")]*

*public string LastName { get; set; }*

*[Column("FirstName")]*

*[Display(Name = "First Name")]*

*public string FirstName { get; set; }*

*[Display(Name = "Full Name")]*

*public string FullName*

*{*

*get*

*{*

*return LastName + ", " + FirstMidName;*

*}*

*}*

*}*

# Object Orientation Questions

* What are the basic concepts of OOP?

*Object oriented programming languages are modular, meaning that certain parts of the program can be modified without effecting the program as a whole. They use classes as “blueprints” and objects as “finished products”. As such they are more easily used by teams.*

* How do you program behavior into your C# class?

*By either writing your own function or referring to one in the references, you give the program behavior, rather than just attributes. Also, further logic can be introduced using loops / switches etc.*

* Explain method overriding.

*Used when calling a new instance of a virtual or abstract class, method overriding assigns some new function or implementation to said new class.*

* What is Inheritance?

*public class vanAbeshi : person*

*{*

*some random func()*

*}*

*Inheritance is a reference of sorts that states that vanAbeshi is a new instance of the person class, but I want it to have all the traits that person has, plus whatever I think of now.*

* What is abstract class?

*Generally a class that is intended as an incomplete base class for other classes to inherit from.*

# MVC Questions

What is an example URL that would call the following controller method, assuming the default routes have been configured?

*www.somebs.com/Catalog*

Part II: What HTTP Verb is used?

*Get*

public class CatalogController : Controller

 {

     public ActionResult Specifications(int id)

     {

         var model = new SpecModel(id);

         return View();

     }

}

Examine the following View for an MVC Application:

@model IEnumerable<MVCGuidedLab.Models.Color>

@{

    ViewBag.Title = "Index";

}

<h2>Index</h2>

<p>

    @Html.ActionLink("Create New", "Create")

</p>

<table class="table">

    <tr>

        <th>

            @Html.DisplayNameFor(model => model.Name)

        </th>

        <th>

            @Html.DisplayNameFor(model => model.Value)

        </th>

        <th></th>

    </tr>

@foreach (var item in Model) {Go

    <tr>

        <td>

            @Html.DisplayFor(modelItem => item.Name)

        </td>

        <td>

            @Html.DisplayFor(modelItem => item.Value)

        </td>

        <td>

            @Html.ActionLink("Edit", "Edit", new { id=item.Id }) |

            @Html.ActionLink("Details", "Details", new { id=item.Id }) |

            @Html.ActionLink("Delete", "Delete", new { id=item.Id })

        </td>

    </tr>

}

</table>

1. What type is the Model?

*IEnumerable*

1. Is the model a single object, or a sequence?

*Sequence*.

1. What properties are on the items in the model?

# Programming Exercises

To turn in this exam, you’ll create a fork of a repository, make modifications to the project in that repository, and submit a pull request with your changes.

We’ll walk you through the github workflow for those pieces. However, the code will be yours.

Go to Github.com and fork the repository <https://github.com/BillWagner/ExperienceITExam>

Then, clone your fork to your desktop. You’ll do that by clicking the “Clone in Desktop” button on the github page.

Now, you’re ready to do your work.

1. Open the program.cs file in the Loops project. The Main method has comments that describe what you should do. Make your changes.
2. Open the program.cs file in the Extension methods project. The Main method has comments that describe the code you should add. Make your changes.
3. Open the program.cs file in the Lazy Evaluation project. The Main method has comments that describe the code you should add. Make those changes as well.
4. Open the program.cs file in the Query Expresssions project. As before, the Main method has comments that describe the code you should or change.
5. Make a new ASP.NET MVC project and add it to the solution. Add a controller called DiceRoll controller. Modify the index method and the index view to display all the combinations of results from rolling 2 six-sided dice. (For example, { 1, 1}, {1, 2} etc.

Commit your changes, and then push them to github.

Now, you’re ready to submit a pull request. Navigate to your fork of the repository in github.com in a browser. Below the Code tab on the right side, you’ll see a link that says “Pull Request”. Click that. Once you’re on the Pull Request page, submit a new pull request. Add your name, and any comments you would like on your pull request, and submit the request.

We can now look at your changes, and see how you did.