**Examen W24 Intra – Feuille de Réponses**

**Feuille de Réponses (14 Questions)**

1-

Identity is an authentication service that can be used in .NET (DOTNET). Essentially, it is a way to secure our applications with roles and authorizations. We use it for example to have registration and login in place.

2-

Entity Framework is an ORM that is used by .NET applications. An ORM is an object-relational mapper, which converts data between a database and an object-oriented programming language such as C# and JavaScript.

3-

ViewData and ViewBag are means to transfer data from a controller to a view. They both temporarily store data until it is no longer needed for the purpose it was used for. However, there is a slight difference between the two. ViewData is a form of a dictionary that allows us to store data, while ViewBag is a dynamic property. It is, in fact, a wrapper that surrounds ViewData and dynamically uses properties to retrieve and set data.

4- (Note : Unsure) :

On doit intégrer TinyMCE dans le CSHTML approprié.

5-

B) [Authorize]

6-

C) if(User.Identity.IsAuthenticated)

7-

Oui, il est possible. Si nous n’avons pas mis [Authorize] dans notre contrôleur, une personne qui connais le lien exact pourrait accéder en ajoutant, par exemple, /create dans le URL et avoir accès à la section create.

Un exemple provenant de ma partie pratique :  
A computer screen shot of text

Description automatically generated

8-

Nous devons aller dans le fichier où le ViewModel c3p se trouve pour s’assurer que nous avons le [Key, Column]. Par la suite, nous allons dans le fichier cshtml approprié (par exemple, l’Index). Nous programmons le model de cette façon (par exemple): @model IEnumerable<Exam.ViewModels.c3p>. On en train de dire de prendre l’information dans le fichier c3p qui se trouve dans le répertoire ViewModels du projet Exam et c’est ce qu’on veut utiliser.

9- (Note : This answer is bare bones)

Il s’agit d’une route. Essentiellement, nous seront redirigé à l’Index.

10- (Note : To ask if this is what the question meant)

Dans un premier temps, il doit faire une modification de sa *default connection* avec la base de données appropriée (par exemple, "Server=.\;Database=Northwind;Trusted\_Connection=True;MultipleActiveResultSets=true;TrustServerCertificate=True"). Par la suite, il doit s’assurer qu’il a les bons *packages* NuGet installés (comme Microsoft.EntityFrameworkCore.SqlServer et Microsoft.EntityFrameworCore.Tools). Finalement, dans le Package Manager Console, il faudra qu’il utilise la commande : Update-Database. En principle, il sera en mesure de voir des nouvelles tables.  
  
Pour ce qui est des roles, il devra ajouter les rôles dans son Visual Studio en ajoutant a *new scaffolded item* et en ajoutant Identity. Il sera en mesure d’utiliser les rôles. Il faudra avant tout cela qu’il passe la commande Scaffold dans la console.

11- (Note : I don’t think I answer this one well.)

La solution la plus simple serait de modifier tout ce qui a des nulls. Le problème est qu’il a des données qui avait des nulls (par exemple, lorsqu’il ajouté les nouvelles tables, s’il a fait une relation entre les Users et les Produits, les anciennes données n’avait pas de User FK associé). Il aurait été préférable qu’au moment où il était en train de modifier sa base de données (sa table) et qu’il autorise les nulls pour ne pas avoir de problèmes.

12-

C) Il a oublié de modifier son service dans le fichier Program.cs

13- (Note : Unsure, this is a 50/50 gamble, but it is because I am modifying things in for the user in the view and controller if i want only some people to have access to things for example)

A) Vrai

14-

Le moment où il est allé à l’index, le ViewData s’est vidé. Le ViewData stocke (garde) les données temporairement. Tant et aussi longtemps qu’on est où il est appelé, il aura les données. Dès qu’on sort de l’emplacement du ViewData, les données disparaissent.