

TP .Net core

Exercice1:

Soit la classe suivante:

```
public class User
{
    public int id { get; set; }
    public string nom { get; set; }
    public int prenom { get; set; }
}
```

- 1.Générer la table de cette classe en sql server à travers EntityFramework.Net Core codeFirst
- 2.Générer les opérations CRUD sur cette table d'une manière automatique

Exercice2: on suppose qu'on a une base de donnée Etudiant(cne,nom,prenom,age)

- 1.Transmettre cette base en classe à travers Entity Framework Core Database Fisrt
- 2.Créer un formulaire qui ajoute un étudiant dans la base de donnée

Exercice3:

- Créer et exécuter une application ASP MVC sous linux
- Essayer d'effectuer des modifications sous Visual studio code

Exercice4:

Créer un projet ASPMVC6.Réaliser l'exemple de cours et tester les trois modes d'injection (Singleton ,Scope et transcient)

```
public interface IService1
{
    String GetValeurInstance();
}
```

```
public class Service1Imp:IService1
{
    public String GetValeurInstance()
    {
        return "Bonjour " + GetHashCode();
    }
}
```

```

public class Service1Controller : Controller
{
    private IService1 s1;
    private IService1 s2;

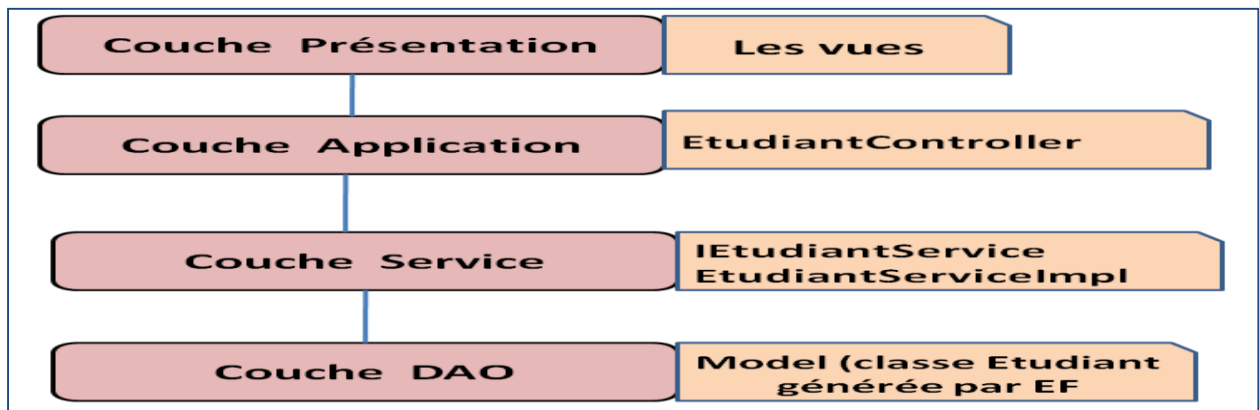
    public Service1Controller(IService1 s1, IService1 s2)
    {
        this.s1 = s1;
        this.s2 = s2;
    }

    public IActionResult Index()
    {
        ViewData["message"] = s1.GetValeurInstance() + " " + s2.GetValeurInstance();
        return View();
    }
}

```

Exercice5:

Réaliser une application ASP MVC .Net Core de gestion des étudiants en suivant l'architecture suivante



-Interface à implémenter dans la classe EtudiantServiceImp

```

public interface IEtudiantService
{
    void Save(etudiant p);
    IEnumerable<etudiant> FindAll();
    etudiant GetEtudiantbyId(int id);
}

```

-La vue de gestion des étudiants

Index			
Ajouter			
nom	prenom	Nom_filiere	
ennaji	maha	Informatique	Details
Mouradi	ahmed	Mecanique	Details
soulami	sara	Informatique	Details
ouazni	ali	Electricite	Details