

Vous répondrez sur ces feuilles, sans les dégrafer, dans les cadres prévus à cet usage.
Les questions sont toutes indépendantes et peuvent être traitées dans l'ordre que vous souhaitez.

Nom : Prénom :
Groupe : Note :

Outils et Modèles pour le Génie Logiciel : Bases de Données

On considère le schéma relationnel suivant, donné en compréhension :

Les schémas des relations :

Intervenants (NumIntervenant , NomIntervenant , PrénomIntervenant , AdresseIntervenant , CPIntervenant , VilleIntervenant , TélIntervenant , NumStatut)

Services (NumService , NomService , TarifService)

Proposer (NumIntervenant , NumService)

Statuts (NumStatut , NomStatut)

Les contraintes de clé primaire :

NumIntervenant est la clé primaire de Intervenants.

NumService est la clé primaire de Services.

(NumIntervenant , NumService) est la clé primaire de Proposer.

NumStatut est la clé primaire de Statuts.

Les contraintes d'intégrité référentielles :

Proposer.NumIntervenant référence **Intervenants.NumIntervenant**

Proposer.NumService référence **Services.NumService**

Intervenants.NumStatut référence **Statuts.NumStatut**

Les contraintes d'unicité :

Unicité de NomStatut

Unicité de NomService

Les contraintes existentielles :

Tous les attributs sont obligatoires.

A SQL.

Écrivez en SQL les requêtes permettant de répondre aux demandes suivantes.

1. Les intervenants (numéro, nom et prénom) proposant plus de 2 services. [/1]

2. Liste des services (numéro et nom du service) avec le nombre d'intervenants proposant chaque service (0 si le service n'est proposé par aucun intervenant). [/1]

3. Liste des services proposés exclusivement par des intervenants n'ayant pas le statut de salarié. [/1]

4. Liste des intervenants (numéro, nom et prénom) proposant le plus grand nombre de services. [/2]

B Visual Basic pour l'application Access.

Nous supposons dans un premier temps que la base de données est implantée sur le logiciel Access.

Le formulaire ci-dessous permet à partir de la sélection d'un intervenant d'afficher la liste des services qu'il propose. Il permet également d'ajouter un nouveau service à la liste des services proposés par cet intervenant et de supprimer un des services proposés par ce même intervenant.

Intervenant : VERGNOLE Camille

Nom : ListeIntervenants

Services proposés :

- soutien scolaire

Nom : ListeServicesProposés

Ajouter service

Supprimer service

Services non proposés :

- assistance administrative
- assistance informatique
- assistance personnes âgées
- assistance personnes handicapées
- bricolage
- entretien linge
- entretien maison
- garde d'enfants
- jardinage
- préparation repas
- soins d'esthétique
- soins et promenades animaux

Nom : ListeServicesNonProposés

Le choix d'une ligne dans la zone de liste déroulante " ListeIntervenants " entraîne la mise à jour des zones de listes affichant la liste des services proposés par cet intervenant (ListeServicesProposés) et la liste des services non proposés par cet intervenant (ListeServicesNonProposés).

1. Ecrivez les requêtes permettant de renseigner ces deux zones de liste. [/2]

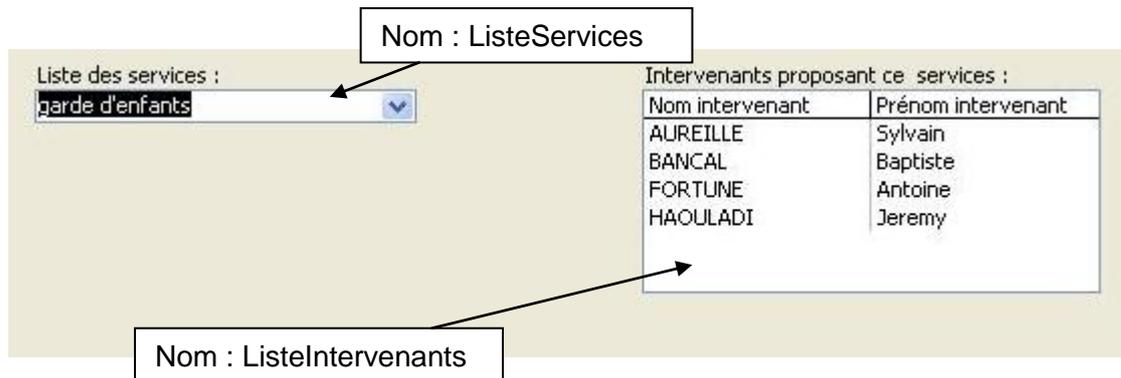
Voici le code VBA (volontairement non commenté) à compléter.

```
Private Sub listeIntervenants_Change()  
Dim labase As DAO.Database  
Dim leformulaire As Form  
Dim ctrlservicesProposés As Control  
Dim ctrlservicesNonProposés As Control  
Dim ctrlIntervenants As Control  
Dim requeteServicesProposés As String  
Dim requeteServicesNonProposés As String  
Set labase = CurrentDb  
Set leformulaire = Me  
Set ctrlservicesProposés = leformulaire.ListeServicesProposés  
Set ctrlservicesNonProposés = leformulaire.ListeServicesNonProposés  
Set ctrlIntervenants = leformulaire.ListeIntervenants  
requeteServicesProposés = .....  
.....  
.....  
ctrlservicesProposés.RowSource = requeteServicesProposés  
  
requeteServicesNonProposés = .....  
.....  
.....  
.....  
ctrlservicesNonProposés.RowSource = requeteServicesNonProposés  
End Sub
```

2. Complétez le code VBA ci-dessous (volontairement non commenté) associé au bouton "Ajouter service". [/2]

```
Private Sub BAjoutService_Click()  
Dim labase As DAO.Database  
Dim leformulaire As Form  
Dim ctrlservicesProposés As Control  
Dim ctrlservicesNonProposés As Control  
Dim ctrlIntervenants As Control  
Dim requeteServicesNonProposés As String  
Dim rstAjout As DAO.Recordset  
Set labase = CurrentDb  
Set leformulaire = Me  
Set ctrlservicesProposés = leformulaire.ListeServicesProposés  
Set ctrlservicesNonProposés = leformulaire.ListeServicesNonProposés  
Set ctrlIntervenants = leformulaire.ListeIntervenants  
  
If ctrlIntervenants <> -1 And ctrlservicesNonProposés <> -1 Then  
    Set rstAjout = labase.OpenRecordset("Proposer", dbOpenDynaset)  
    rstAjout.AddNew  
    rstAjout!NumIntervenant = ctrlIntervenants.Column(0, ctrlIntervenants.ListIndex)  
  
    .....  
  
    rstAjout.Update  
    rstAjout.Close  
    leformulaire.Refresh  
Else  
  
    .....  
  
End If  
End Sub
```

Le formulaire ci-dessous permet à partir de la sélection d'un service d'afficher la liste des intervenants proposant ce service.



3. Ecrire le code VBA permettant l'affichage des intervenants proposant le service sélectionné. [/2]

C SQL Server : programmation Transact-SQL..

La base de données a été transférée sur SQL Server.

1. Expliquez le programme Transact SQL (volontairement non commenté) ci-dessous. [/2]

```
declare @NumeroService int;
declare @NomService nchar(30);
declare Curseur1 cursor for
    select NumService , NomService
    from Services where NumService in
        (select distinct NumService from Proposer
         join Intervenants on Proposer.NumIntervenant = Intervenants.NumIntervenant
         join statuts on Intervenants.NumStatut = Statuts.NumStatut
         where NomStatut = 'salarié' and villeIntervenant = 'Bordeaux' )

open Curseur1
fetch next from curseur1 into @NumeroService , @NomService
while @@FETCH_STATUS = 0
    begin
        print cast(rtrim(cast(@NumeroService as nchar)) as nchar(5)) + ' ' + @NomService
        fetch next from curseur1 into @NumeroService , @NomService
    end
close Curseur1
deallocate Curseur1
go
```

2. **Ecrivez un programme Transact SQL qui affiche la liste des intervenant regroupés par service [/4]**
Présentation souhaitée.

1	assistance administrative
1	AUREILLE Sylvain
2	BABEL Raphael
5	CORNEAU Jahson
5	assistance informatique
1	AUREILLE Sylvain
8	DRODE Samuel
13	GENINATTI Benjamin
20	ORTOLAN Florian
2	assistance personnes âgées
9	FORTUNE Antoine
11	GAROT Emeline
...	

D SQL Server : programmation Visual Studio.

1. Expliquez la procédure Visual Basic (volontairement non commentée) ci-dessous. [/1]

```
Sub procedure1(ByVal numero As Integer)
    Dim connexion As String = "Data Source=leServeur;Database=adomicile;Integrated Security=true"
    Using connection As New SqlClient.SqlConnection(connexion)
        Dim req As String = "select * from Intervenants where NumIntervenant =" & Str(numero) & " ;"
        Dim commd As New SqlClient.SqlCommand(req, connection)
        connection.Open()
        Dim unDataReader As SqlClient.SqlDataReader = commd.ExecuteReader()
        If unDataReader.HasRows() Then
            Console.WriteLine(String.Format(vbCrLf & "Informations sur numéro {0}", numero))
            unDataReader.Read()
            Console.WriteLine(String.Format(" InformationA : {0}", unDataReader (1)))
            Console.WriteLine(String.Format(" InformationB: {0}", unDataReader (2)))
            Console.WriteLine(String.Format(" InformationC: {0}", unDataReader (6)))
        Else
            Console.WriteLine(String.Format(vbCrLf & "Il n'y a pas de numéro {0}", numero))
        End If
        unDataReader.Close()
    End Using
    Console.WriteLine(String.Format("Tapez [Entrée] pour continuer"))
    Console.Read()
End Sub
```

2. **Ecrivez la procédure Visual Basic qui affiche le nombre d'intervenants proposant le service dont le numéro est passé en paramètre. La procédure doit afficher un message d'information si aucun intervenant ne propose ce service ou si le numéro du service saisi est inexistant. [/2]**