

I. Script Transact-SQL

1) SQL et Transact-SQL

Pour travailler avec les bases de données de façon optimale, SQL Server, comme les autres SGBDR, s'appuie massivement sur le langage SQL. Pour accroître les possibilités de traitement des données au niveau du serveur, SQL Server propose le langage Transact-SQL qui apporte des possibilités de langage procédural au langage SQL qui est ensembliste. Ces deux langages sont étroitement liés au sein de SQL Server, mais il est important de souligner que le langage Transact-SQL est spécifique à SQL Server.

Transact-SQL est au centre de l'utilisation de SQL Server car toutes les applications qui communiquent avec une instance de SQL Server le font en envoyant au serveur des instructions Transact-SQL, quelle que soit l'interface utilisateur de l'application.

2) Script SQL sous SQL Server Management Studio

Pour créer un script SQL avec SQL Server Management Studio, il faut faire apparaître la fenêtre permettant la saisie et l'exécution de scripts. Plusieurs possibilités sont offertes comme le menu

Fichier / Nouveau / Requête avec la connexion actuelle, ou bien le raccourci-clavier Ctrl-N, ou bien encore le bouton **Nouvelle requête** présent dans la barre d'outils de SQL Server Management Studio.

II. Création de bases de données en SQL

1) Création de bases de données

A l'aide de requêtes SQL, créer les bases de données **carnet_adresses** et **test**.

Remarques :

- Après chaque exécution de requête, vérifier le bon déroulement dans la fenêtre **Messages**.
- Sauvegarder les requêtes dans un fichier nommé **tp03_nombinome1-nombinome2.sql**.

2) Suppression de bases de données

Donner la requête SQL permettant de supprimer la base **test**.

III. Base de données carnet_adresses

Soit la base de données **carnet_adresses** dont le schéma logique est présenté **Figure 1**.

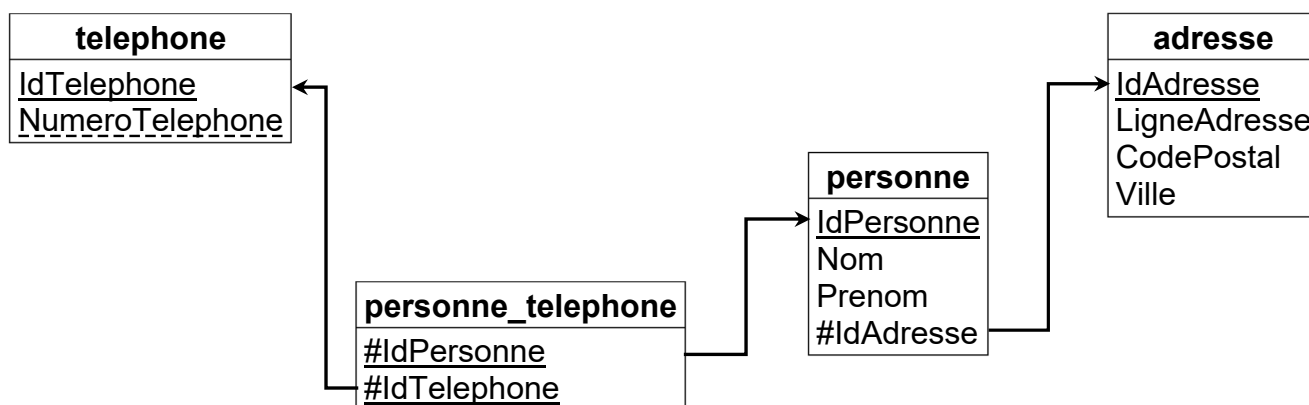


Figure 1 : Schéma logique de la base de données carnet_adresses

Ce schéma logique a en fait été obtenu après normalisation de la base de données qui ne comportait à l'origine qu'une seule table dont le schéma de relation était donné par :

carnet_adresses (PrenomNom, Adresse, Telephone)

et dont le contenu est présenté **Figure 2**.

PrenomNom	Adresse	Telephone
Pierre Girard	72 rue de la Gare, 69001 Lyon	0654201645
Jean Vigneau	18 rue Faraday, 59000 Lille	
Gérard Mercier	25 rue Lemaitre, Lyon 69003	0643311305
Pierre Girard	72 rue de la Gare, 69001 Lyon	0654201527
Maurice Avron	8 chemin de Cluzel, 31000, Toulouse	0922584500
Thierry Vilminot	3 avenue des Roses, 69001 Lyon	
Claudine Monti	112 rue Neuve, Nice 06000	0622964525
Gérard Mercier	14 place de la gare, 35000 Rennes	0650542335
Catherine Avron	8 chemin de Cluzel, 31000, Toulouse	0922584500
Pierre Ferard	65 rue du Touffenet, 86000, Poitiers	+33654201600

Figure 2 : Contenu de la base de données **carnet_adresses**

Remarque : seule l'adresse principale de chaque personne est indiquée dans cette table. Gérard Mercier qui habite Lyon et Gérard Mercier qui habite Rennes sont donc deux personnes différentes.

IV. Création des tables et manipulations des données de la base carnet_adresses

1) Création des tables de la base de données carnet_adresses

Après avoir déterminé les types des différentes données, donner les requêtes SQL permettant de créer les tables de la base de données **carnet_adresses**.

Remarque : les valeurs des identifiants **IdTelephone**, **IdPersonne** et **IdAdresse** des tables **telephone**, **personne** et **adresse** seront générés par auto-incrémentation en utilisant la propriété SQL **IDENTITY**.

2) Chargement des tables adresse et telephone

Charger les tables **adresse** et **telephone** de la base de données **carnet_adresses** à partir des fichiers Excel **adresse.xls** et **telephone.xls**.

(cliquer sur **Tâches / Importer des données...** du menu contextuel de la bases de données **carnet_adresses** en choisissant **Micosoft Excel** comme type de **Source de données** et **SQL Server Native Client 11.0** comme **Destination**).

3) Chargement des tables personne et personne_telephone

Donner les requêtes SQL permettant de de charger les tables **personne** et **personne_telephone** de la base de données **carnet_adresses**.

Remarque : l'écriture des requêtes d'insertion des données pourra être simplifiée par copier-coller des valeurs à partir du fichier Excel **carnet_adresses.xls** qui contient les données de la table **carnet_adresses** d'origine.

4) Saisie de nouvelles personnes dans la base de données carnet_adresses

Donner les requêtes SQL permettant de saisir les personnes suivantes dans la base de données **carnet_adresses** :

PrenomNom	Adresse	Telephone
Théo Avron	8 chemin de Cluzel, 31000, Toulouse	0922584500
Carole Tortua	7 rue Verget, 75005 Paris	
Benoît Tortua	7 rue Verget, Paris 5e	0686914988

5) Requêtes d'extraction de données de la base de données carnet_adresses

Donner les requêtes SQL permettant de répondre aux questions suivantes :

- a) Afficher les identifiants, les noms et les prénoms des personnes référencées dans la base de données **carnet_adresses**.
- b) Afficher les identifiants et les noms des personnes qui se prénomment Pierre.
- c) Afficher, triée par ordre alphabétique, la liste des villes référencées dans la table **adresse**.
- d) Afficher les identifiants, les noms, les prénoms et les villes des personnes référencées dans la base de données **carnet_adresses**.
- e) Afficher les noms des personnes qui habitent à Lyon ou à Paris.

V. Sauvegarde et suppression de la base de données carnet_adresses

- 1) A partir du menu contextuel de la base de données **carnet_adresses**, cliquer sur *Tâches / Sauvegarder...* afin de sauvegarder la base dans un format (**.bak**) permettant de la restaurer ensuite (choix *Restaurer la base de données...* du menu contextuel **Bases de données**).
- 2) Supprimer la base de données **carnet_adresses** (cliquer sur *Supprimer* dans le menu contextuel de la base de données concernée).