

Tp : Transact-sql

Durée : 3h

Mode d'authentification :

Authentification Windows : Si vous choisissez ce mode d'authentification, cela veut dire que le serveur de base de données, lorsque vous essayez de vous connecter, ne vous demandera pas de mot de passe. Utiliser ce mode d'authentification si vous n'avez pas de compte sur le serveur de base de données. C'est avec ce mode que l'on se connecte pour la première fois.

Authentification SQL server : Si vous choisissez ce mode d'authentification, cela veut dire que vous avez un compte sur le serveur de bases de données. Vous avez besoin d'un nom d'utilisateur, d'un mot de passe et d'une base de données. C'est ce mode d'authentification que l'on va utiliser durant toute la session. C'est ce mode d'authentification que vous allez avoir en entreprise.

Étape 1 : Changer le mode d'authentification

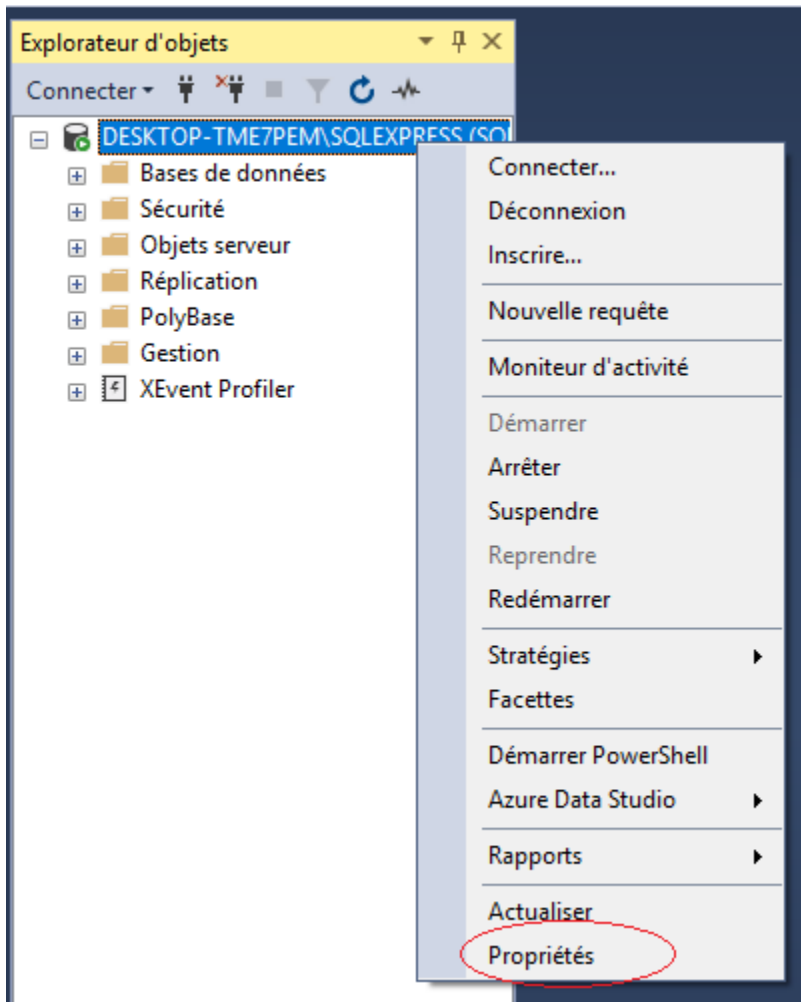
Lorsqu'on établit une connexion pour la première fois, nous allons faire une authentification Windows. (Vous n'avez pas encore de compte sur le Serveur SQL server) — voir la figure suivante.

Attention :

Le nom du serveur est le nom de votre ordinateur\nnom de l'instance

Sinon, dérouler le nom du serveur, faire parcourir (ou <Browse for more ...> et trouver votre serveur et son instance.

Une fois que vous êtes connecté, allez sur les propriétés de votre connexion et changez le mode d'authentification. → Figure suivante.



Bouton droit sur votre serveur, puis propriété **Sécurité**

A l'onglet Sécurité, choisir Mode d'authentification SQL Server et Windows. Faites OK. Redémarrer le serveur. (Bouton droit puis redémarrer.).

Sélectionner une page

- Général
- Mémoire
- Processeurs
- Sécurité**
- Connexions
- Paramètres de base de données
- Avancé
- Autorisations

Script ? Aide

Authentification du serveur

☐ Mode d'authentification Windows

☒ Mode d'authentification SQL Server et Windows

Audit de connexion en cours

☐ Aucune

☒ Échecs de connexion uniquement

☐ Réussites de connexion uniquement

☐ Échecs et réussites de connexion

Compte proxy du serveur

☐ Activer le compte proxy du serveur

Compte proxy :

Mot de passe :

Options

☐ Activer le suivi d'audit C2

☐ Chaînage des propriétés des bases de données croisées

Connexion

Serveur : M-INFO-SY\SQLEXPRESS2017

Connexion : CLG\Saliha.Yacoub

[Afficher les propriétés de connexion](#)

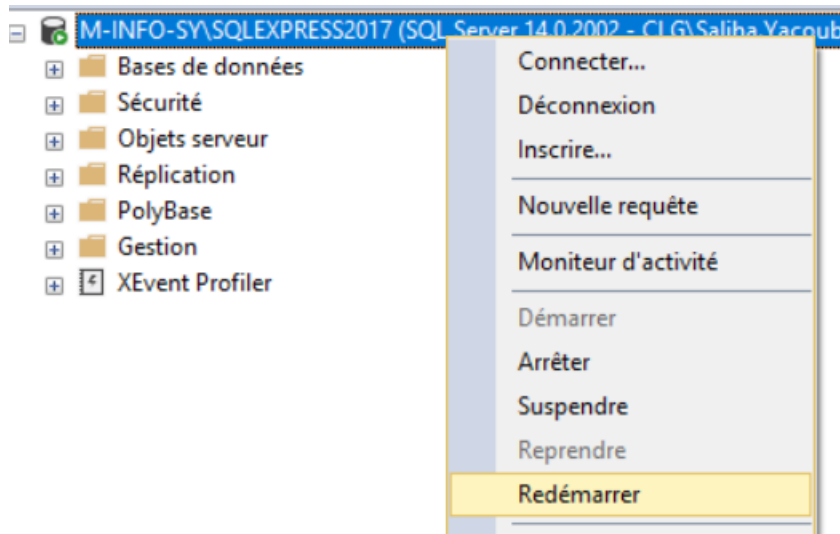
Progression

Prêt

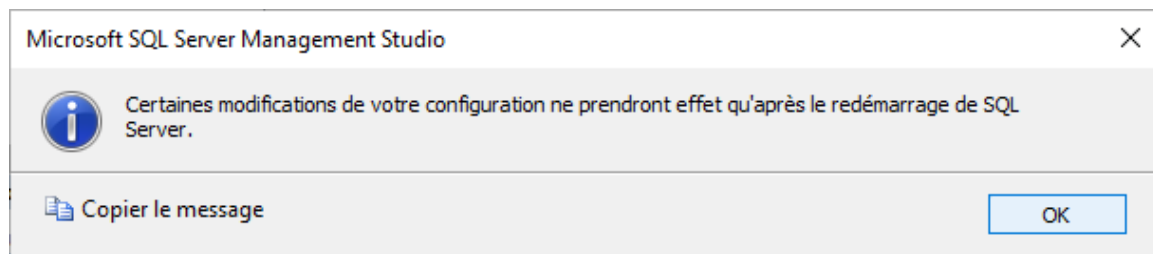
OK Annuler

Attention : 

Vous devez redémarrer le serveur

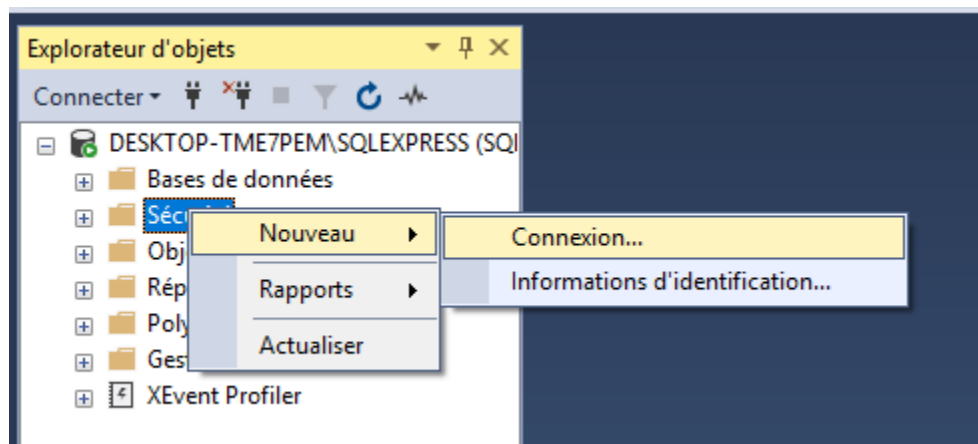


Il est probable que le serveur vous fasse une mise en garde quant au changement du mode d'authentification et qu'il faut redémarrer le serveur. Faîtes juste OK.



Étape 2 : Créer une nouvelle connexion

Sur le bouton droit de l'onglet **Sécurité**, créer une nouvelle connexion.



- Donner un nom significatif sans caractères spéciaux et sans accents
- Choisir Authentification SQL Server.
- Choisir un mot de passe qui respecte la stratégie des mots de passe Windows Server

Sélectionner une page

- Général
- Rôles du serveur
- Mappage d'utilisateur
- Éléments sécurisables
- État

Script ? Aide

Nom d'accès :

☐ Authentification Windows
☒ **Authentification SQL Server**

Mot de passe :

Confirmer le mot de passe :

☐ Spécifier l'ancien mot de passe

Ancien mot de passe :

☒ Appliquer la stratégie de mot de passe
☒ Appliquer l'expiration du mot de passe
☐ L'utilisateur doit changer de mot de passe à la prochaine connexion

☐ Mappé au certificat
☐ Mappé à la clé asymétrique
☐ Mapper aux informations d'identification

Informations d'identification mappées

Informations ...	Fournisseur

Base de données par défaut :

Langue par défaut :


Connexion

Serveur : DESKTOP-TME7PEM\SQLEXPRESS

Connexion : DESKTOP-TME7PEM\Travail

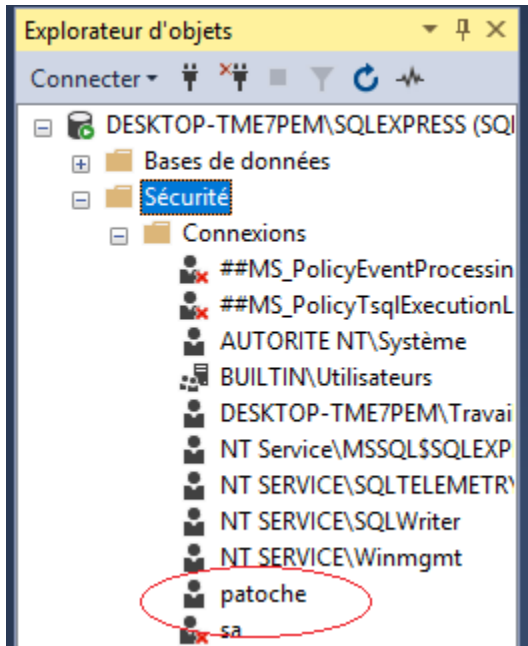
[Afficher les propriétés de connexion](#)

Progression

 Prêt

- Décocher l'utilisateur doit changer le mot de passe.
- Vous pouvez décocher la case « Conserver la stratégie des mots de passe. Mais ce n'est pas conseillé.
- Comme vous n'avez pas de base de données, la connexion utilise la Base de données par défaut qui **master**.
- Ne vous inquiétez pas, vous aller avoir votre propre base de données


Une fois que cette étape est terminée, vérifiez que votre connexion est bien créée. Pour cela allez dans l'onglet Sécurité puis Connexion et repérez votre connexion



Étape 3 : Attribuer les rôles

Pour pouvoir créer votre propre base de données vous devez posséder les droits nécessaires (ou le rôle).

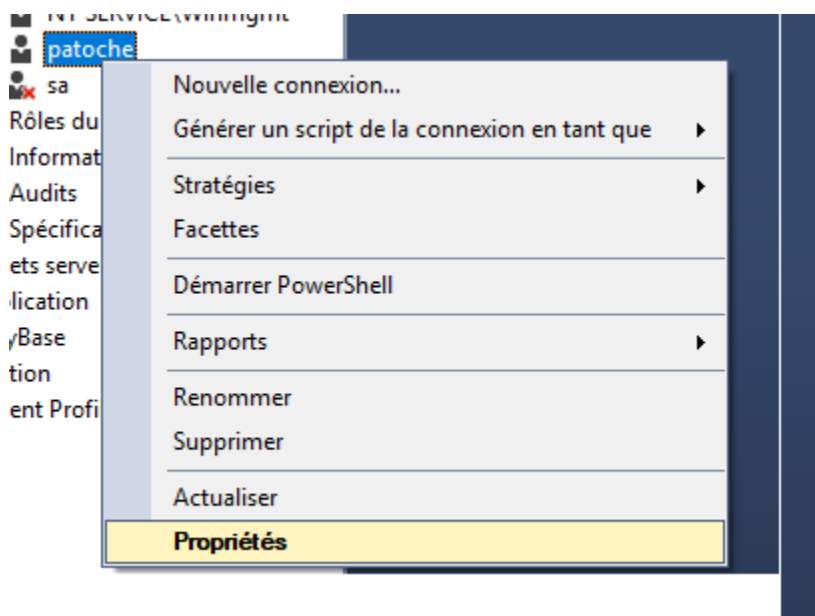
Si vous êtes administrateur alors vous avez déjà ces rôles, sinon vous devez les attribuer à votre connexion avant de créer la bd.

Attention : 

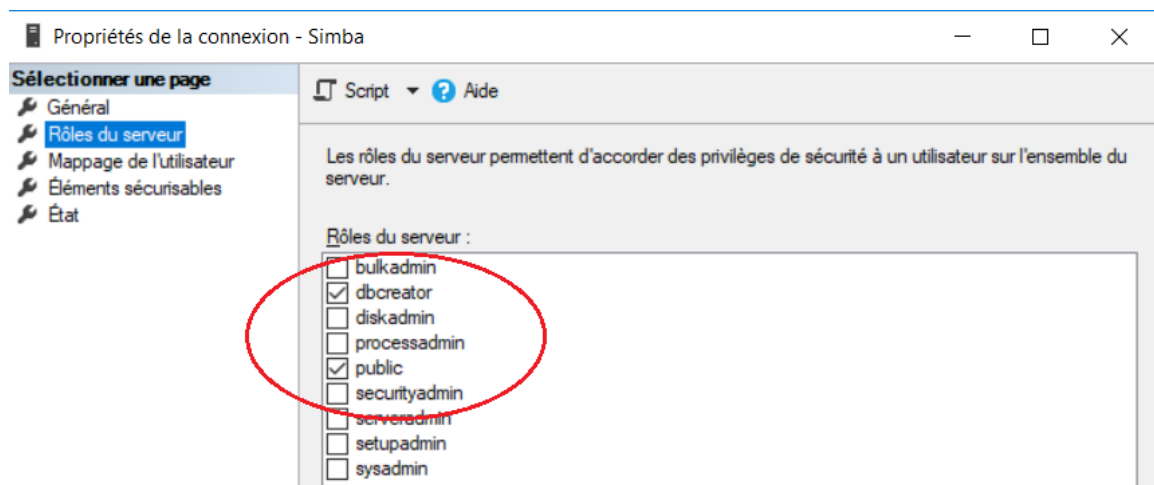
Pour créer la base de données vous devez avoir au moins le rôle **dbcreator**

Les membres du rôle de serveur **dbcreator** peuvent créer, modifier, supprimer et restaurer n'importe quelle base de données.

Pour donner les droits à votre connexion, allez à votre connexion, bouton droit, propriétés puis rôle du serveur



Puis Rôles du serveur.



Puis cocher **dbcreator** puis cliquer sur OK.

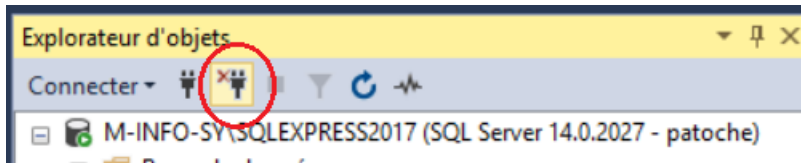
Attention :

Ne jamais donner le rôle sysadmin. Les membres du rôle sysadmin peuvent effectuer toute activité sur le serveur. Faites attention !!

Étape 4, Connexion avec l'authentification SQL Server et création de la base de données.

Vous pouvez vous déconnecter du serveur et vous reconnecter avec votre nouvelle connexion (SQL Server) comme suit.

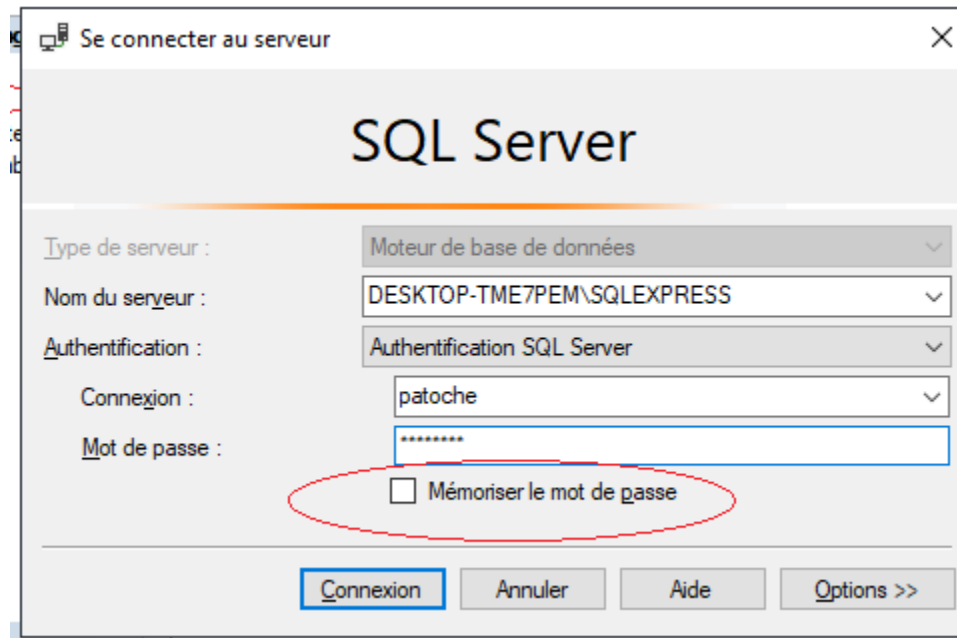
Pour vous déconnecter du serveur, utiliser le bouton Déconnecter



Ou bien Bouton droit sur la Votre serveur, puis **déconnecter** .

Attention :

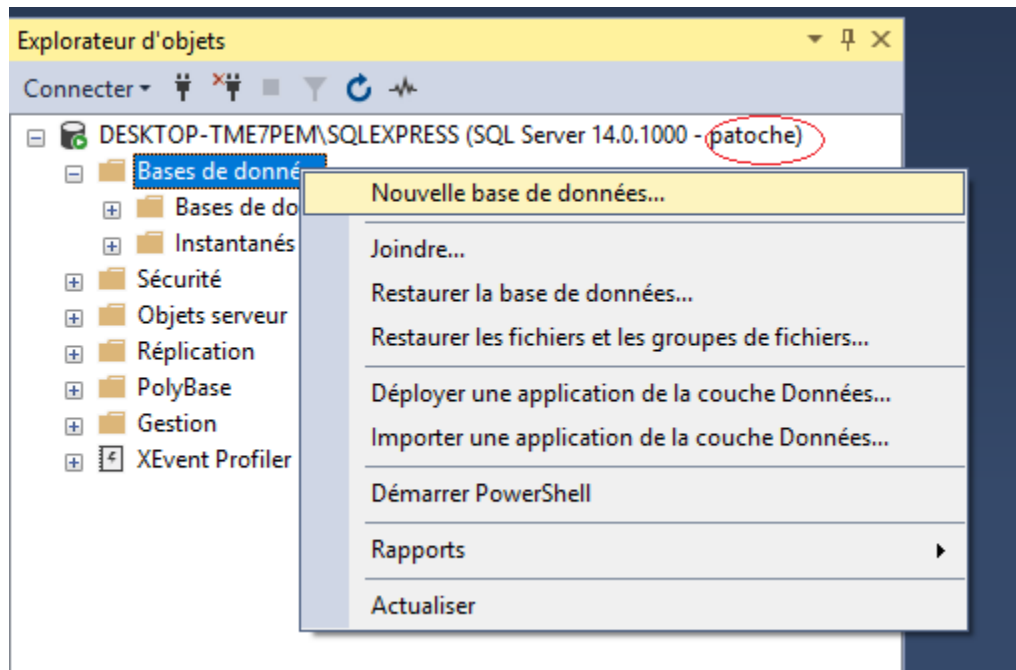
Ne jamais mémoriser le mot de passe



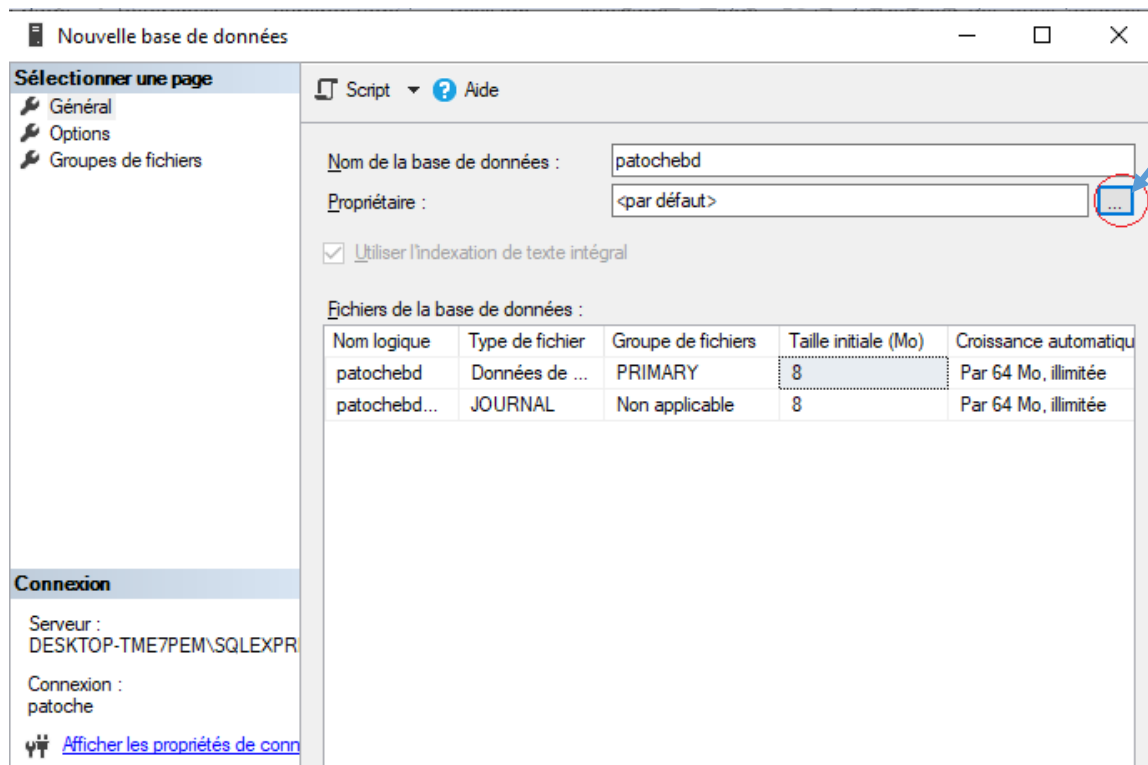
Attention :

Ne jamais mémoriser le mot de passe

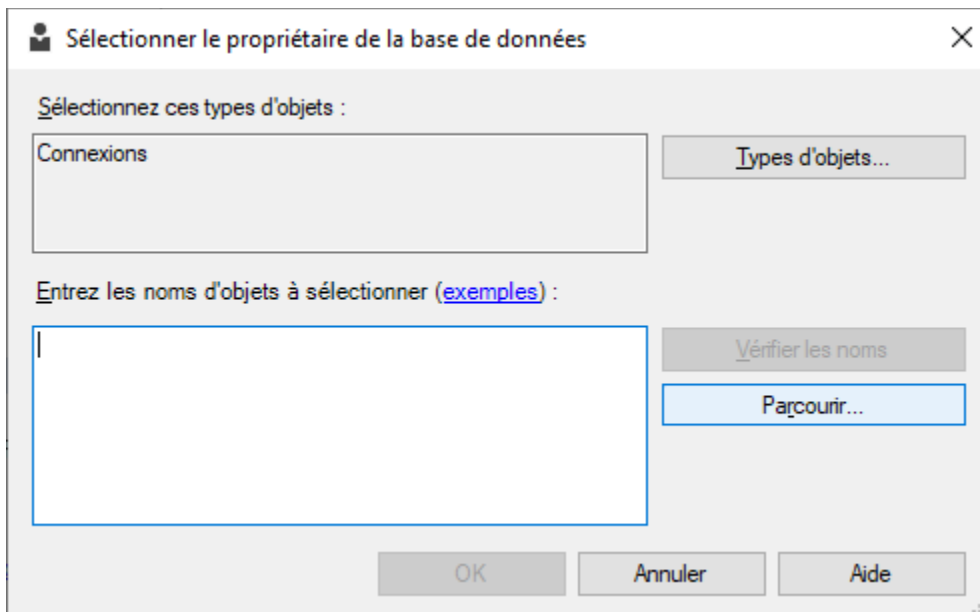
Pour créer une nouvelle base de données, placez-vous à l'onglet bases de données, puis nouvelle base de données. Ou utiliser la commande CREATE DATABASE



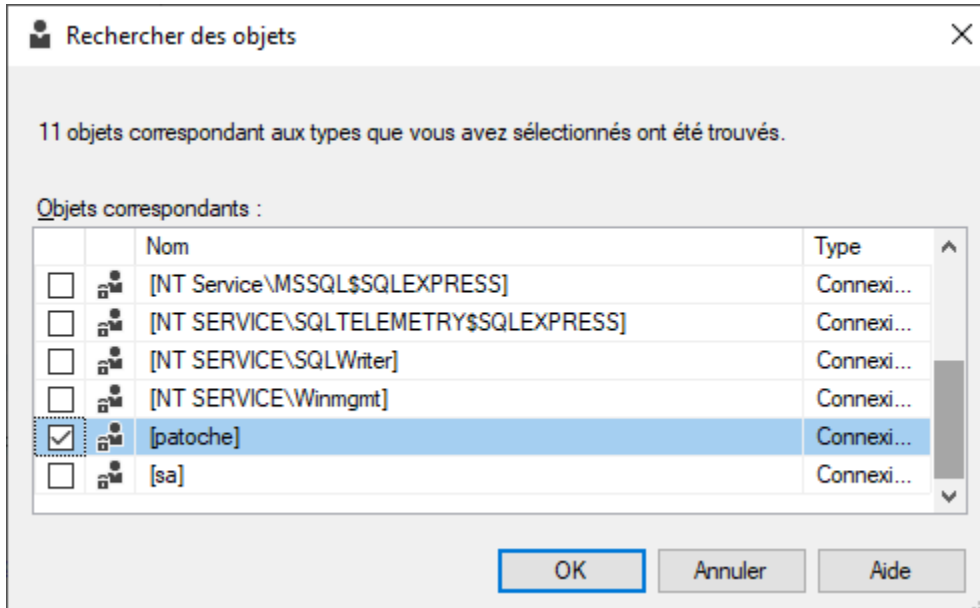
Donnez un nom significatif à votre base de données.



Avant de cliquer sur OK, cliquer sur propriétaire, vous allez avoir la figure suivante :



Cliquez ensuite sur parcourir, puis trouvez votre connexion et cochez-la. (voir figure suivante).



Cliquez OK sur chaque fenêtre

Après la création de la base de données, nous allons faire en sorte que le login pointe directement sur la nouvelle base de données.

Sous l'onglet Sécurité, déroulez les connexions. Repérez la vôtre. Puis bouton droit de la souris et Propriétés.

Rôles du serveur
Mappage d'utilisateur
Éléments sécurisables
État

Connexion

Serveur :
DESKTOP-TME7PEM\SQLEXPRESS

Connexion :
patoche

[Afficher les propriétés de connexion](#)

Progression

Prêt

Nom d'accès : patoche Rechercher...

☐ Authentification Windows

☒ Authentification SQL Server

Mot de passe :

Confirmer le mot de passe :

☐ Spécifier l'ancien mot de passe

Ancien mot de passe :

☒ Appliquer la stratégie de mot de passe

☒ Appliquer l'expiration du mot de passe

☐ L'utilisateur doit changer de mot de passe à la prochaine connexion

☐ Mappé au certificat

☐ Mappé à la clé asymétrique

☐ Mapper aux informations d'identification

Informations d'identification mappées

Informations ...	Fournisseur
------------------	-------------

Base de données par défaut : patochebd

Langue par défaut :

master

model

msdb

patochebd

tempdb

Ajouter

Supprimer

Annuler

Choisir ensuite le nom de votre BD par défaut. Tester à nouveau votre connexion.

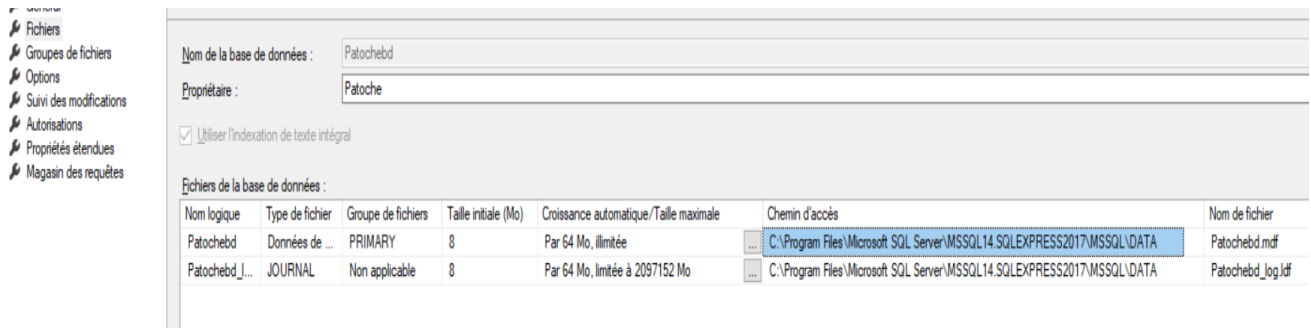
Important :

Vous pouvez également créer votre base de données avec la commande CREATE

```
CREATE DATABASE nomde laBD;
```

Où est stockée la base de données ?

En cliquant sur le bouton droit de votre base de données, puis propriétés à l'onglet fichier vous allez trouver les deux fichiers de la bd et leur emplacement.



Ces fichiers sont dans :

C:\Program Files\Microsoft SQL Server\MSSQL14.SQLEXPRESS2017\MSSQL\DATA

Vous y trouverez deux types de fichiers pour chaque BD : L'un **.mdf** et l'autre **.ldf**

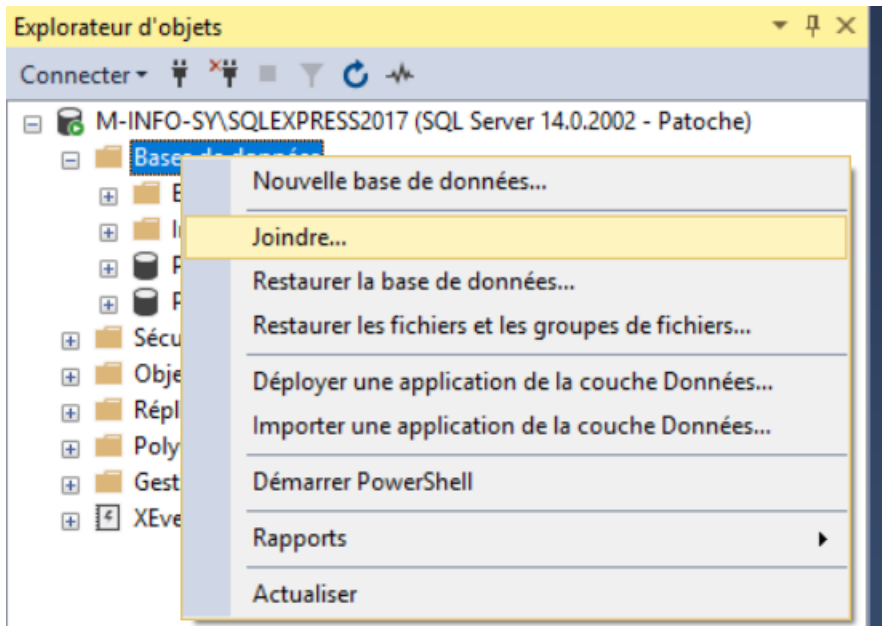
Les données sont stockées dans un fichier MDF, toutes les transactions, les modifications de la base de données SQL Server effectuées par chaque transaction sont stockées dans un fichier LDF

Patochebd.mdf et Patochebd_log.ldf

📁 Ce PC > Windows (C:) > Programmes > Microsoft SQL Server > MSSQL14.SQLEXPRESS2017 > MSSQL > DATA

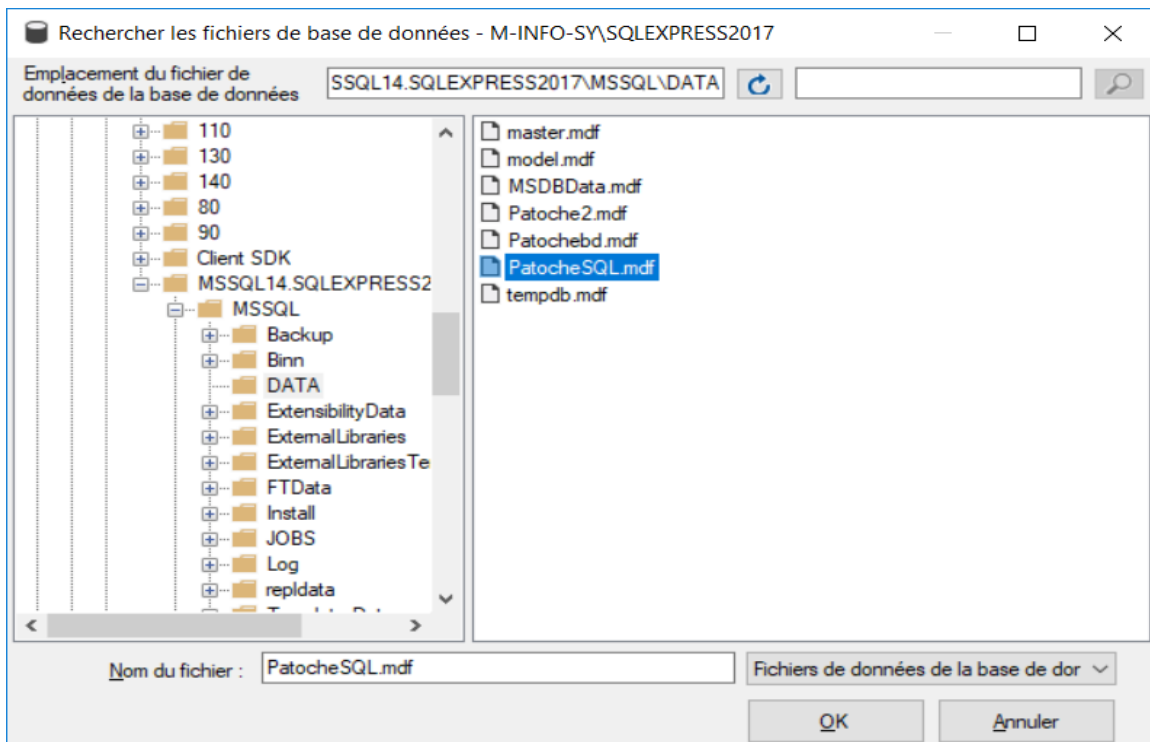
	Nom	Modifié le	Type	Taille
Je	mas:er	2019-07-25 22:48	SQL Server Database	5 504 Ko
	mas:log	2019-07-25 22:48	SQL Server Database	2 048 Ko
ements	mocel	2019-07-25 22:48	SQL Server Database	8 192 Ko
its	mocellog	2019-07-25 22:48	SQL Server Database	8 192 Ko
	MS_AgentSigningCertificate	2018-09-10 10:49	Certificat de securi...	1 Ko
	MSDDBData	2019-07-25 22:48	SQL Server Database	18 240 Ko
ile	MSDDBLog	2019-07-25 22:48	SQL Server Database	29 504 Ko
ile-v1	Patoche2	2019-07-25 22:48	SQL Server Database	8 192 Ko
	Patoche2_log	2019-07-25 22:48	SQL Server Database	8 192 Ko
	Patochebd	2019-07-25 22:48	SQL Server Database	8 192 Ko
	Patochebd_log	2019-07-25 22:48	SQL Server Database	8 192 Ko
Actualisation 2015-2017	PatocheSQL	2019-07-25 22:48	SQL Server Database	8 192 Ko
tion 2015-2017	PatocheSQL_log	2019-07-25 22:48	SQL Server Database	8 192 Ko
ha	tempdb	2019-07-28 10:58	SQL Server Database	8 192 Ko
	templog	2019-07-28 10:58	SQL Server Database	8 192 Ko

Vous pouvez récupérer votre base en faisant : Bouton droit sur Bases de données, puis **Joindre**. Vous aurez la fenêtre suivante.



Cliquer sur le bouton Ajouter. Choisir le fichier en question (le fichier.mdf) puis faire OK, puis OK.

Votre base de données va apparaître dans l'explorateur d'objets. Vous pouvez alors l'exploiter comme vous voulez.



Attention : 

Une base de données ne peut être jointe plus qu'une fois.

Lorsque vous essayez de joindre une Base de données déjà jointe, cela provoquera une erreur.

Il faut que vos fichiers soient dans le bon dossier.

Attention Récupération de la base de données: 

Pour récupérer votre base de données effectuer les étapes suivantes :

1- Dans tous les cas garder vos scripts SQL.

2-Copier les deux fichiers .mdf et .ldf de votre base de données (patochebd et patoche_log) dans votre clé USB

3-pour ouvrir les fichiers que vous avez copiés dans votre clé USB sur un autre ordinateur, vous devez d'abord JOINDRE le fichier.mdf

4-Si vous êtes certains que votre BD a été copiée proprement alors vous pouvez la supprimer pour qu'il n'y ait pas de plagiat.

Chapitre 3, création des tables

Types de données SQL Server

Types numériques exacts

Type	À partir de	À
bigint	-9.223.372.036.854.775.808	9.223.372.036.854.775.807
int	-2147483648	2147483647
smallint	-32768	32767
tinyint	0	255
bit	0	1
Decimal	-10^{38}	$10^{38}-1$
numeric	-10^{38}	$10^{38}-1$
money	-922,337,203,685,477.5808	+922,337,203,685,477.5807
smallmoney	-214,748.3648	+214,748.3647

Numerics approximatif

Type	À partir de	À
float	$1,79 \times 10^{-308}$	$1,79 \times 10^{308}$
reel	-3.40×10^{38}	3.40×10^{38}

datetime et smalldatetime

Type	À partir de	À
datetime (3,33 exactitude millisecondes)	1 janvier 1753	31 déc 9999
smalldatetime (précision de 1 minute)	1 janvier 1900	6 juin 2079

Chaînes de caractères

Type	Description
char	De longueur fixe de caractères Unicode avec une longueur maximum de 8000 caractères.
varchar	Texte unicode de longueur variable allant jusqu'à 2 Go.
text	Texte non unicode de longueur maximale 2Go

Les chaînes de caractères Unicode

Type	Description
nchar	la longueur de données Unicode-fixe avec une longueur maximale de 4000 caractères.
nvarchar	la longueur de données Unicode et variable, avec une longueur maximum de 4000 caractères.
nvarchar (max)	longueur Unicode données variables avec une longueur maximale de 230 caractères (SQL Server 2005 uniquement).
ntext	la longueur de données Unicode et variable, avec une longueur maximale de 1073741823 caractères.

Binary Cordes

Type	Description
binaire	De longueur fixe des données binaires d'une longueur maximale de 8000 octets.
varbinary	De longueur variable des données binaires d'une longueur maximale de 8000 octets.
varbinary (max)	De longueur variable des données binaires d'une longueur maximale de 231 octets (SQL Server 2005 uniquement).
image	De longueur variable des données binaires d'une longueur maximale de 2147483647 octets.

Pour plus de détails, allez sur :

<https://docs.microsoft.com/fr-fr/sql/t-sql/data-types/data-types-transact-sql?view=sql-server-2017>

La propriété « IDENTITY » d'une table

Vous pouvez mettre en œuvre des colonnes d'identification à l'aide de la propriété IDENTITY. Ce qui permet de réaliser un auto incrément sur une colonne.

En général, la propriété IDENTITY se définit sur la clé primaire.

Si la propriété IDENTITY est définie sur une colonne, alors c'est le système qui insère des données dans cette colonne (pas l'utilisateur).

Attention :



Vous ne pouvez pas modifier une colonne existante pour y ajouter la propriété IDENTITY.

Exemple1 :

```
create table Eleves
(
  num int identity,
  nom varchar(30),
  prenom varchar(30),
  constraint pkeleve primary key(num));

insert into Eleves (nom,prenom) values ('Patoche', 'Alain');
insert into Eleves (nom,prenom) values ('Simba', 'Chat');
```

Remarquez :

1. La colonne num a la propriété IDENTITY.
2. Nous n'avons pas inséré dans la colonne num. C'est le système qui le fait pour nous.
3. Le num de Patoche sera 1, le num de Simba sera 2 et ainsi de suite.

Lorsque vous utilisez la propriété IDENTITY pour définir une colonne d'identification, tenez compte des éléments suivants :

- Une table ne peut comprendre qu'une colonne définie à l'aide de la propriété IDENTITY, et cette colonne doit être définie à l'aide d'un type de données **decimal**, **int**, **numeric**, **smallint**, **bigint** ou **tinyint**.
- Vous pouvez spécifier la valeur de départ et l'incrément. La valeur par défaut est 1 dans les deux cas.
- La colonne d'identification ne doit ni accepter les valeurs NULL, ni contenir une définition ou un objet DEFAULT. **En général c'est la clé primaire.**
- La colonne peut être référencée dans une liste de sélection par l'emploi du mot clé \$IDENTITY après la définition de la propriété IDENTITY. La colonne peut également être référencée par son nom.
- SET IDENTITY_INSERT ON peut être utilisé pour désactiver la propriété IDENTITY d'une colonne en activant les valeurs à insérer explicitement.

Exemple 2

La numérotation automatique commence à 10 et elle est à pas de 2.

```
create table ClientsInfo  
(  
  numcl smallint IDENTITY(10,2),  
  nomcl varchar(30),  
  constraint pkcl primary key (numcl)  
);  
  
insert into Clientsinfo (nomcl) values ('Gavroche');
```

Si on veut faire une insertion manuelle dans la colonne num (IDENTITY) il faut mettre IDENTITY_INSERT a ON

Exemple 3 :

```
set identity_insert eleves on;  
  
insert into eleves (num,nom, prenom) values  
(20, 'Simpson', 'Fred');
```

Attention, le num de Simpson est 20

Si on veut revenir à l'incrémementation automatique il faut faire

```
set identity_insert eleves off;  
  
insert into eleves(nom, prenom) values ('Simon', 'Pascal');
```

Attention: le num de Simon est 21.

Création des tables avec SQL Server

Les mêmes syntaxes s'appliquent à la création de tables avec SQL Server.

Exemple1 : Dans l'exemple qui suit remarquez la colonne **identity** et le type **money**

```
create table programmes
(codep char(3),
nomprogramme varchar(30),
constraint pkprg primary key(codep)
);

create table etudiants
(
numad int identity ,
nom varchar(20),
prenom varchar(30),
salaire money,
codep char(3),
constraint fkprg foreign key(codep)
references programmes(codep),
constraint pktudiant primary key(numad)
);
```

Remarque1 : Avec SQL Server on peut définir un seul INSERT INTO et donner la liste des valeurs.

```
insert into programmes values
('inf', 'Informatique'),
('tge', 'Technique de genie'),
('ele', 'Electronique'),
('sim', 'Sciences maths info');
```

Pour exécuter un commit après une opération DML, il faut mettre l'opération entre

```
begin transaction;
opérations DML
commit;
```

Exemple 2

```
begin transaction;  
insert into etudiants (nom, prenom, salaire, codep) values  
('Patoche', 'Alain', 12.33, 'tge');  
insert into etudiants (nom, prenom, salaire, codep) values  
('Gavroche', 'Miserable', 1.33, 'inf');  
insert into etudiants (nom, prenom, salaire, codep) values  
('Bien', 'Henry', 18.33, 'inf');  
insert into etudiants (nom, prenom, salaire, codep) values  
('Leriche', 'Alain', 40.00, 'inf');  
commit;
```

En général, l'ensemble des contraintes que nous avons vues avec ORACLE se définissent de la même façon avec SQL Server.

```
alter table etudiants add constraint  
cksal check(salaire>1);
```