



Ressourcesinformatiques

 + QUIZ

Version numérique

**OFFERTE!**

[www.editions-eni.fr](http://www.editions-eni.fr)

# SQL Server 2019

## SQL, Transact-SQL

Conception et réalisation  
d'une base de données

(avec exercices pratiques et corrigés)

En téléchargement

 code SQL des exemples

 corrigés des exercices

Hervé BOISGONTIER  
Jérôme GABILLAUD



Les éléments à télécharger sont disponibles à l'adresse suivante :  
**<http://www.editions-eni.fr>**  
Saisissez la référence ENI de l'ouvrage **RI19SQL** dans la zone de recherche et validez. Cliquez sur le titre du livre puis sur le bouton de téléchargement.

## Avant-propos

### Chapitre 1

#### Le modèle relationnel

1. Introduction . . . . .	15
1.1 Qu'est-ce qu'une base de données ? . . . . .	15
1.2 Qu'est-ce qu'un système de gestion de bases de données ? . . . . .	16
1.3 Qu'est-ce que le langage SQL ? . . . . .	17
1.4 Qu'est-ce que le langage Transact-SQL ? . . . . .	17
1.5 Qu'allez-vous apprendre à réaliser avec cet ouvrage ? . . . . .	17
2. Les tables . . . . .	18
2.1 Les attributs . . . . .	18
2.2 Les enregistrements . . . . .	19
2.3 La représentation d'une table . . . . .	19
2.3.1 La forme condensée . . . . .	19
2.3.2 La forme étendue . . . . .	19
2.3.3 La forme textuelle . . . . .	20
3. L'identification d'un enregistrement . . . . .	20
3.1 Clé primaire . . . . .	21
3.2 Clé secondaire . . . . .	22
4. La nullité . . . . .	23
5. Les domaines de valeurs . . . . .	24
6. Les contraintes d'intégrité référentielle et les clés étrangères . . . . .	25

7.	La normalisation des tables . . . . .	28
7.1	La première forme normale . . . . .	28
7.2	La deuxième forme normale . . . . .	29
7.3	La troisième forme normale . . . . .	31
8.	Le schéma relationnel de la base de données et le dictionnaire des données . . . . .	32
9.	L'algèbre relationnelle . . . . .	34
9.1	Les opérateurs . . . . .	34
9.1.1	L'union . . . . .	35
9.1.2	L'intersection . . . . .	36
9.1.3	La différence . . . . .	36
9.1.4	La division . . . . .	36
9.1.5	La restriction . . . . .	38
9.1.6	La projection . . . . .	39
9.1.7	Le produit cartésien . . . . .	40
9.1.8	Les jointures . . . . .	42
9.1.9	Les calculs élémentaires . . . . .	48
9.1.10	Les calculs d'agrégat . . . . .	49
9.2	Les étapes de résolution d'un problème . . . . .	52
9.2.1	La forme de la table résultante . . . . .	52
9.2.2	Les tables concernées . . . . .	52
9.2.3	Les attributs concernés . . . . .	52
9.2.4	L'ordonnancement des opérations . . . . .	53
9.2.5	Un exemple plus complexe . . . . .	56
10.	Exercices . . . . .	59
10.1	L'identification des clés primaires et des clés secondaires . . . . .	59
10.2	Les formes normales . . . . .	62
10.3	L'identification des contraintes d'intégrité référentielle . . . . .	62
10.4	La réalisation du dictionnaire des données . . . . .	62
10.5	L'algèbre relationnelle . . . . .	63

- 11. Correction des exercices . . . . . 65
  - 11.1 L'identification des clés primaires et des clés secondaires . . . . . 65
  - 11.2 Les formes normales . . . . . 66
  - 11.3 L'identification des contraintes d'intégrité référentielle . . . . . 67
  - 11.4 La réalisation du dictionnaire des données . . . . . 68
  - 11.5 L'algèbre relationnelle. . . . . 69

**Chapitre 2**  
**Implémentation des bases de données**

- 1. Présentation de SQL Server . . . . . 83
  - 1.1 L'installation de l'environnement . . . . . 85
    - 1.1.1 L'installateur . . . . . 87
    - 1.1.2 SQL Server . . . . . 91
    - 1.1.3 SQL Server Management Studio . . . . . 96
  - 1.2 L'utilisation de SQL Server Management Studio . . . . . 97
    - 1.2.1 La connexion . . . . . 97
    - 1.2.2 Les fenêtres . . . . . 98
- 2. Les principes élémentaires de la gestion d'une base de données . . . 102
  - 2.1 Les trois catégories d'instructions . . . . . 102
  - 2.2 Les instructions. . . . . 103
  - 2.3 Les commentaires . . . . . 103
  - 2.4 Les identificateurs. . . . . 104
  - 2.5 Les instructions du Data Definition Language . . . . . 105
    - 2.5.1 La création . . . . . 105
    - 2.5.2 Le contexte d'exécution . . . . . 106
    - 2.5.3 La modification . . . . . 107
    - 2.5.4 La suppression . . . . . 108
- 3. Les tables . . . . . 108
  - 3.1 La création. . . . . 109
  - 3.2 Les types de données . . . . . 114
    - 3.2.1 Les chaînes de caractères. . . . . 114
    - 3.2.2 Les valeurs numériques. . . . . 115

3.2.3	Les valeurs binaires . . . . .	116
3.2.4	Les dates. . . . .	117
3.2.5	Les autres types. . . . .	118
3.2.6	Les synonymes pour coller à la norme. . . . .	120
3.2.7	Les types de données définis par l'utilisateur . . . . .	121
3.2.8	Les attributs requis . . . . .	125
3.3	Les contraintes . . . . .	126
3.3.1	Le positionnement des contraintes . . . . .	126
3.3.2	Le nommage des contraintes . . . . .	127
3.3.3	Les clés primaires. . . . .	128
3.3.4	Les clés secondaires. . . . .	130
3.3.5	Les clés étrangères . . . . .	132
3.3.6	Les vérifications. . . . .	136
3.3.7	Les valeurs par défaut . . . . .	140
3.4	Les valeurs auto-incrémentées et les séquences . . . . .	142
3.4.1	Les valeurs auto-incrémentées . . . . .	142
3.4.2	Les séquences. . . . .	145
3.5	Les colonnes calculées. . . . .	148
3.6	La modification d'une table . . . . .	149
3.6.1	La suppression d'une contrainte. . . . .	150
3.6.2	L'ajout d'une contrainte . . . . .	151
3.6.3	La modification d'une contrainte. . . . .	151
3.6.4	L'activation et la désactivation d'une contrainte . . . . .	152
3.6.5	L'ajout d'un attribut . . . . .	152
3.6.6	La modification d'un attribut. . . . .	153
3.6.7	La suppression d'un attribut. . . . .	153
3.6.8	Le renommage. . . . .	154
3.7	La suppression d'une table. . . . .	154
4.	Les index . . . . .	155
4.1	La notion d'index . . . . .	155
4.2	L'intérêt d'une bonne indexation. . . . .	156

4.3	Les index organisés et les index non organisés	158
4.3.1	Les tables sans index organisé	158
4.3.2	Les index organisés	159
4.3.3	Les index non organisés	159
4.3.4	Les index couvrants	160
4.3.5	Les index filtrés	160
4.3.6	Les index et les colonnes calculées	160
4.3.7	Les index et les calculs d'agrégat	161
4.4	La création	161
4.5	La suppression d'un index	166
4.6	La reconstruction d'un index	167
4.7	Les statistiques	168
4.8	L'obtention d'informations	170
5.	Les schémas	172
5.1	L'intérêt	172
5.2	La création	173
5.3	Le nom complet d'une table	173
6.	La gestion d'une base de données	174
6.1	Le schéma de la base de données	174
6.2	La gestion de l'espace de stockage	176
6.3	La gestion de l'objet DATABASE	178
6.3.1	La création d'une base de données	179
6.3.2	La modification de la taille des fichiers	182
6.3.3	La suppression d'une base de données	186
6.3.4	La configuration d'une base de données	186
6.4	Les bases de données en mémoire	194
6.4.1	La création d'une base de données en mémoire	195
6.4.2	La création d'une table en mémoire	195
7.	La surveillance et la vérification des bases et des objets	197
8.	Exercice	197
8.1	La création de la base de données	197
8.2	La création des tables	198

8.3	L'ajout d'une colonne calculée . . . . .	200
8.4	L'ajout d'index pour accélérer les jointures . . . . .	201
8.5	L'ajout d'index pour accélérer les recherches et les tris . . . . .	201
8.6	La création d'une base de données plus finement paramétrée . . . . .	201
9.	Solutions des exercices . . . . .	202
9.1	La création de la base de données . . . . .	202
9.2	La création des tables . . . . .	202
9.3	L'ajout d'une colonne calculée . . . . .	204
9.4	L'ajout d'index pour accélérer les jointures . . . . .	204
9.5	L'ajout d'index pour accélérer les recherches et les tris . . . . .	205
9.6	La création d'une base de données plus finement paramétrée . . . . .	205

## Chapitre 3

### Les ordres du SQL

1.	Les bases du langage SQL . . . . .	207
1.1	Les expressions . . . . .	207
1.2	Les opérateurs . . . . .	210
1.2.1	Les opérateurs arithmétiques . . . . .	210
1.2.2	La concaténation de chaînes de caractères . . . . .	211
1.2.3	Les opérateurs binaires . . . . .	211
1.2.4	Les opérateurs de comparaison . . . . .	211
1.3	Les fonctions . . . . .	214
1.3.1	Les fonctions d'agrégation . . . . .	215
1.3.2	Les fonctions mathématiques . . . . .	216
1.3.3	Les fonctions trigonométriques . . . . .	218
1.3.4	Les fonctions logarithmiques . . . . .	218
1.3.5	La fonction de tirage de nombres aléatoires . . . . .	219
1.3.6	Les fonctions sur les dates . . . . .	219
1.3.7	Les fonctions sur les chaînes de caractères . . . . .	222
1.3.8	Les fonctions système . . . . .	225
1.3.9	Les fonctions de conversion de type . . . . .	229
1.3.10	Les fonctions logiques . . . . .	232

- 2. Les instructions du Data Manipulation Language . . . . . 232
  - 2.1 L'insertion de lignes . . . . . 232
  - 2.2 La modification de lignes . . . . . 238
  - 2.3 La suppression de lignes . . . . . 242
  - 2.4 L'extraction de lignes . . . . . 246
  - 2.5 Les opérations de l'algèbre relationnelle . . . . . 247
    - 2.5.1 La sélection de colonnes . . . . . 247
    - 2.5.2 La restriction . . . . . 249
    - 2.5.3 Les calculs élémentaires . . . . . 251
    - 2.5.4 La projection . . . . . 252
    - 2.5.5 Les calculs d'agrégat . . . . . 254
    - 2.5.6 Le produit cartésien . . . . . 256
    - 2.5.7 La jointure . . . . . 257
    - 2.5.8 La jointure externe . . . . . 259
    - 2.5.9 Le tri . . . . . 262
    - 2.5.10 L'union . . . . . 266
    - 2.5.11 La différence . . . . . 268
    - 2.5.12 L'intersection . . . . . 268
    - 2.5.13 L'extraction des premières lignes seulement . . . . . 269
  - 2.6 Les requêtes de création de tables et les tables temporaires . . . 272
  - 2.7 L'optimiseur de requête . . . . . 273
  - 2.8 Les tables CTE . . . . . 274
  - 2.9 La génération de lignes statistiques . . . . . 276
  - 2.10 Les sous-requêtes . . . . . 283
  - 2.11 Le pivot . . . . . 286
    - 2.11.1 Le principe de fonctionnement . . . . . 286
    - 2.11.2 Un exemple d'utilisation . . . . . 287
    - 2.11.3 Le pivot en sens inverse . . . . . 289
  - 2.12 La fusion de données . . . . . 290
- 3. Les vues . . . . . 292
  - 3.1 La création . . . . . 294
  - 3.2 La manipulation des données . . . . . 297
  - 3.3 La suppression . . . . . 298

4. Exercices . . . . .	298
4.1 La suppression des données . . . . .	298
4.2 L'insertion du jeu d'essai . . . . .	298
4.3 Le test des contraintes d'intégrité . . . . .	299
4.4 La traduction de l'algèbre relationnelle en requêtes SQL. . . . .	299
4.5 L'écriture de requêtes SQL. . . . .	299
4.6 La création d'une vue . . . . .	300
5. Correction des exercices . . . . .	301
5.1 La suppression des données . . . . .	301
5.2 L'insertion du jeu d'essai . . . . .	301
5.3 Le test des contraintes d'intégrité . . . . .	304
5.4 La traduction de l'algèbre relationnelle en requêtes SQL. . . . .	305
5.5 L'écriture de requêtes SQL. . . . .	307
5.6 La création d'une vue . . . . .	311

## Chapitre 4

### Transact-SQL : le langage procédural

1. Le SQL procédural . . . . .	313
1.1 Les variables . . . . .	314
1.1.1 Les variables utilisateur. . . . .	314
1.1.2 Les variables système . . . . .	315
1.1.3 L'affichage . . . . .	319
1.2 Les transactions . . . . .	319
1.2.1 Le principe . . . . .	319
1.2.2 La gestion des transactions. . . . .	320
1.2.3 La gestion des verrous. . . . .	323
1.2.4 Les transactions à validation différée. . . . .	332
1.3 Les lots d'instructions et les scripts. . . . .	333
1.4 Le contrôle de flux . . . . .	334
1.4.1 Le retour. . . . .	334
1.4.2 L'expression CASE. . . . .	335
1.4.3 Les blocs d'instructions. . . . .	337

1.4.4	La structure de contrôle alternative . . . . .	337
1.4.5	Les boucles . . . . .	338
1.4.6	Les valeurs des lignes affectées . . . . .	339
1.5	Les curseurs . . . . .	340
1.5.1	La déclaration . . . . .	341
1.5.2	L'ouverture . . . . .	343
1.5.3	La récupération d'une ligne . . . . .	344
1.5.4	La fermeture . . . . .	344
1.5.5	La désallocation . . . . .	345
1.5.6	Des exemples . . . . .	345
1.6	Les exceptions . . . . .	347
1.6.1	Les messages d'erreur . . . . .	347
1.6.2	Le déclenchement d'une erreur par le programmeur . . . . .	349
1.6.3	La création d'un message d'erreur . . . . .	350
1.6.4	La gestion des erreurs . . . . .	355
2.	Les procédures stockées . . . . .	359
3.	Les fonctions stockées . . . . .	363
3.1	La création . . . . .	364
3.2	La modification . . . . .	367
3.3	La suppression . . . . .	367
3.4	L'utilisation du résultat d'une fonction en tant que table . . . . .	368
4.	L'exécution des procédures et fonctions . . . . .	369
4.1	L'appel . . . . .	369
4.2	Le contexte d'exécution . . . . .	371
5.	Les déclencheurs . . . . .	379
6.	Exercices . . . . .	388
6.1	La création d'une fiche de location et ses lignes associées . . . . .	388
6.2	Le montant d'une fiche . . . . .	388
6.3	L'enregistrement du paiement d'une fiche . . . . .	389
6.4	La synthèse des fiches . . . . .	389
6.5	La mise à jour automatique de l'état des fiches . . . . .	390

6.6	La vérification de la disponibilité des articles pour la location . . . . .	390
7.	Correction des exercices . . . . .	391
7.1	La création d'une fiche de location et ses lignes associées . . . . .	391
7.2	Le montant d'une fiche . . . . .	391
7.3	L'enregistrement du paiement d'une fiche . . . . .	392
7.4	La synthèse des fiches. . . . .	392
7.5	La mise à jour automatique de l'état des fiches . . . . .	393
7.6	La vérification de la disponibilité des articles pour la location . . . . .	393

## Chapitre 5

### Gestion des données distribuées

1.	Introduction . . . . .	395
2.	SQL Server Integration Services . . . . .	397
2.1	Les principes de fonctionnement. . . . .	397
2.2	L'assistant d'importation et d'exportation . . . . .	399
2.3	Les autres utilitaires . . . . .	413
2.3.1	L'utilitaire d'exécution de package. . . . .	413
2.3.2	Les utilitaires en ligne de commande. . . . .	415
3.	Le transport d'une base de données. . . . .	415
3.1	Le détachement d'une base . . . . .	416
3.1.1	Depuis SQL Server Management Studio. . . . .	416
3.1.2	Avec un script Transact-SQL . . . . .	418
3.2	Le rattachement d'une base. . . . .	419
3.2.1	Depuis SQL Server Management Studio. . . . .	420
3.2.2	Avec un script Transact-SQL . . . . .	421
3.3	Le déplacement d'une base ou des fichiers . . . . .	422
4.	Les exports et imports au format CSV. . . . .	422
4.1	Les instructions BULK . . . . .	422
4.1.1	L'import . . . . .	423
4.1.2	L'utilisation comme une table . . . . .	423

- 4.2 Le programme BCP . . . . . 426
- 5. Les serveurs liés . . . . . 431
  - 5.1 La gestion des serveurs liés . . . . . 436
    - 5.1.1 L'ajout d'un serveur lié . . . . . 437
    - 5.1.2 La suppression d'un serveur lié . . . . . 439
  - 5.2 La gestion des utilisateurs distants . . . . . 440
  - 5.3 L'utilisation des ressources distantes . . . . . 441
  - 5.4 Les synonymes . . . . . 442

**Chapitre 6**  
**Les types évolués**

- 1. Introduction . . . . . 445
- 2. Le format XML . . . . . 445
  - 2.1 Le type XML . . . . . 448
  - 2.2 La manipulation d'une colonne de type XML . . . . . 454
    - 2.2.1 Les opérations du DML . . . . . 454
    - 2.2.2 Les méthodes spécifiques . . . . . 455
  - 2.3 L'indexation d'une colonne de type XML . . . . . 459
    - 2.3.1 L'index principal . . . . . 460
    - 2.3.2 L'index secondaire . . . . . 461
  - 2.4 Les langages XQuery et XPath . . . . . 462
  - 2.5 L'instruction FOR XML . . . . . 469
  - 2.6 La méthode OPENXML . . . . . 472
  - 2.7 La méthode OPENROWSET . . . . . 474
- 3. Le type table value parameter . . . . . 474
- 4. Les structures hiérarchiques . . . . . 476
  - 4.1 Le type de données HierarchyId . . . . . 476
  - 4.2 Les index . . . . . 477
  - 4.3 Les méthodes . . . . . 478
- 5. Les données images . . . . . 481
- 6. Les données spatiales . . . . . 486

7. Exercices . . . . .	490
7.1 La mise en place d'une hiérarchie des catégories d'articles . . .	490
7.2 L'affichage de la hiérarchie des catégories. . . . .	491
7.3 L'export des articles au format XML. . . . .	492
8. Correction des exercices . . . . .	492
8.1 La mise en place d'une hiérarchie des catégories d'articles . . .	492
8.2 L'affichage de la hiérarchie des catégories. . . . .	493
8.3 L'export des articles au format XML. . . . .	494

## Chapitre 7

### Le Common Language Runtime (CLR)

1. SQL Server et le CLR . . . . .	495
1.1 Le Transact-SQL ou le CLR ? . . . . .	496
1.2 Le code CLR . . . . .	498
1.2.1 L'accès à des ressources externes . . . . .	499
1.2.2 Les instructions SQL. . . . .	499
1.2.3 Les types de données. . . . .	499
1.2.4 L'accès aux données. . . . .	500
1.2.5 La localisation du code . . . . .	500
2. La création du code CLR géré dans SQL Server . . . . .	501
2.1 L'activation de la prise en charge du code CLR . . . . .	501
2.2 Le travail dans Visual Studio . . . . .	501
2.3 Les types de données personnalisés . . . . .	503
2.4 La publication du code . . . . .	507
2.5 L'utilisation d'un type personnalisé . . . . .	510
2.6 Les procédures stockées . . . . .	512
2.7 Les déclencheurs de base de données. . . . .	513
2.8 Les fonctions de calcul d'agrégat . . . . .	516
3. La manipulation d'une base de données avec PowerShell . . . . .	519
3.1 L'installation de SMO . . . . .	519
3.2 L'utilisation de SMO . . . . .	520

- 4. Exercices ..... 522
  - 4.1 Un mot de passe pour les clients ..... 522
  - 4.2 Des graphiques de valeurs ..... 523
- 5. Correction des exercices ..... 524
  - 5.1 Un mot de passe pour les clients ..... 524
  - 5.2 Des graphiques de valeurs ..... 526

**Annexes**

- 1. Schéma de la base des exercices ..... 529
- 2. Le script de création de la base ..... 530
- 3. Le jeu d'essai ..... 533
- 4. Glossaire ..... 535

- Index ..... 537



## Chapitre 4

# Transact-SQL : le langage procédural

### 1. Le SQL procédural

SQL Server est un serveur de base de données relationnelle et à ce titre, il fournit tous les éléments pour stocker de façon structurée les données mais aussi les outils nécessaires pour travailler avec les données au travers de SQL. Avec le Transact-SQL, il est également possible de définir des traitements procéduraux directement dans la base de données. Ces traitements vont pouvoir être utilisables par tous les utilisateurs de la base sous réserve qu'ils possèdent les privilèges nécessaires. Il est possible de conserver la définition de ces traitements et de les rendre paramétrables par l'intermédiaire de la création de procédures et de fonctions.

Des traitements procéduraux pourront également être mis en place pour définir des contraintes d'intégrité complexes, il s'agira alors de déclencheurs de base de données (TRIGGER).

Le Transact-SQL est un langage procédural qui intègre complètement et nativement le langage SQL. Il est ainsi possible de tirer parti des deux langages. Par exemple, il va être très facile de définir une variable en Transact-SQL puis d'inclure cette variable dans une requête SQL. Lors de l'exécution de la requête, c'est la valeur contenue par la variable qui est prise en compte.

Le Transact-SQL n'a pas pour objectif de se substituer aux requêtes SQL mais plutôt de permettre un complément par rapport aux tâches réalisables en SQL.

À l'aide du Transact-SQL, il va être possible de définir des procédures et des fonctions dans la base de données. Les déclencheurs de base de données vont permettre la mise en place au niveau de la base de règles métiers complexes.

## 1.1 Les variables

### 1.1.1 Les variables utilisateur

Une variable est une zone mémoire, caractérisée par un nom et un type, permettant de stocker une valeur. Les variables Transact-SQL doivent obligatoirement être déclarées avant leur utilisation. Elles peuvent ensuite remplacer n'importe quelle expression dans les instructions SQL.

#### Déclaration de variables

```
DECLARE @nom_variable type [= expr][, ...][;]
```

nom\_variable

Nom précédé du caractère @.

type

Type système ou défini par l'utilisateur.

#### Valorisation des variables

```
SELECT @nom_variable = expr [, ...][FROM...]
```

### Exemple

Dans les scripts suivants, les variables sont tout d'abord définies. La variable @choix est ensuite valorisée avec la valeur 20. Puis une requête de type SELECT permet de renseigner le contenu des variables @nom et @prenom. Cette requête fonctionne correctement car la restriction sur une valeur unique de la clé primaire permet de garantir qu'une seule ligne de données sera remontée depuis la base. Enfin, la dernière instruction SELECT permet de visualiser le contenu des variables @nom et @prenom.

```
DECLARE @choix int;  
DECLARE @nom CHAR(30), @prenom CHAR(30);  
SELECT @choix=20;  
SELECT @nom=nom, @prenom=prenom  
FROM clients  
WHERE idClient = @choix;  
SELECT @nom, @prenom;
```

Résultats		Messages	
	(Aucun nom de colonne)		(Aucun nom de colonne)
1	LABICHE		Adeline

### 1.1.2 Les variables système

Ces variables sont définies par le système, et peuvent être utilisées seulement en lecture. Elles se distinguent des variables utilisateur par le double @.

#### @@CONNECTIONS

Nombre de connexions ou de tentatives de connexion depuis le dernier démarrage de SQL Server.

#### @@CPU\_BUSY

Temps consacré par l'unité centrale à SQL Server depuis le dernier démarrage de celui-ci. Le résultat est exprimé en unité CPU. Il faut multiplier par @@TIMETICKS pour obtenir un résultat en microsecondes.

#### @@CURSOR\_ROWS

Nombre de lignes affectées dans le dernier curseur ouvert. Les curseurs sont présentés dans la section Les curseurs de ce chapitre.

**@@DATEFIRST**

Renvoie la valeur courante du paramètre SET DATEFIRST.

**@@DBTS**

Valeur du type de données timestamp courant pour la base de données.

**@@ERROR**

Dernier numéro d'erreur généré par le système.

**@@FETCH\_STATUS**

Contient le statut d'une commande de curseur FETCH.

**@@IDENTITY**

Enregistre la dernière valeur IDENTITY insérée.

**@@IDLE**

Temps, en millisecondes, pendant lequel SQL Server est resté inactif depuis son dernier démarrage.

**@@IO\_BUSY**

Temps, en millisecondes, consacré par SQL Server à effectuer des opérations d'entrée/sortie depuis son dernier démarrage.

**@@LANGID**

Identificateur de la langue actuellement utilisée.

**@@LANGUAGE**

Langue actuellement utilisée.

**@@LOCK\_TIMEOUT**

Timeout, en millisecondes, de la session en cours.

**@@MAX\_CONNECTIONS**

Nombre maximal de connexions simultanées qu'il est possible d'établir avec SQL Server.

### @@MAX\_PRECISION

Renvoie le niveau de précision utilisé par les types de données **decimal** et **numeric**.

### @@NESTLEVEL

Niveau d'imbrication de l'instruction en cours d'exécution.

### @@OPTIONS

Informations sur les valeurs courantes des options SET.

### @@PACK\_RECEIVED

Nombre de paquets entrants lus par SQL Server depuis son dernier démarrage.

### @@PACK\_SENT

Nombre de paquets sortants écrits par SQL Server depuis son dernier démarrage.

### @@PACKET\_ERRORS

Nombre d'erreurs qui se sont produites alors que SQL Server envoyait ou recevait des paquets depuis son dernier démarrage.

### @@PROCID

Identificateur de la procédure stockée Transact-SQL du traitement en cours d'exécution.

### @@REMSERVER

Renvoie le nom du serveur contenu dans l'enregistrement des noms d'accès d'un serveur distant.

### @@ROWCOUNT

Nombre de lignes affectées par la dernière instruction.

### @@SERVERNAME

Nom du serveur SQL local.

**@@SERVICENAME**

Nom du service en cours d'exécution.

**@@SPID**

Numéro d'identification du processus courant sur le serveur.

**@@TEXTSIZE**

Longueur maximale, en octets, des données texte ou image renvoyées par une instruction SELECT.

**@@TIMETICKS**

Nombre de millisecondes par pulsation.

**@@TOTAL\_ERRORS**

Nombre d'erreurs rencontrées par SQL Server en lisant ou en écrivant des données depuis son dernier démarrage.

**@@TOTAL\_READ**

Nombre de lectures de données sur le disque effectuées par SQL Server depuis son dernier démarrage.

**@@TOTAL\_WRITE**

Nombre d'écritures de données sur le disque effectuées par SQL Server depuis son dernier démarrage.

**@@TRANCOUNT**

Nombre de transactions actuellement actives pour l'utilisateur courant.

**@@VERSION**

Date, numéro de version et type de processeur de la version courante de SQL Server.

Ces variables système contiennent de nombreuses informations et en les interrogeant il est déjà possible d'obtenir beaucoup d'informations sur le serveur.

### Exemple

Dans l'exemple ci-dessous, la variable @@VERSION est interrogée et la version exacte de SQL Server est retournée. Avec ce numéro de version, il est par exemple possible de savoir quels Service Packs sont appliqués sur le serveur en s'appuyant sur les tableaux de correspondance disponibles sur le site web Technet de Microsoft.

```
SELECT @@VERSION;
Microsoft SQL Server 2019 (RTM-GDR) (KB4517790) - 15.0.2070.41 (X64)
Oct 28 2019 19:56:59 Copyright (C) 2019 Microsoft Corporation
Developer Edition (64-bit) on Windows 10 Pro 10.0 <X64> (Build 18363: )
```

### 1.1.3 L'affichage

L'instruction PRINT affiche un message.

#### Syntaxe

```
PRINT {'texte' | @variable | @@variablesystème}
```

#### Exemple

```
DECLARE @nb INT;
SELECT @nb=COUNT(*) FROM Clients;
PRINT 'Nombre de clients : '
PRINT @nb;
```

#### **Affichage**

```
Nombre de clients :
5
```

## 1.2 Les transactions

### 1.2.1 Le principe

Il est possible de définir une transaction comme un ensemble indivisible d'instructions, c'est-à-dire dans lequel toutes les instructions réussissent ou bien aucune. Cette notion de transaction est bien présente dans le monde réel, par exemple lors d'un retrait d'argent à un distributeur automatique bancaire les deux éléments indivisibles sont le débit du compte et la distribution des billets. Si l'un de ces éléments ne peut être exécuté correctement (et quelle qu'en soit la raison) alors c'est l'ensemble qui est mis en échec.

En effet, il n'est pas envisageable de distribuer des billets sans que le compte ne soit débité du montant correspondant.

## 1.2.2 La gestion des transactions

Le premier point à prendre en considération est le verrouillage des informations. En effet, lorsque SQL Server lit des données ou les modifie, il verrouille les lignes d'informations manipulées. Ce verrouillage dure le temps de l'exécution de l'instruction ou le temps de la transaction.

Suivant le type de verrous posés, il est possible ou non à d'autres transactions d'accéder simultanément à la même information.

Une transaction est un ensemble d'instructions de manipulations de données s'exécutant dans une même unité de travail. La validation d'une transaction assure que toutes les instructions en faisant partie se sont correctement terminées, l'annulation de la transaction assurera l'annulation de l'ensemble des instructions.

Seules les instructions du DML (SELECT, INSERT, UPDATE, DELETE) sont prises en compte dans une transaction.

Les transactions sont utiles pour assurer l'intégrité et la cohérence des données lors de modifications multiples, pour améliorer les performances, pour tester les effets de modification, pour gérer les verrouillages.

Une transaction est caractérisée par le mot-clé ACID (*Atomicity Consistency Isolation Durability*) qui peut se traduire en français par Atomique Consistance Indépendance Durée.

- Atomique car une transaction représente une unité atomique (non divisible) de travail pour le serveur.
- Consistance car à la fin de la transaction les informations présentes dans la base doivent être consistantes, c'est-à-dire cohérentes par rapport aux règles de structuration des données mises en place.