

ADMINISTRATION ET SÉCURITÉ DES BASES DE DONNÉES

Installer et configurer efficacement un SGBDR**Construire des requêtes avancées**

Notes de cours – Semaine-01 (A1, A2)

Par

Professeur: **Robert Yavo**Email: ryavo@hotmail.com WhatsApp: +225-07-88-63-26-58

Table des matières

A1 – Installer et configurer efficacement un SGBD	5
1. Planification de l’installation de Microsoft SQL Server 2019	5
1.1 Les spécifications de la capacité maximale de Microsoft SQL Server	5
1.1.1 Database Engine Objects (Les objets du Moteur de Base de données)	5
1.1.2 SQL Server Utility Objects (Les objets de l’utilitaire SQL Server).....	7
1.1.3 Replication Objects (Les objets de Replication).....	7
1.1.4 Compute capacity limits by edition (Limite de calcul de capacité par Éditions) 8	
1.2 Les exigences Matériels et Logiciels	8
1.3 Considérations sécuritaires	9
1.3.1 Avant d'installer SQL Server	9
1.3.2 Après avoir installé SQL Server	9
1.4 Protocol Réseau	9
1.5 Plusieurs versions et instances de SQL Server.....	10
Installation distincte de SSRS	10
Prévenir les conflits d’Adresse IP	10
1.6 Configuration des différentes langues de SQL Server	10
Vérification des paramètres de langues de Windows	10
Paramètres à vérifier dans Windows	10
Modification du paramètre d'interface utilisateur du système d'exploitation	11
Modification des paramètres régionaux de l'utilisateur du système d'exploitation 11	
Pour modifier les paramètres régionaux du système.....	11
1.7 Emplacements de fichiers et mappage du registre	11
Emplacements des fichiers d’installation	11
Mappage du Registre	12
2. Construction d’un Laboratoire Virtuel SQL Server sur Windows	13
2.1 Création et configuration des machines virtuelles (TD-01).....	13
2.1.1 Télécharger et installer Oracle VirtualBox 6.1.36	13
2.1.2 Créer les machines virtuelles (VM)	15
2.1.3 Configurer les machines virtuelles	17
2.1.4 Configurer le Réseau Virtuel	18
2.2 Installation de Windows 2019 Server dans les Machines Virtuelles.....	20

2.2.1 Installation et Configuration de Windows Server 2019 du Domain Controller DV-DC	21
2.2.3 Installation et Configuration de Windows-Server-2019-VM1	30
2.2.4 Télécharger le SQL Server Management Studio (SSMS) sur votre PC	33
3. Installation et configuration de base (MS SQL Server 2019)	33
3.1 Installation de SQL Server 2019 et de SSMS.....	33
4. Configuration avancée du réseau et des 2 Serveurs SQL Server 2019	38
4.1 Création des comptes de Domain pour les services SQL Server	38
4.2 Configuration du Serveur DV-SQL01 de la Windows-Server-2019-VM1.....	41
4.3 Connection au Moteur de la Base de données SQL Server (DV-SQL01) avec SSMS	44
4.4 Configuration Avancée du Réseau Virtuel	46
4.4.2 Déployer un utilitaire de SQL Server Maintenance Solution	46
4.4.3 Activer TCP/IP pour SQL Server.....	50
4.4.4 Configurer le Firewall de Windows dans la machine virtuelle	51
4.5 Configuration d'une nouvelle machine virtuelle et Serveur SQL DV-SQL02	53
5. Tester les connexions et les communications des Serveurs.....	54
A2 – Construire et exécuter des requêtes avancées	55
1. Rappels sur la conception d'un SGBDR	55
1.1 Définition	55
1.2 Types de SGBD	55
1.3 Le Diagramme d'Entité-Relation (ER-Diagram)	55
1.4 Processus de DESIGN d'un SGBDR.....	57
1.5 Normalisation.....	58
1.6 Meilleures pratiques dans le Design d'une Base de données relationnelle	59
1.6.1. Tenez compte de chaque point de vue lors de la planification	59
1.6.2. Choisissez un type de base de données.....	59
1.6.3. Normalisez vos données	60
1.6.4. Rendre les structures transparentes.....	60
1.6.5. Définissez les contraintes pour maintenir l'intégrité des données	60
1.6.6. Documentez tout	60
1.6.7. Prévoyez d'augmenter le temps de sauvegarde.....	60
1.6.8. Gardez la confidentialité primaire	61
1.6.9. Optimisez pour la vitesse	61

1.6.10. Gardez la base de données sur son propre serveur	61
2. Requêtes SQL avancées (LDD, LMD, LCD, LCT)	62
2.1 Définition SQL	62
2.2 Langage de Définition de Données (LDD)	62
2.3 Langage de Manipulation de Données (LMD)	66
2.4 Langage de Contrôle de Transaction (LCT)	70
2.5 Langage de Contrôle de Données (LCD)	70
2.6 Connexions à distance	73
Références.....	74
Annexes	74
Annexe 1 – Best practices of Relational Database Design (20 steps)	74
Annexe 2 – Liste des Fonctions dans SQL Server 2019	76

A1 – Installer et configurer efficacement un SGBDR

1. Planification de l'installation de Microsoft SQL Server 2019

L'installation efficace d'un SGBDR comme Microsoft SQL Server 2019 de façon professionnelle, requiert une procédure de planification en 3 étapes que l'Administrateur des Bases de Données (DBA en anglais) devrait considérer.

- Passer en revue les exigences d'installation, les vérifications de la configuration système et les considérations de sécurité pour une installation de SQL Server.
- Exécuter le programme d'installation de SQL Server du fabricant Microsoft pour installer ou mettre à jour une version du produit. Il est prudent de suivre les procédures et documentations d'installation ou de mise à niveau du fabricant.
- Utiliser les utilitaires SQL Server pour configurer SQL Server.

1.1 Les spécifications de la capacité maximale de Microsoft SQL Server

1.1.1 Database Engine Objects (Les objets du Moteur de Base de données)

Le tableau suivant présente la valeur maximale des objets du moteur SQL Server 64-bit

QL Server Database Engine object	Max values for SQL Server (64-bit)
Batch size	65,536 * (network packet size)
Byte length of a string containing Transact-SQL statements (batch size)	65,536 * (network packet size)
Bytes per short string column	8,000
Bytes per GROUP BY, ORDER BY	8,060
Bytes per index key	900 bytes for a clustered index. 1,700 bytes for a non clustered index.
Bytes per index key for memory-optimized tables	2,500 bytes for a non clustered index. No limit for a hash index, as long as all index keys fit in-row.
Bytes per foreign key	900
Bytes per primary key	900
Bytes per row	8,060
Bytes per row in memory-optimized tables	8,060
Bytes in source text of a stored procedure	Lesser of batch size or 250 MB
Bytes per varchar(max), varbinary(max), xml, text, or image column	2 ³¹ -1
Characters per ntext or nvarchar(max) column	2 ³⁰ -1
Clustered indexes per table	1
Columns in GROUP BY, ORDER BY	Limited only by number of bytes
Columns or expressions in a GROUP BY WITH CUBE or GROUP BY WITH ROLLUP statement	10

Columns per index key	32
Columns per foreign key or primary key	32
Columns per INSERT statement	4,096
Columns per SELECT statement	4,096
Columns per table	1,024
Columns per UPDATE statement	4,096
Columns per view	1,024
Connections per client	Maximum value of configured connections
Database size	524,272 terabytes
Databases per instance of SQL Server	32,767
Filegroups per database	32,767
Filegroups per database for memory-optimized data	1
Files per database	32,767
File size (data)	16 terabytes
File size (log)	2 terabytes
Data files for memory-optimized data per database	4,096 in SQL Server 2014 (12.x). The limit is less strict on SQL Server 2016 (13.x) and later.
Delta file per data file for memory-optimized data	1
Foreign key table references per table	Outgoing = 253.
Incoming = 10,000.	For restrictions, see Create Foreign Key Relationships.
Identifier length (in characters)	128
Instances per computer	50 instances on a stand-alone server. 25 failover cluster instances when using a shared cluster drive as storage. 50 failover cluster instances with SMB file shares as the storage option.
Indexes per memory-optimized table	999 starting SQL Server 2017 (14.x) and in Azure SQL Database.
8 in SQL Server 2016 (13.x) and SQL Server 2014 (12.x).	
Locks per instance of SQL Server	Limited only by memory
Nested stored procedure levels	32
Nested subqueries	32
Nested transactions	4,294,967,296
Nested trigger levels	32
Non clustered indexes per table	999
Number of distinct expressions in the GROUP BY clause when any of the following are present: CUBE, ROLLUP, GROUPING SETS, WITH CUBE, WITH ROLLUP	32
Number of grouping sets generated by operators in the GROUP BY clause	4,096
Parameters per stored procedure	2,100

Parameters per user-defined function	2,100
REFERENCES per table	253
Rows per table	Limited by available storage
Tables per database	Limited by total number of objects in a database
Partitions per partitioned table or index	15,000
Statistics on non-indexed columns	30,000
Tables per SELECT statement	Limited only by available resources
Triggers per table	Limited by number of objects in a database
User connections	32,767
XML indexes	249

1.1.2 SQL Server Utility Objects (Les objets de l'utilitaire SQL Server)

SQL Server Utility object	Maximum values for SQL Server (64-bit)
Computers (physical computers or virtual machines) per SQL Server Utility	100
Instances of SQL Server per computer	5
Total number of instances of SQL Server per SQL Server Utility	200
User databases per SQL Server instance, including data-tier applications	50
Total number of user databases per SQL Server Utility	1,000
File groups per database	1
Data files per file group	1
Log files per database	1
Volumes per computer	3

1.1.3 Replication Objects (Les objets de Replication)

SQL Server Replication object	Max values for SQL Server 64-bit
Articles (merge publication)	2,048
Articles (snapshot or transactional publication)	32,767
Columns in a table (merge publication)	246
Columns in a table (SQL Server snapshot or transactional publication)	1,000
Columns in a table (Oracle snapshot or transactional publication)	995
Bytes for a column used in a row filter (merge publication)	1,024
Bytes for a column used in a row filter (snapshot or transactional publication)	8,000

1.1.4 Compute capacity limits by edition (Limite de calcul de capacité par Éditions)

SQL Server edition	Maximum compute capacity for a single instance (SQL Server Database Engine)	Maximum compute capacity for a single instance (AS, RS)
Enterprise Edition: Core-based Licensing*	Operating system maximum	Operating system maximum
Developer	Operating system maximum	Operating system maximum
Standard	Limited to lesser of 4 sockets or 24 cores	Limited to lesser of 4 sockets or 24 cores
Express	Limited to lesser of 1 socket or 4 cores	Limited to lesser of 1 socket or 4 cores

1.2 Les exigences Matériels et Logiciels

Matériels (Components)	Exigences (Requirements)
Disque Dur	Minimum 6 GB d'espace vide
Écran	Super-VGA 800x600
Internet	Accès Internet
Mémoire RAM	Min: 0.5GB pour Express et 1 GB pour toutes les autres Recommandé : 1GB pour et 4GB pour les autres
Type de CPU (x64)	AMD Opteron/Athlon 64, Intel Xeon avec EM64T support
Vitesse CPU	Min 1.4 Ghz, Recommandé : 2Ghz et plus.
Software (Logiciels)	
Système d'Exploitation	Windows 10 TH1 1507 ou plus récent (Dev.) Windows Server 2016 ou plus récent
.NET Framework	Intégré dans le système d'Exploitation
Storage types (type de stockage)	
Local Disk	Supporte les Disques durs de taille de secteur 512B et 4KB. Peut avoir des erreurs si plus de 4KB
Shared storage	Stockage partagé
Storage Spaces Direct (S2D) Ex : Dell PowerEdge R730XD	Serveurs S2D qui contiennent des stockages de haute disponibilité et d'évolution, comme SAN ou NAS
Domain Controller	
PDC ou BDC	Il n'est pas conseillé d'installer SQL Server dans Contrôleur de Domaine (DC) pour raison de sécurité.

1.3 Considérations sécuritaires

La sécurité est importante pour chaque produit et chaque entreprise. En suivant les meilleures pratiques simples, on peut éviter beaucoup de problèmes de sécurité. Il y a donc des meilleures pratiques de sécurité à prendre en compte avant d'installer SQL Server et même après l'installation de SQL Server.

1.3.1 Avant d'installer SQL Server

Suivez ces meilleures pratiques lorsque vous configurez l'environnement du serveur :

Améliorer la sécurité physique

Utiliser des pare-feux

Isoler les services (Exécuter les Services du SQL Server avec des accès Usagers différents)

Configurer un système de fichiers sécurisé (NTFS ou ReFS)

Désactiver NetBIOS et le bloc de messages du serveur (UDP 137/138 et TCP139/445)

Éviter d'installer SQL Server sur un contrôleur de domaine.

1.3.2 Après avoir installé SQL Server

Nous allons voir plus tard, les meilleurs pratiques de configuration sécuritaire dans la section Sécurité des Bases de données

1.4 Protocol Réseau

Un serveur peut écouter ou surveiller plusieurs protocoles réseau à la fois. Cependant, chaque protocole doit être configuré. Si un protocole particulier n'est pas configuré, le serveur ne peut pas écouter sur ce protocole. Après l'installation, vous pouvez modifier les configurations de protocole à l'aide du gestionnaire de configuration SQL Server

(SQL Server Configuration Manager).

Une instance par défaut de SQL Server est configurée pour le **port TCP/IP 1433** et nommée **pipe \\.\pipe\sql\query**. Les instances nommées SQL Server sont configurées pour les ports dynamiques TCP, avec un numéro de port attribué par le système d'exploitation.

Si vous ne pouvez pas utiliser d'adresses de port dynamiques (par exemple, lorsque les connexions SQL Server doivent passer par **un serveur pare-feu** configuré pour passer par des adresses de port spécifiques). Sélectionnez un numéro de port non attribué.

Pour améliorer la sécurité, la connectivité réseau n'est pas entièrement activée lorsque SQL Server est installé. Pour activer, désactiver et configurer les protocoles réseau une fois l'installation terminée, utilisez la zone Configuration réseau SQL Server du **Gestionnaire de configuration SQL Server**.

1.5 Plusieurs versions et instances de SQL Server

Installation distincte de SSRS

Dans les versions antérieures à SQL Server 2016, les 3 éléments suivants étaient liés à SQL Server et compatibles avec l'installation de plusieurs instances sur le même ordinateur :

Moteur de base de données (Database Engine)

Services d'analyse (SSAS)

Reporting Services (SSRS)

À partir de SQL Server 2016, SQL Server Reporting Services (SSRS) a une installation distincte.

Prévenir les conflits d'Adresse IP

Des conflits se produisent généralement lorsque deux instances du moteur de base de données sont toutes deux configurées pour utiliser le port TCP par défaut (1433). Pour éviter les conflits, configurez une instance pour utiliser un port fixe autre que celui par défaut. La configuration d'un port fixe est généralement plus facile sur l'instance autonome. La configuration du moteur de base de données pour utiliser différents ports empêchera un conflit inattendu d'adresse IP/de port TCP qui bloque le démarrage d'une instance lorsqu'une instance de cluster de basculement SQL Server échoue sur le nœud de secours.

1.6 Configuration des différentes langues de SQL Server

Vérification des paramètres de langues de Windows

Configurer le système d'exploitation pour prendre en charge les versions localisées.

Les versions localisées de SQL Server sont prises en charge sur les versions anglaises des systèmes d'exploitation pris en charge via l'utilisation des paramètres du pack d'interface utilisateur multilingue (MUI) de Windows.

Toutefois, vous devez vérifier certains paramètres du système d'exploitation avant d'installer une version localisée de SQL Server sur un serveur qui exécute un système d'exploitation en anglais avec un paramètre MUI non anglais.

Paramètres à vérifier dans Windows

- Le paramètre d'interface utilisateur du système d'exploitation
- Le paramètre régional de l'utilisateur du système d'exploitation
- Le paramètre régional du système

Si les paramètres ne correspondent pas à la langue du serveur SQL localisé à installer, utilisez les procédures suivantes pour définir correctement ces paramètres du système d'exploitation :

Modification du paramètre d'interface utilisateur du système d'exploitation

- S'il n'est pas déjà installé, installez la MUI du système d'exploitation qui correspond à votre version localisée de SQL Server.
- Dans le Panneau de configuration, ouvrez **Options régionales et linguistiques**.
- Dans l'onglet **Langues**, pour **Langue utilisée dans les menus et les boîtes de dialogue**, sélectionnez une valeur dans la liste.
- Ce paramètre affectera la langue de l'interface utilisateur de SQL Server, il doit donc correspondre à votre version localisée de SQL Server.
- Cliquez sur **Appliquer** pour confirmer la modification et sur **OK** pour fermer la fenêtre.

Modification des paramètres régionaux de l'utilisateur du système d'exploitation

- S'il n'est pas déjà installé, installez la MUI du système d'exploitation qui correspond à votre version localisée de SQL Server.
- Dans le Panneau de configuration, ouvrez **Options régionales et linguistiques**.
- Dans l'onglet **Options régionales**, pour Sélectionner un **élément pour qu'il corresponde à ses préférences**, sélectionnez une valeur dans la liste.
Ce paramètre affectera la mise en forme des données spécifiques à la culture.
- Cliquez sur **Appliquer** pour confirmer la modification et sur **OK** pour fermer la fenêtre.

Pour modifier les paramètres régionaux du système

- S'il n'est pas déjà installé, installez la MUI du système d'exploitation qui correspond à votre version localisée de SQL Server.
- Dans le Panneau de configuration, ouvrez **Options régionales et linguistiques**.
- Dans l'onglet **Avancé**, pour **Sélectionner une langue correspondant à la version linguistique des programmes non Unicode que vous souhaitez utiliser**, sélectionnez une valeur dans la liste.
Ce paramètre permettra au programme d'installation de SQL Server de choisir le meilleur classement par défaut pour votre installation SQL Server.
- Cliquez sur **Appliquer** pour confirmer la modification et sur **OK** pour fermer la fenêtre

1.7 Emplacements de fichiers et mappage du registre

Emplacements des fichiers d'installation

Lors de l'installation de SQL Server, un ID d'instance est généré pour chaque composant serveur. Les composants serveur de cette version de SQL Server sont le moteur de base de données. (Analysis Services et Reporting Services).

L'ID d'instance par défaut est construit en utilisant le format suivant :

- MSSQL pour le moteur de base de données, suivi du numéro de version majeure, suivi d'un trait de soulignement et de la version mineure le cas échéant, et d'un point, suivi du nom de l'instance.
- MSAS for Analysis Services, suivi du numéro de version majeure, suivi d'un trait de soulignement et de la version mineure le cas échéant, et d'un point, suivi du nom de l'instance.
- MSRS pour Reporting Services, suivi du numéro de version majeure, suivi d'un trait de soulignement et de la version mineure le cas échéant, et d'un point, suivi du nom de l'instance.

Voici des exemples d'ID d'instance par défaut dans cette version de SQL Server :

- MSSQL{nn}.MSSQLSERVER pour une instance par défaut de SQL Server.
- MSAS{nn}.MSSQLSERVER pour une instance par défaut de SQL Server Analysis Services.
- MSSQL{nn}.MyInstance pour une instance nommée de SQL Server nommée "MyInstance".

La structure de répertoires d'une instance nommée SQL Server qui inclut le moteur de base de données et Analysis Services, nommée « MyInstance », et installée dans les répertoires par défaut serait la suivante :

- C:\Program Files\Microsoft SQL Server\MSSQL{nn}.MyInstance\
- C:\Program Files\Microsoft SQL Server\MSAS{nn}.MyInstance\

Vous pouvez spécifier n'importe quelle valeur pour l'ID d'instance, mais évitez les caractères spéciaux et les mots clés réservés.

Mappage du Registre

Le registre est créé sous HKLM\Software\Microsoft\Microsoft SQL Server\<Instance_ID> pour les composants prenant en charge les instances. Par exemple,

HKLM\Software\Microsoft\Microsoft SQL Server\MSSQL{nn}.MyInstance

HKLM\Software\Microsoft\Microsoft SQL Server\MSAS{nn}.MyInstance

HKLM\Software\Microsoft\Microsoft SQL Server\MSRS{nn}.MyInstance

Le registre maintient également un mappage de l'ID d'instance au nom de l'instance. Le mappage entre l'ID d'instance et le nom d'instance est conservé comme suit :

- [HKEY_LOCAL_MACHINE\Software\Microsoft\Microsoft SQL Server\Noms d'instance\SQL] "<NomInstance>"="MSSQL{nn}"
- [HKEY_LOCAL_MACHINE\Software\Microsoft\Microsoft SQL Server\Noms d'instance\OLAP] "<NomInstance>"="MSAS{nn}"
- [HKEY_LOCAL_MACHINE\Software\Microsoft\Microsoft SQL Server\Noms d'instance\RS] "<NomInstance>"="MSRS{nn}"

2. Construction d'un Laboratoire Virtuel SQL Server sur Windows

Nous allons dans un premier temps créer des machines virtuelles (VM) à l'aide de l'outil Oracle VirtualBox et installer Windows Server 2019, puis SQL Server 2019 et SSMS, SSRS.

2.1 Création et configuration des machines virtuelles (TD-01)

2.1.1 Télécharger et installer Oracle VirtualBox 6.1.36

Installez d'abord Python 3.10.8 Windows Installer (32-bit)

<https://www.python.org/downloads/windows/>

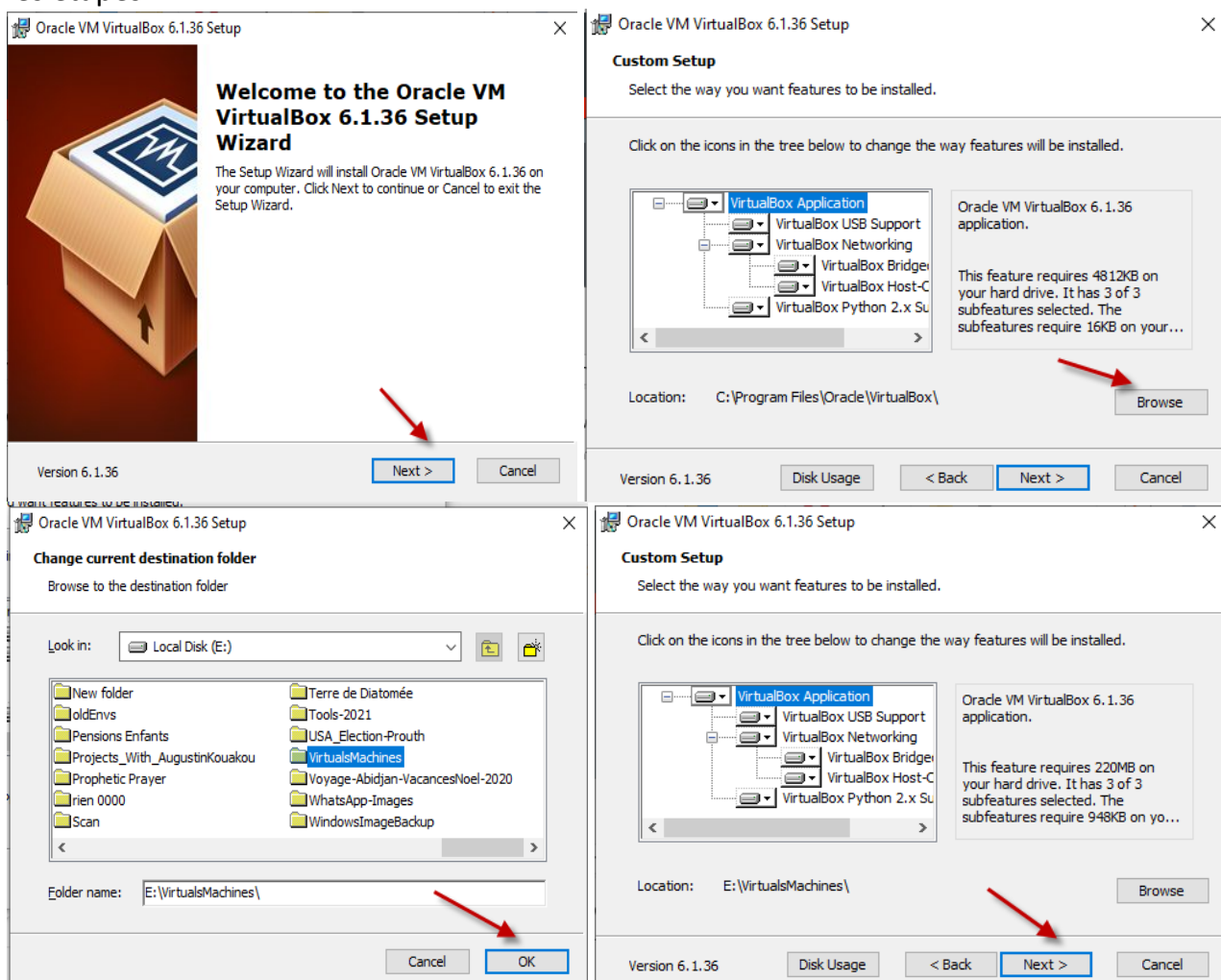
Lien URL: https://www.virtualbox.org/wiki/Download_Old_Builds_6_1

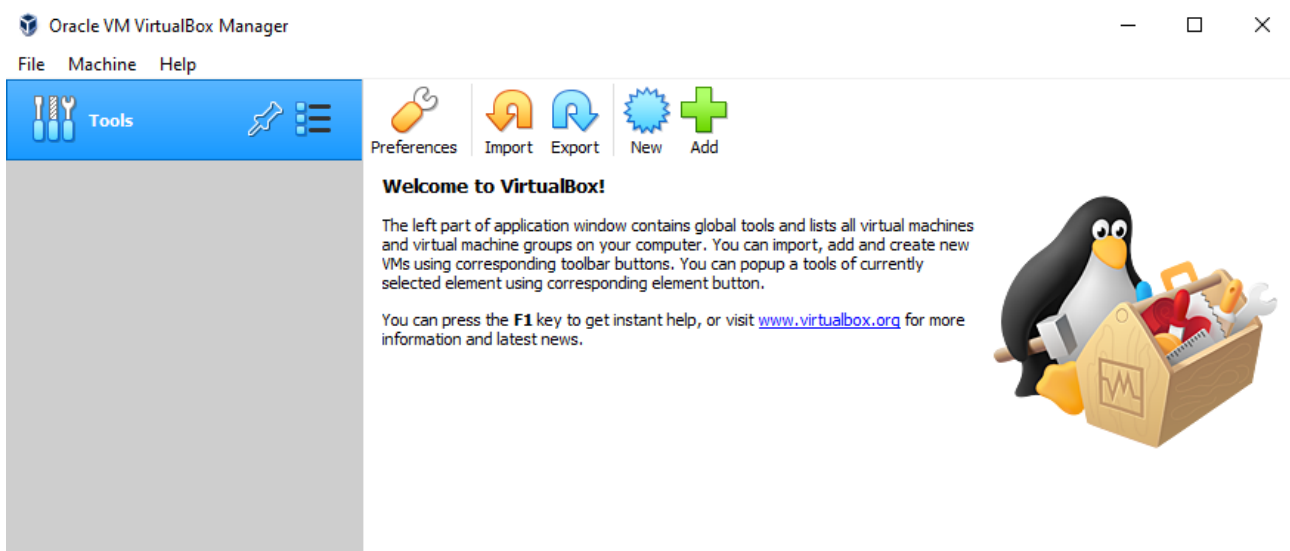
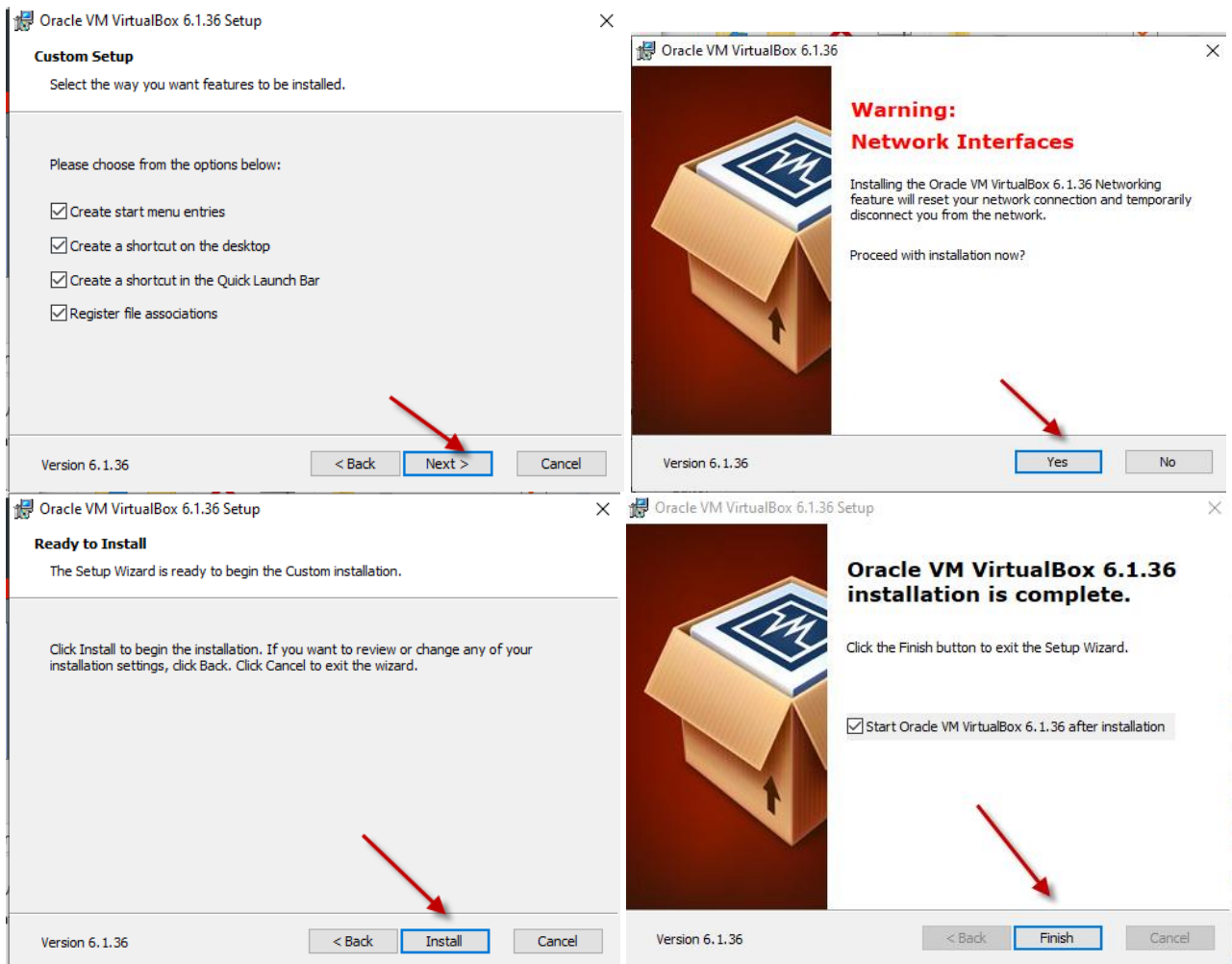
- Dans la section **VirtualBox 6.1.36 (Release July 19 2022)**, cliquez sur **Windows hosts** et sauvez le fichier (**VirtualBox-6.1.36-152435-Win.exe**) dans un nouveau dossier Outils.

Sauvez aussi l'**Extension Pack** dans le même dossier Outils en cliquant sur ce lien :

https://download.virtualbox.org/virtualbox/6.1.36/Oracle_VM_VirtualBox_Extension_Pack-6.1.36a-152435.vbox-extpack

- Cliquez 2 fois sur **VirtualBox-6.1.36-152435-Win.exe** pour lancer l'installation et suivez les étapes.

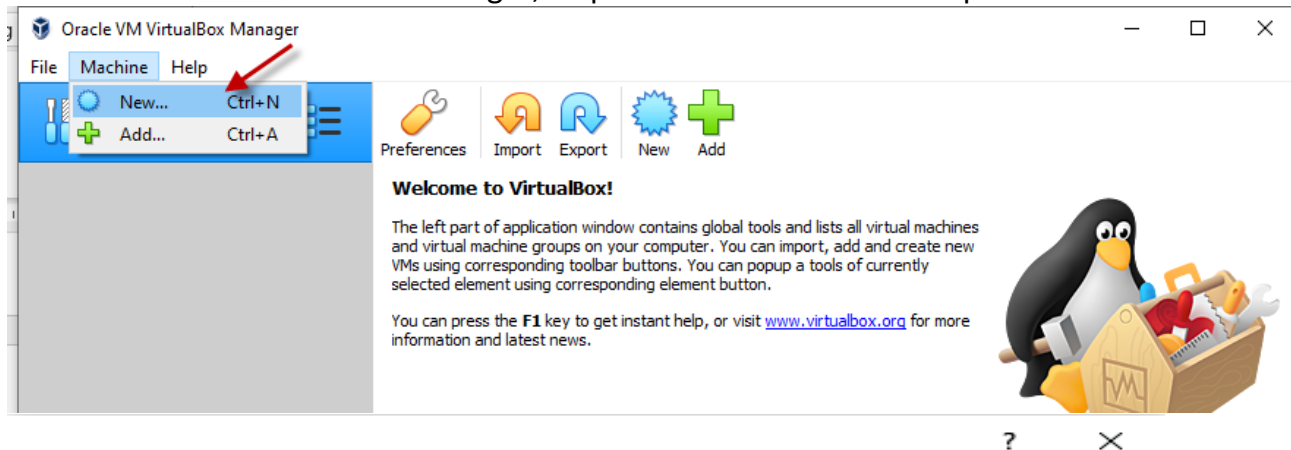




Vous devriez obtenir cette fenêtre ci-haut une fois l'installation réussie.

2.1.2 Créer les machines virtuelles (VM)

- Dans Oracle VM VirtualBox Manager, cliquez sur le menu **Machine** puis sur **New**




← Create Virtual Machine


Name and operating system

Please choose a descriptive name and destination folder for the new virtual machine and select the type of operating system you intend to install on it. The name you choose is used throughout VirtualBox to identify this machine.

1 → Name: 2 →

Machine Folder:

Type: 

Version: 

3 → 4 →

Tapez le nom de la machine virtuelle, et sélectionnez le dossier (choisir **Other** étape 2), puis la Version **Windows 2019 (64 bit)** et cliquez sur Next. Comme indiqué sur la figure.

- La RAM recommandée est 2048 MB mais vous pouvez choisir **4096 MB**.

← Create Virtual Machine

Memory size

Select the amount of memory (RAM) in megabytes to be allocated to the virtual machine.

The recommended memory size is **2048 MB**.

MB

4 MB 32768 MB

← Create Virtual Machine

Hard disk

If you wish you can add a virtual hard disk to the new machine. You can either create a new hard disk file or select one from the list or from another location using the folder icon.

If you need a more complex storage set-up you can skip this step and make the changes to the machine settings once the machine is created.

The recommended size of the hard disk is **50.00 GB**.

☐ Do not add a virtual hard disk

☒ Create a virtual hard disk now

☐ Use an existing virtual hard disk file

← Create Virtual Hard Disk

Hard disk file type

Please choose the type of file that you would like to use for the new virtual hard disk. If you do not need to use it with other virtualization software you can leave this setting unchanged.

- ☒ VDI (VirtualBox Disk Image)
- ☐ VHD (Virtual Hard Disk)
- ☐ VMDK (Virtual Machine Disk)

Storage on physical hard disk

Please choose whether the new virtual hard disk file should grow as it is used (dynamically allocated) or if it should be created at its maximum size (fixed size).

A **dynamically allocated** hard disk file will only use space on your physical hard disk as it fills up (up to a maximum **fixed size**), although it will not shrink again automatically when space on it is freed.

A **fixed size** hard disk file may take longer to create on some systems but is often faster to use.

- ☒ Dynamically allocated
- ☐ Fixed size

Expert Mode

Next

Cancel

Next

Cancel

← Create Virtual Hard Disk

File location and size

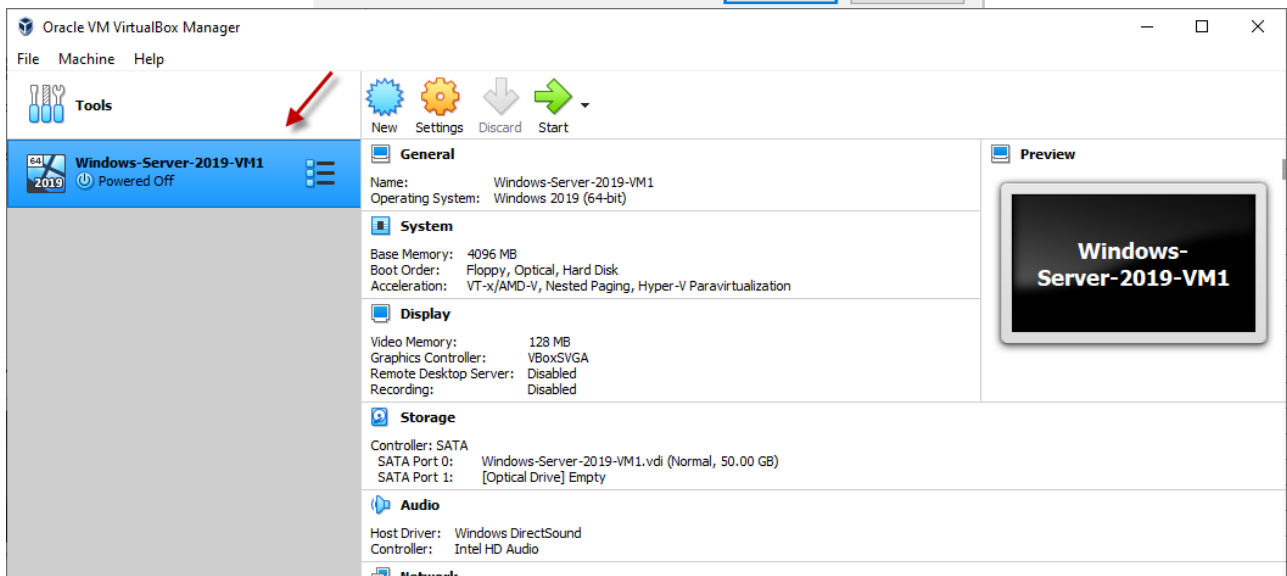
Please type the name of the new virtual hard disk file into the box below or click on the folder icon to select a different folder to create the file in.

Select the size of the virtual hard disk in megabytes. This size is the limit on the amount of file data that a virtual machine will be able to store on the hard disk.

4.00 MB 50.00 GB 2.00 TB

Create

Cancel

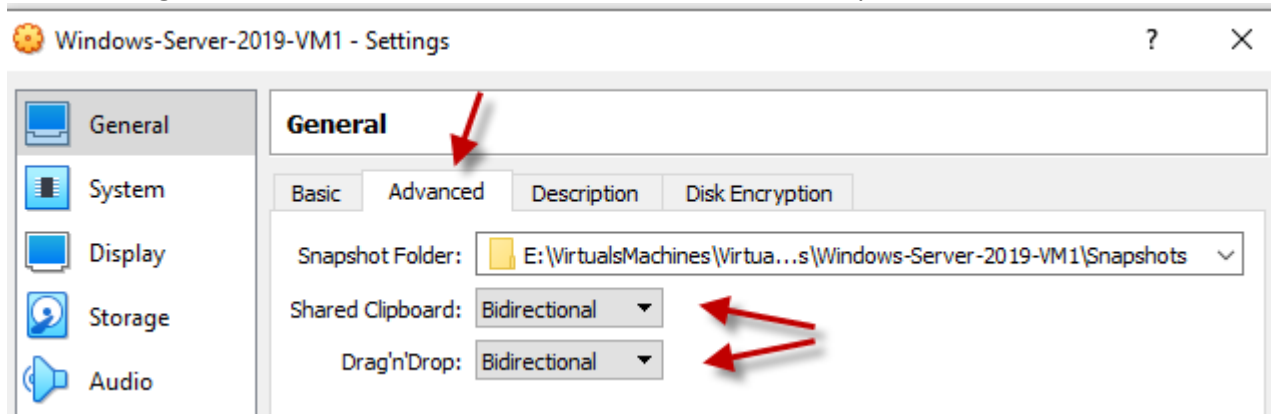


Machine Virtuelle **Windows-Server-2019-VM1** (Powered off) créée avec succès.

2.1.3 Configurer les machines virtuelles

A) Activer le Bidirectionnel du Presse-Papiers et du Glisser-Déplacer.

- Clic-Droit sur Windows-Server-2019-VM1 et choisir **Settings** ou (Ctrl + S)
- Dans l'onglet **Advanced** et choisir **Bidirectional** 2 fois et cliquer OK



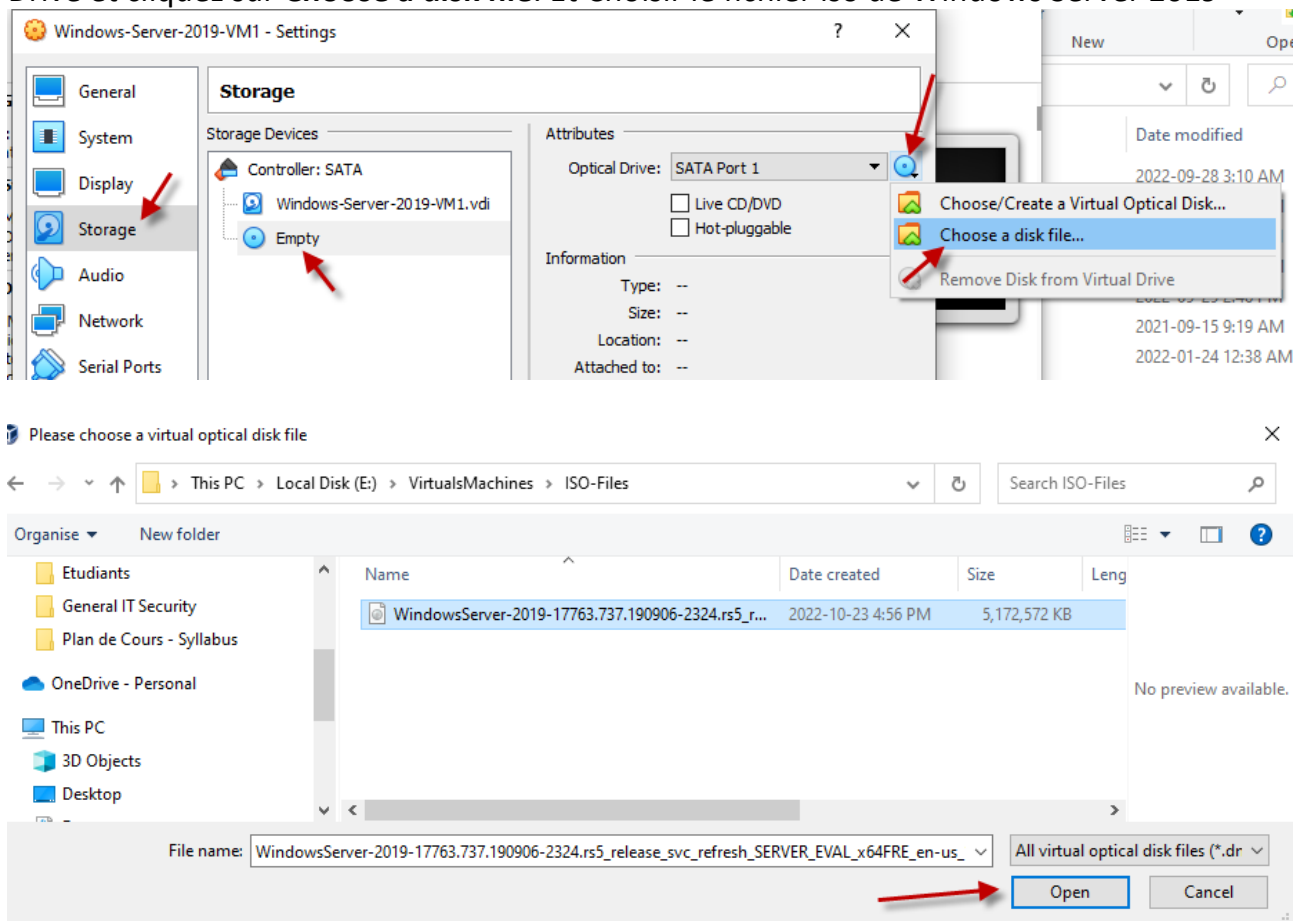
B) Monter le fichier iso Windows Server 2019

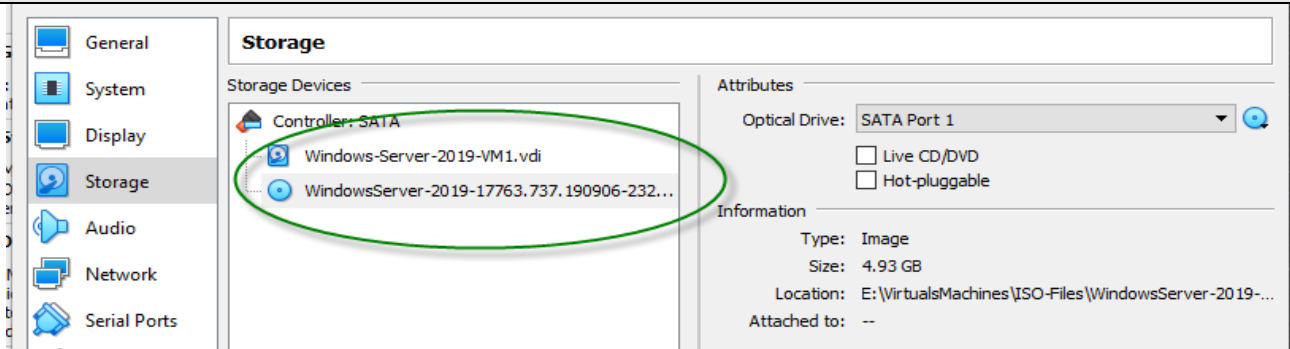
Télécharger le fichier iso de la version d'évaluation de Windows Server 2019 et sauvegarder le dans un sous-dossier **ISO-Files** que vous créez dans votre dossier **VirtualsMachines**.

<https://info.microsoft.com/ww-landing-windows-server-2019.html>

Pour ceux qui n'arriveront pas, demandez au Professeur une copie sur Clef USB.

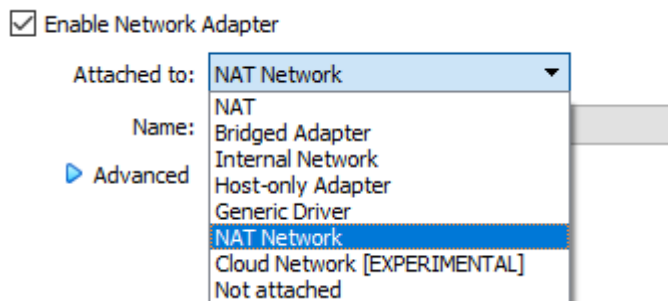
Dans le menu vertical de **Settings**, cliquez sur **Storage > Empty** puis sur l'icône de Optical Drive et cliquez sur **Choose a disk file**. Et Choisir le fichier iso de Windows Server 2019





2.1.4 Configurer le Réseau Virtuel

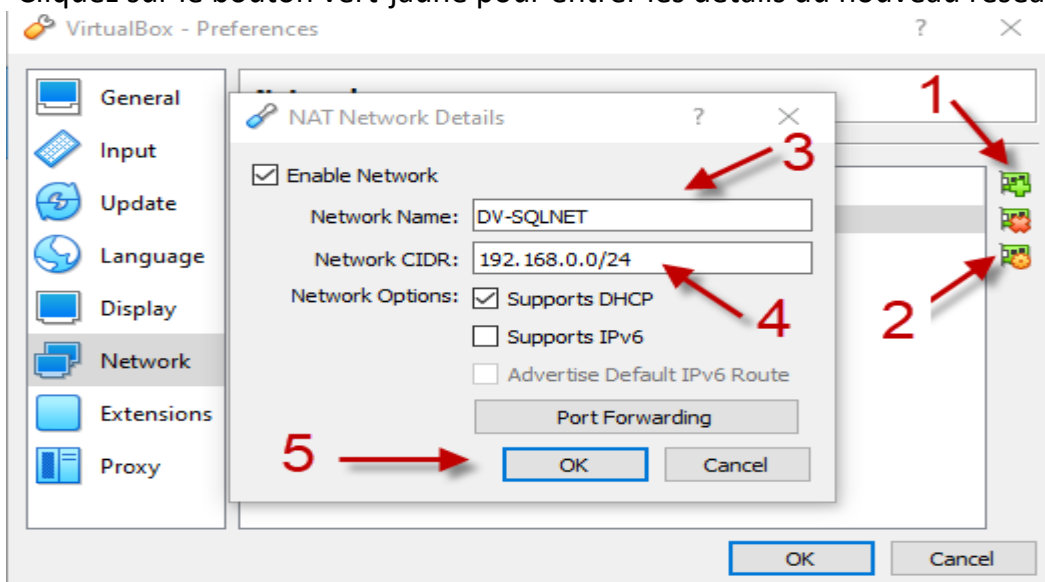
Comme un réseau physique où les serveurs doivent se connecter et communiquer au sein du réseau, les machines virtuelles du laboratoire virtuel doivent également communiquer entre elles via leurs adresses IP uniques ainsi que leur masque de sous-réseau et leur passerelle par défaut. Chaque machine a besoin d'une carte réseau et d'une adresse IP pour communiquer sur un réseau d'ordinateurs. Oracle VM VirtualBox a plusieurs adaptateurs réseaux comme le montre l'image suivante :



Les plus utilisés sont Nat Network, Bridged Adapter et Host-only Adapter

a) Créer un réseau virtuel nommé DV-SQLNET

- Allez dans le menu **Files>Preferences** (Ctrl+G) et Sélectionnez **Network**.
- Cliquez sur le bouton vert Add New NatNetwork pour ajouter un nouveau réseau Nat.
- Cliquez sur le bouton vert-jaune pour entrer les détails du nouveau réseau DV-SQLNET:



b) Créer un Domain Controller (DC)

Comme dans une entreprise réelle, nous allons créer un Domain Controller (DC) qui permet aux administrateurs de gérer les usagers et les ressources de l'Infrastructure technologique. Son rôle sera Active Directory Domain Services (AD DS) pour répondre aux authentications et à la gestion des politiques de groupes et documents (Group Policy) du domaine.

- Créer la machine virtuelle qui va jouer le rôle de Domain Controller (**DV-DC**)

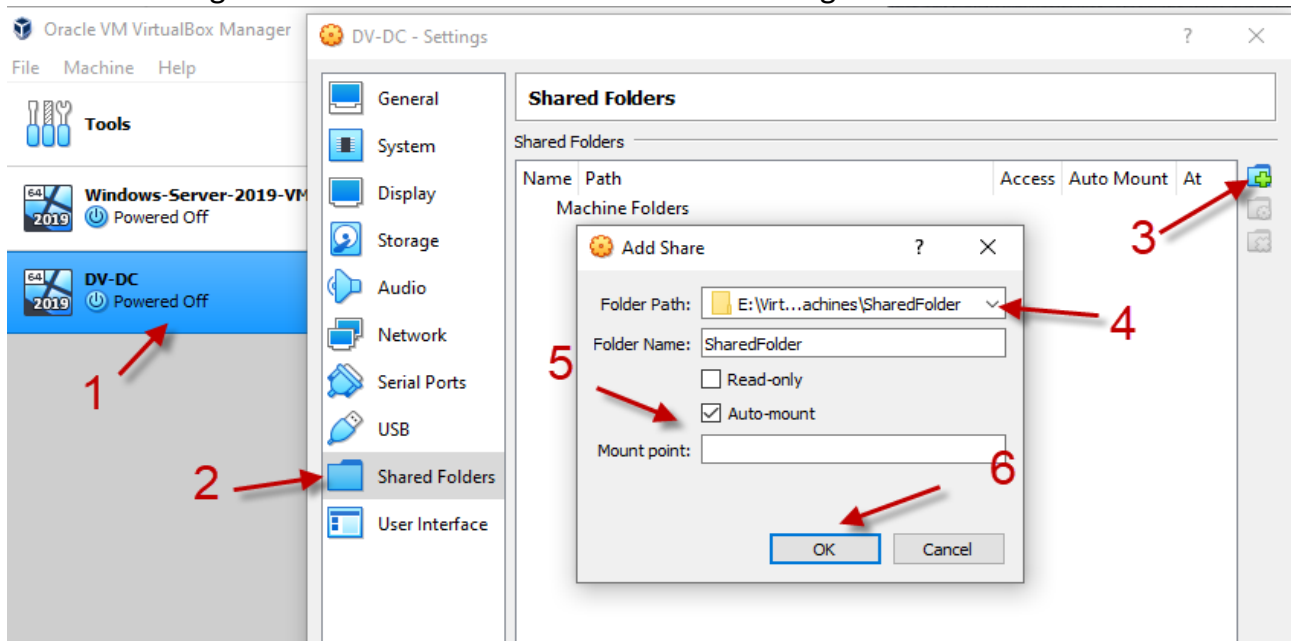
Reprenez les étapes de création de machine virtuelle vues plus haut.

- Configurer le Domain Controller (DV-DC) de façon suivante

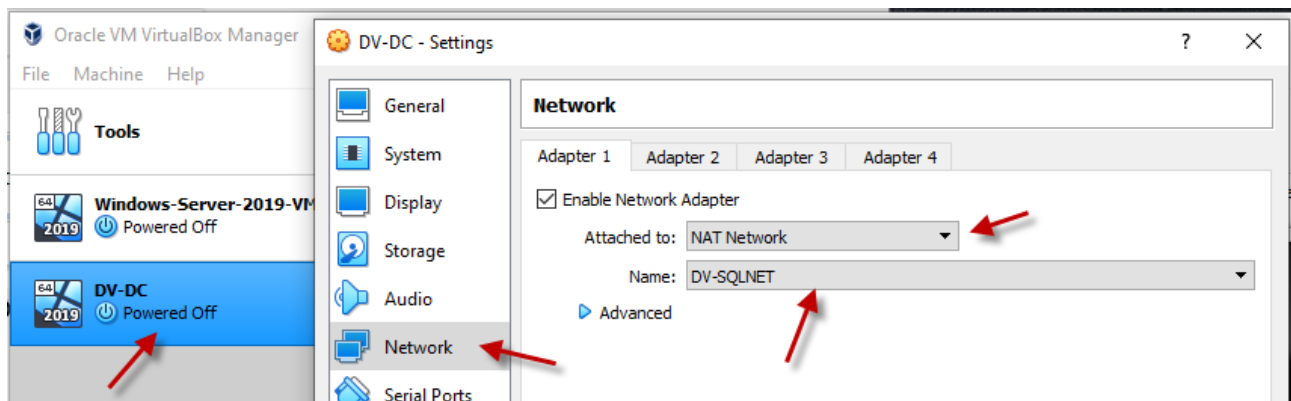
Configurez le presse-papiers partagé, le glissez-déplacer, montez le fichier ISO de Windows Server 2019 et ajouter cette machine au Réseau Virtuel NAT Network et ajoutez un dossier partagé. Répétez les mêmes étapes vues plus haut.

Puis créer un nouveau dossier appelé **SharedFolder** ds **E:\VirtualsMachines**

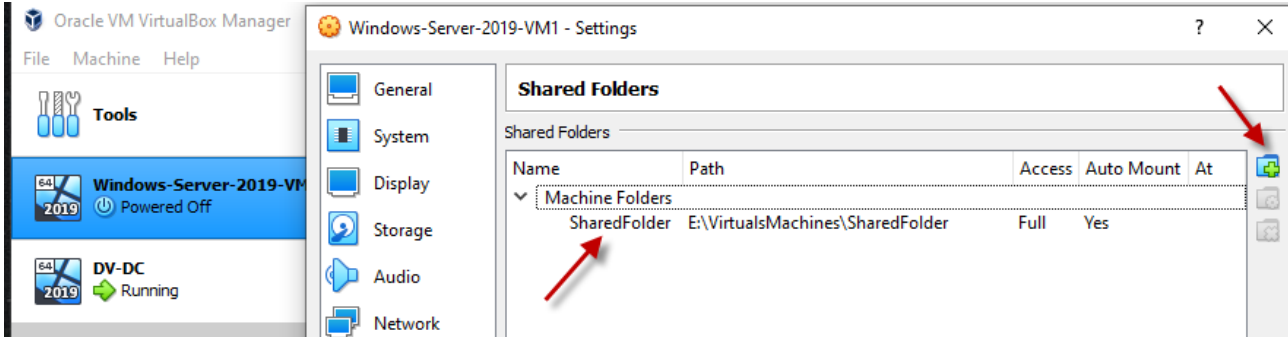
Allez dans l'onglet vertical **Shared Folders** du menu Settings du DV-DC



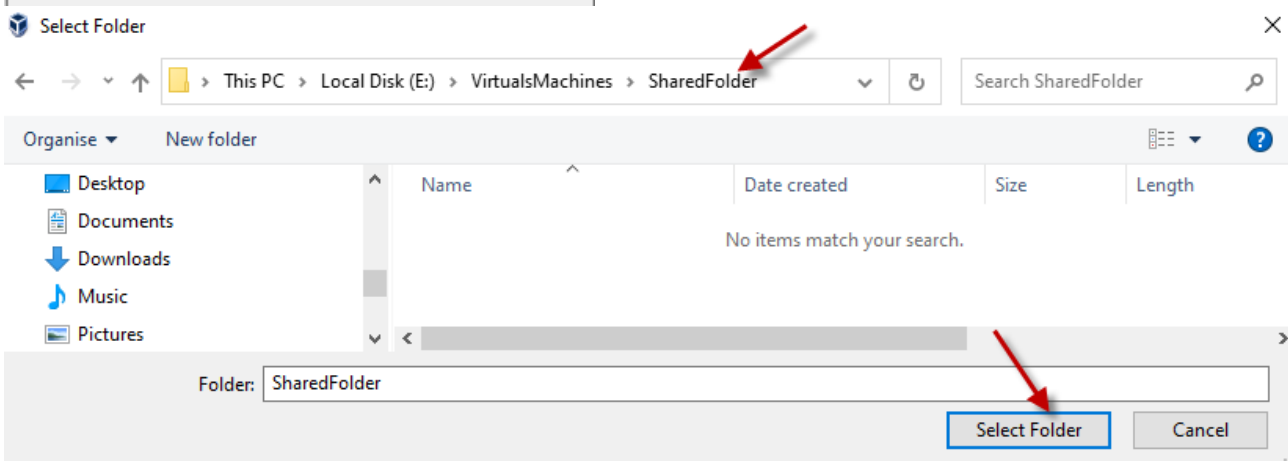
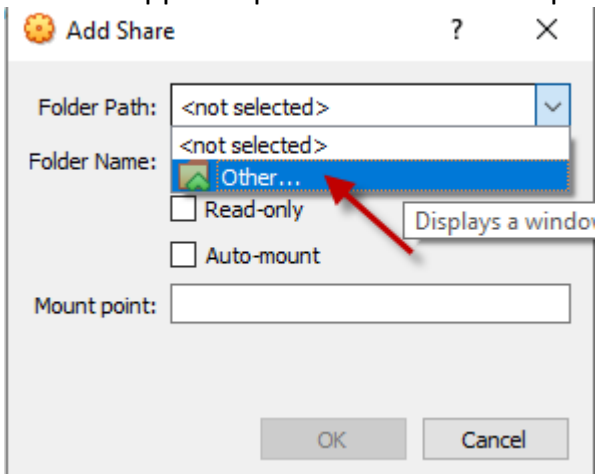
Et maintenant attachez DV-DC au même réseau DV-SQLNET que la toute première machine virtuelle créée : Windows-Server-2019-VM1.



c) Spécifiez les dossiers partagés dans Windows-2019-Server-VM1 aussi
 Assurez vous que Adapter 1 dans le Network de Settings est attaché sur **NAT Network** et le domain **DV-SQLNET**. (**Settings > Network > Adapter 1**)
 Ensuite cliquez sur Settings > Shared Folders > pour ajouter le même dossier créé et partagé plus haut : SharedFolder



Si cela n'apparait pas alors il faudra cliquer sur le bouton Bleu-vert pour l'ajouter.



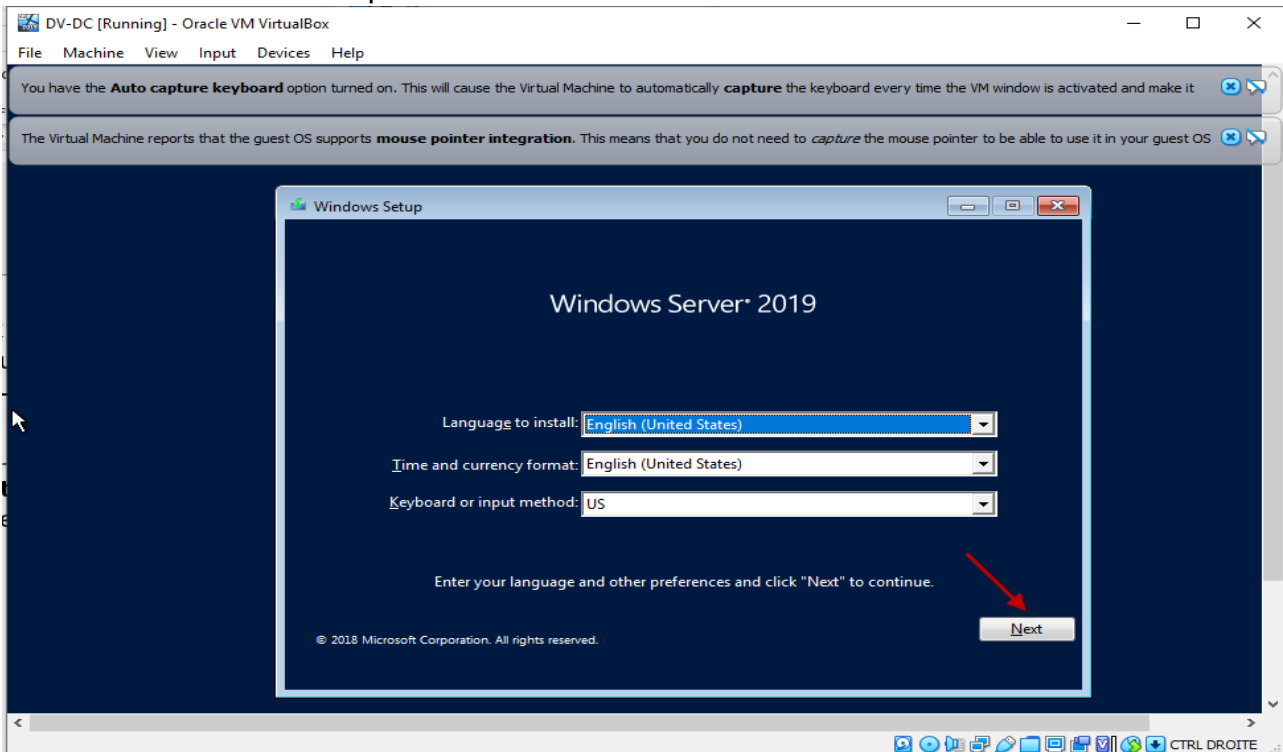
2.2 Installation de Windows 2019 Server dans les Machines Virtuelles

Nous allons installer Windows 2019 Server dans les 2 machines virtuelles déjà créées **DV-DC** et **Windows-Server-2019-VM1** et configurées.

2.2.1 Installation et Configuration de Windows Server 2019 du Domain Controller DV-DC

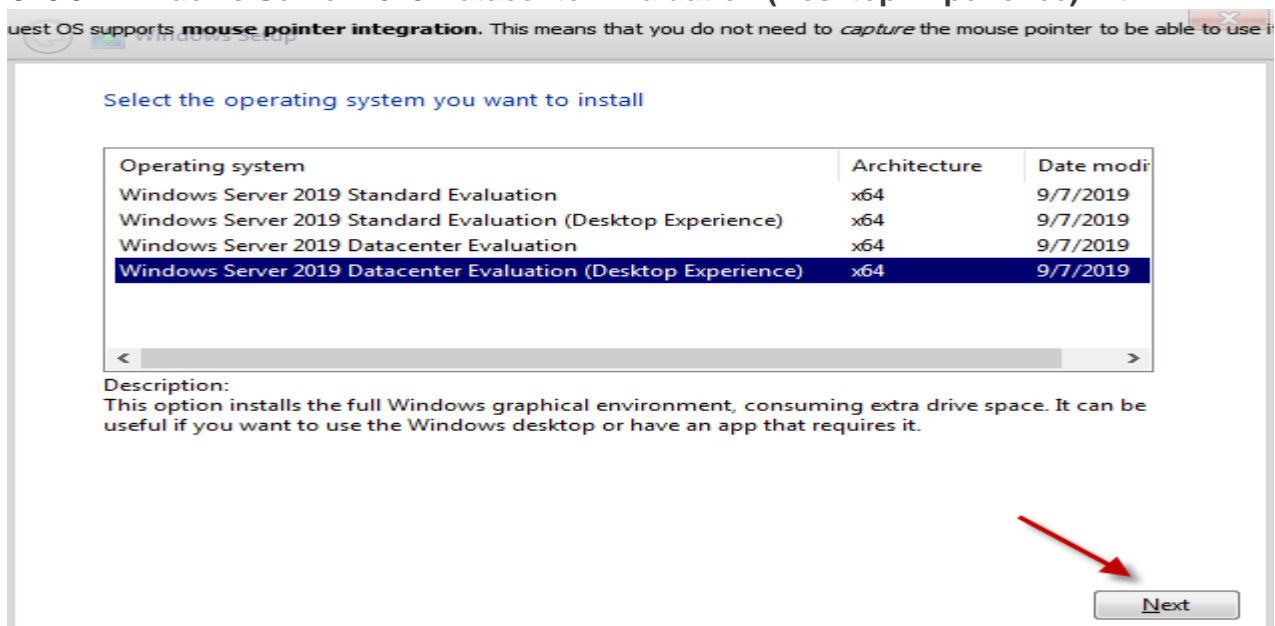
Installation

Sélectionnez **DV-DC** et cliquez sur le bouton vert **START**



Laissez les choix par défaut et cliquez sur **NEXT** puis sur **INSTALL NOW**.

Choisir **Windows Server 2019 Datacenter Evaluation (Desktop Experience)**. Et **NEXT**



Accepter les termes et choisir l'option **Custom: Install Windows only (advanced)**

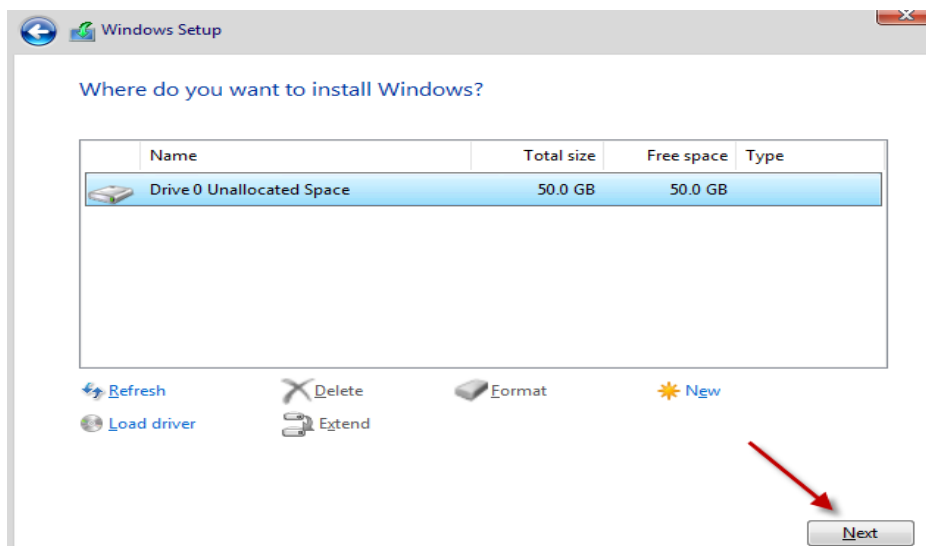
Which type of installation do you want?

Upgrade: Install Windows and keep files, settings, and applications

The files, settings, and applications are moved to Windows with this option. This option is only available when a supported version of Windows is already running on the computer.

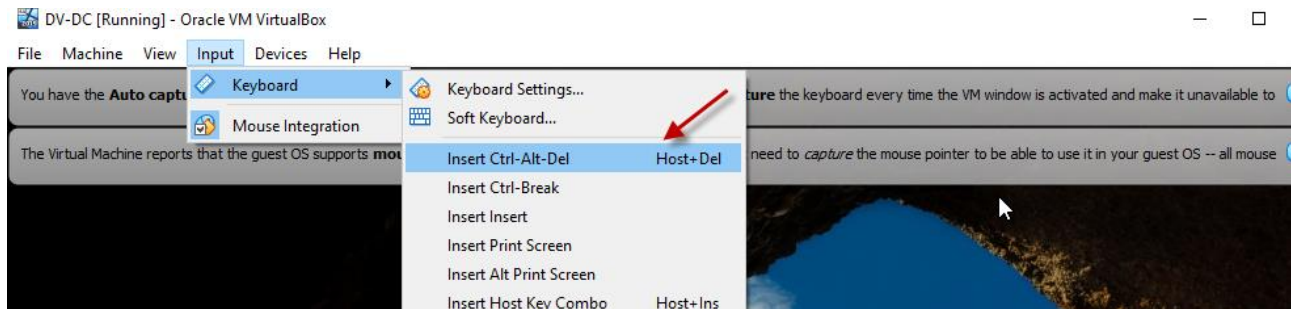
Custom: Install Windows only (advanced)

The files, settings, and applications aren't moved to Windows with this option. If you want to make changes to partitions and drives, start the computer using the installation disc. We recommend backing up your files before you continue.



Et laissez terminer l'installation puis choisir le mot de passe **Robert123** pour le Compte Administrateur et cliquez sur Finish.

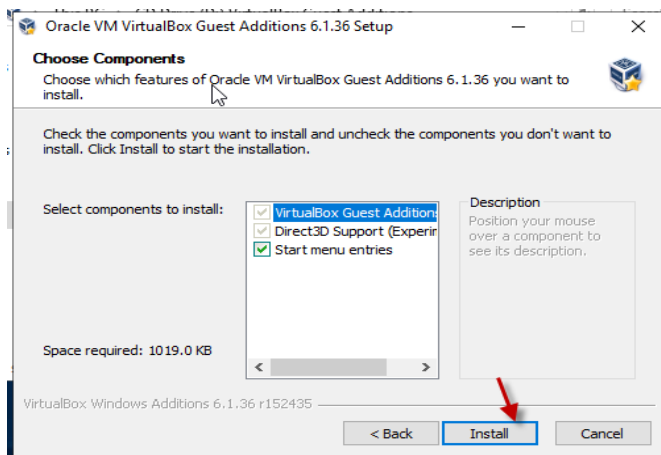
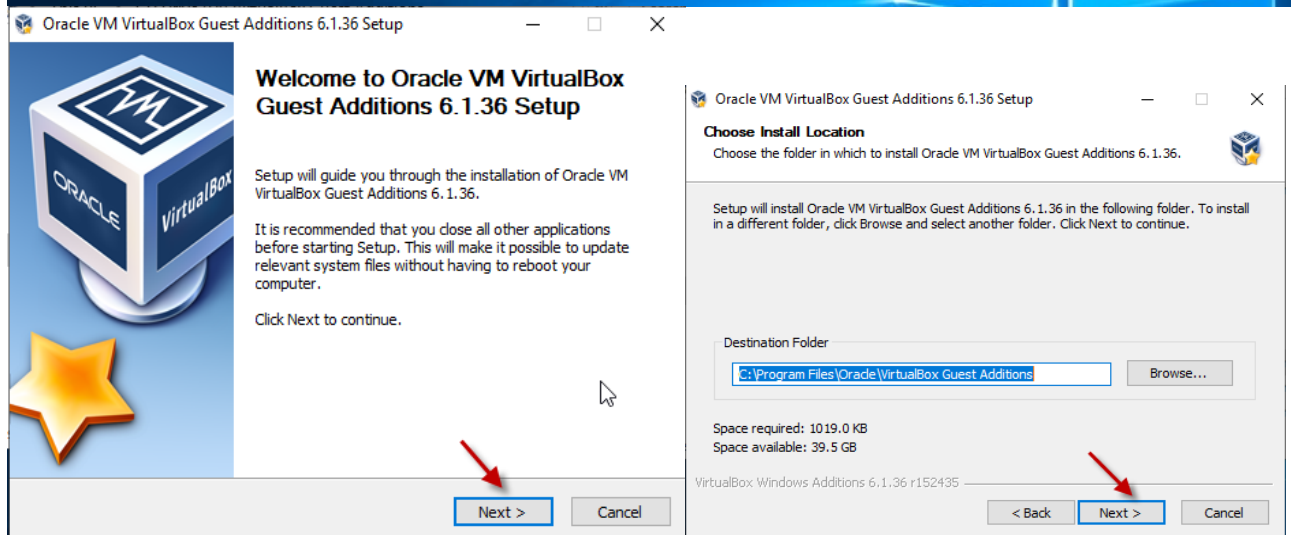
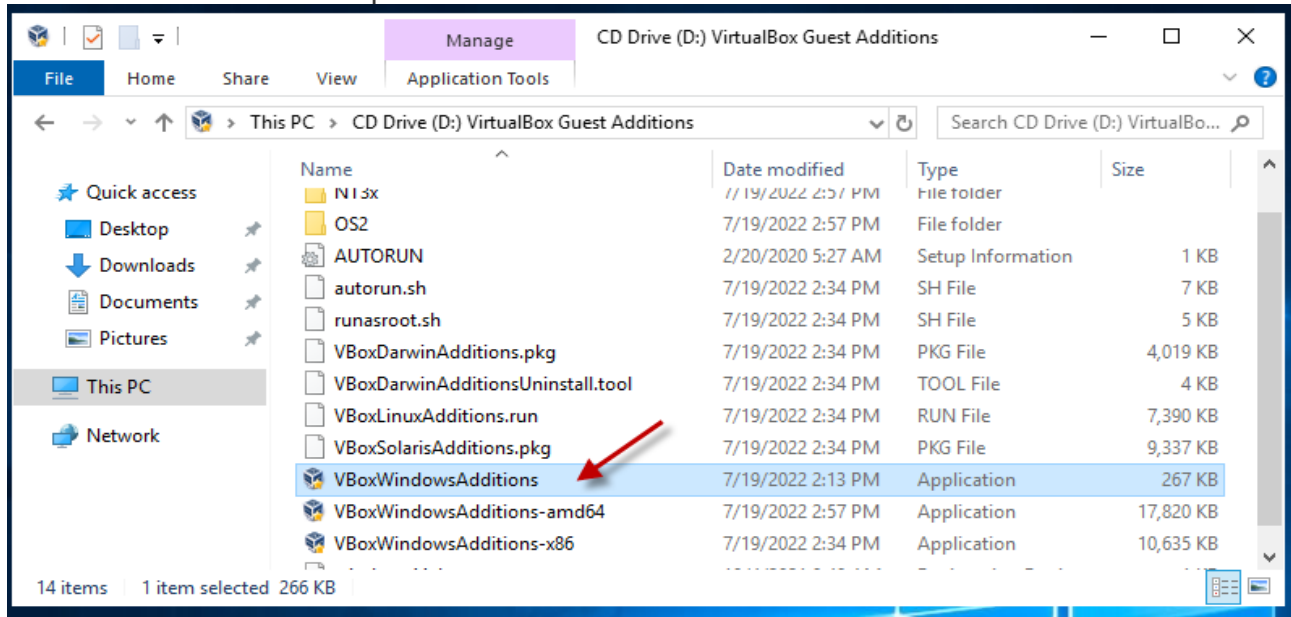
Une fois terminé, appuyez sur le menu **Input>Keyboard>Insert Ctrl+Alt+Del** pour se connecter sur le serveur DC.



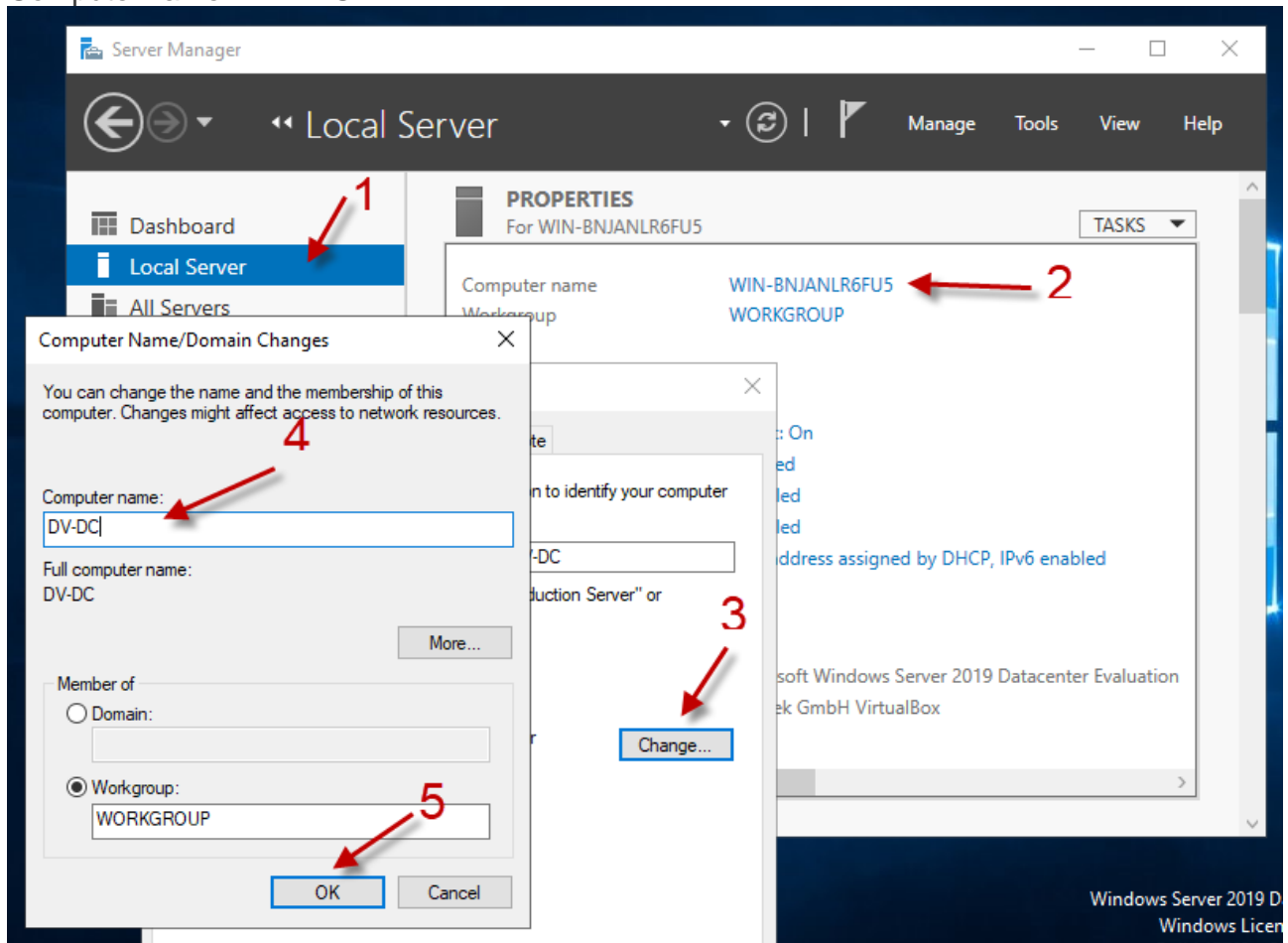
Entrez votre mot de passe : **Robert123** puis Cliquez sur **Yes** dans la boîte de dialogue de confirmation du réseau pour confirmer que vous souhaitez que votre PC soit détectable par d'autres PC et périphériques sur ce réseau.

Installer **Oracle VM VirtualBox Guest Additions** à partir du menu **Devices > Insert Guest Additions CD image**. Puis appuyez sur le bouton **Win+E** du clavier.

Sélectionnez **This PC** dans le **Windows Explorer** qui s'ouvre puis cliquez 2 fois sur **CD Drive (D:) VirtualBox Guest Additions** pour l'ouvrir. Puis cliquez 2 fois sur **VBoxWindowsAdditions** pour lancer l'installation.

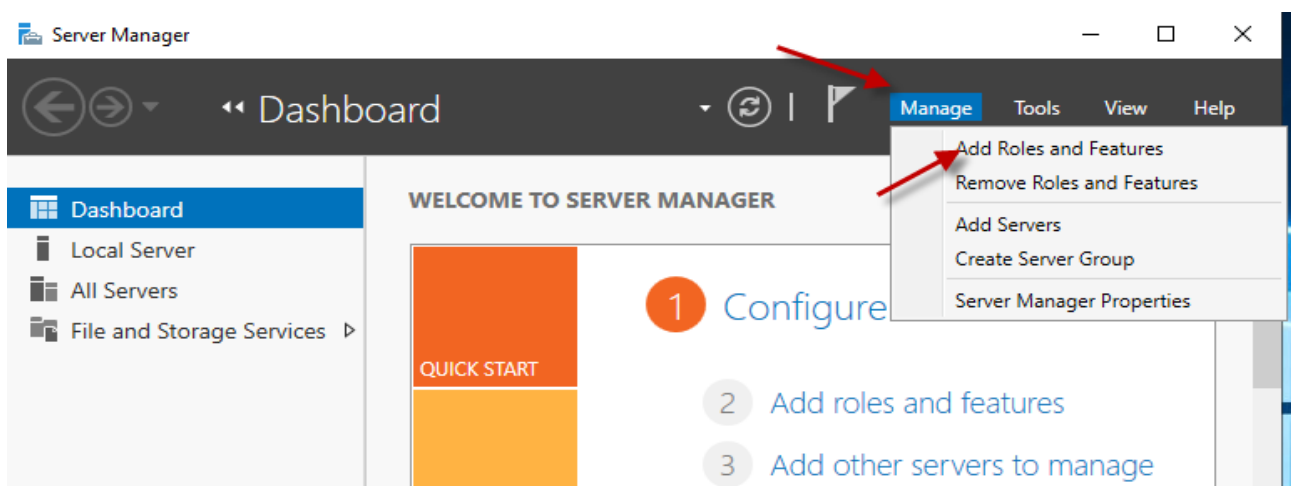


Cliquez sur Finish et le Serveur DV-DC va redémarrer. Se connecter à Nouveau.
 Ouvrir le **Server Manager** de Windows Server 2019 dans DV-DC
 Cliquez sur Local Server à gauche et renommer le nom de l'ordinateur.
 ComputerName = DV-DC



Fermez et cliquez sur **Restart Now**.
 Puis Reconnectez-Vous

Nous allons maintenant créer le domaine **DV-SQLNET.COM** et lui donner le rôle de : **Active Directory Domain Services (AD DS)** et le configurer ensuite en tant que **Domain Controller**.



Sélectionnez **Add Roles and Features** dans **Manage** et Cliquez sur NEXT
 Cliquez sur NEXT puis NEXT encore.

Select installation type

Before You Begin

Installation Type

Server Selection

Server Roles

Features

Confirmation

Select the installation type. You can install roles and features on a running physical computer or machine, or on an offline virtual hard disk (VHD).

☒ **Role-based or feature-based installation**

Configure a single server by adding roles, role services, and features.

☐ **Remote Desktop Services installation**

Install required role services for Virtual Desktop Infrastructure (VDI) to create a virtual machine or session-based desktop deployment.

Select destination server

Before You Begin

Installation Type

Server Selection

Server Roles

Features

Confirmation

Results

Select a server or a virtual hard disk on which to install roles and features.

☒ Select a server from the server pool

☐ Select a virtual hard disk

Server Pool

Filter:		
Name	IP Address	Operating System
DV-DC	192.168.0.4	Microsoft Windows Server 2019 Datacenter Evaluation

1 Computer(s) found

This page shows servers that are running Windows Server 2012 or a newer release of Windows Server, and that have been added by using the Add Servers command in Server Manager. Offline servers and newly-added servers from which data collection is still incomplete are not shown.

< Previous

Next >

Install

Cancel

Important de choisir le bon Rôle ici. Donc cliquez sur : **Active Directory Domain Services**

Before You Begin

Installation Type

Server Selection

Server Roles

Features

Confirmation

Select one or more roles to install on the selected server.

Roles

- ☐ Active Directory Certificate Services
- ☒ **Active Directory Domain Services**
- ☐ Active Directory Federation Services
- ☐ Active Directory Lightweight Directory Services
- ☐ Active Directory Rights Management Services

Description

Active Directory Domain Services (AD DS) stores information about objects on the network and makes this information available to users and network administrators. AD DS

Puis cliquez sur **Add Features** et puis **NEXT**

Add Roles and Features Wizard

Add features that are required for Active Directory Domain Services?

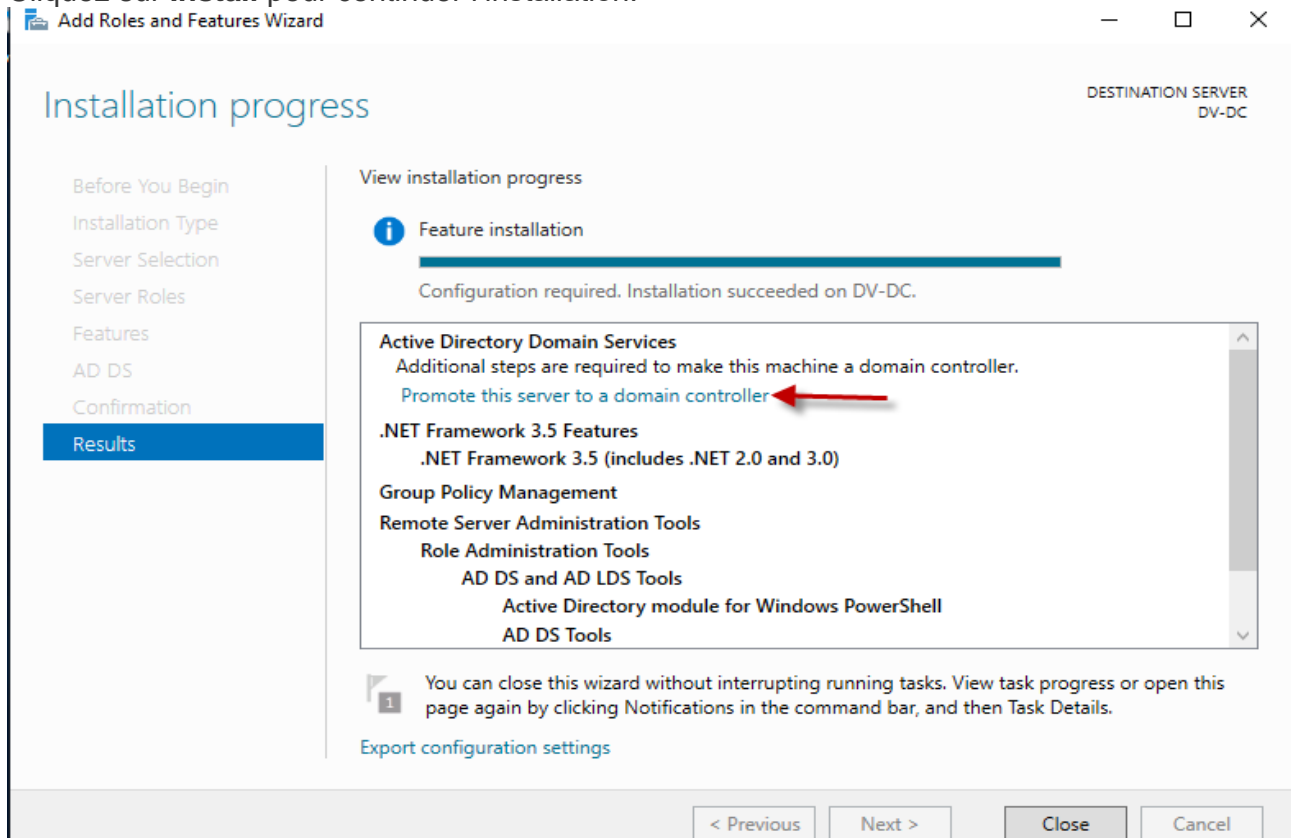
You cannot install Active Directory Domain Services unless the following role services or features are also installed.

- [Tools] Group Policy Management
- Remote Server Administration Tools
 - Role Administration Tools
 - AD DS and AD LDS Tools
 - Active Directory module for Windows PowerShell
 - AD DS Tools
 - [Tools] Active Directory Administrative Center
 - [Tools] AD DS Snap-Ins and Command-Line Tools

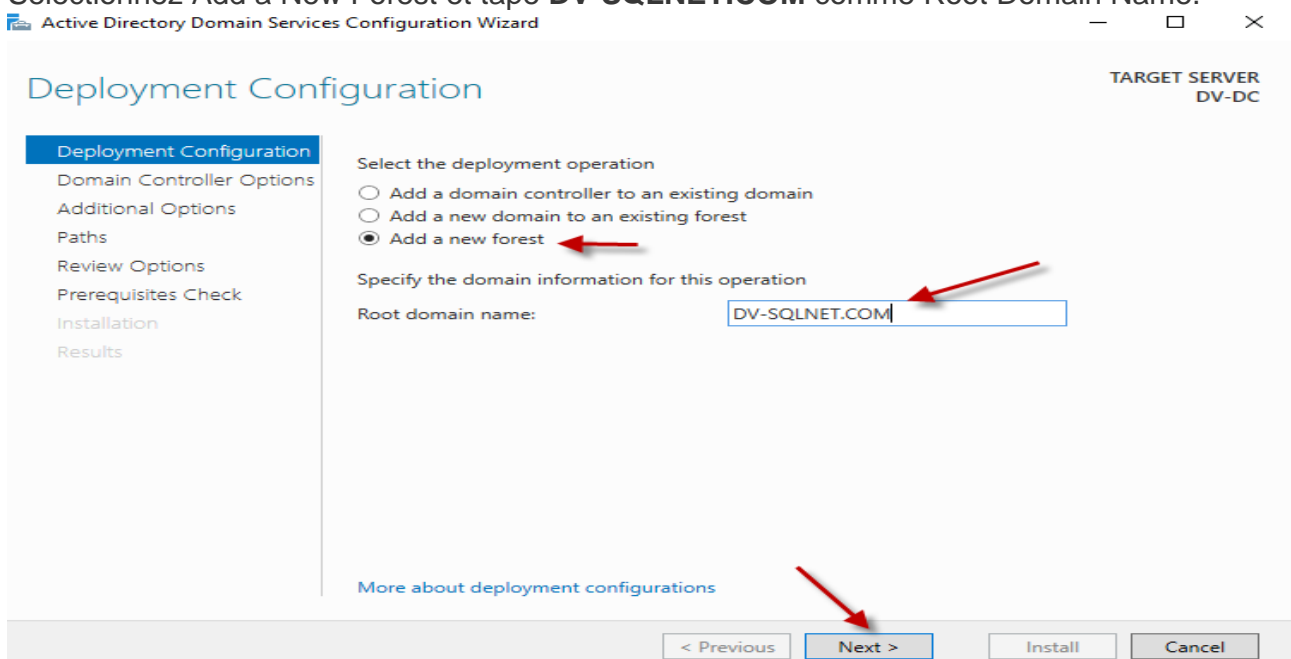
☒ Include management tools (if applicable)

Add Features Cancel

Cochez **.NET Framework 3.5 Deatures** et cliquez sur NEXT
 Cliquez encore sur **NEXT** car **AD DS** est déjà sélectionné
 Cliquez sur **Install** pour continuer l'installation.



Cliquez sur **Promote this server to a domain controller**
 Sélectionnez **Add a New Forest** et tape **DV-SQLNET.COM** comme Root Domain Name.



Cliquez sur NEXT
 Entrez le mot de passe **Robert123** pour le (DSRM) puis cliquez **NEXT**

Active Directory Domain Services Configuration Wizard

Domain Controller Options

TARGET SERVER
DV-DC

Deployment Configuration

Domain Controller Options

DNS Options

Additional Options

Paths

Review Options

Prerequisites Check

Installation

Results

Select functional level of the new forest and root domain

Forest functional level: Windows Server 2016

Domain functional level: Windows Server 2016

Specify domain controller capabilities

☒ Domain Name System (DNS) server

☒ Global Catalog (GC)

☐ Read only domain controller (RODC)

Type the Directory Services Restore Mode (DSRM) password

Password:

Confirm password:

[More about domain controller options](#)

< Previous
Next >
Install
Cancel

Laissez le NetBIOS domain name a **DV-SQLNET** puis cliquez sur NEXT.

Additional Options

TARGET SERVER
DV-DC

Deployment Configuration

Domain Controller Options

DNS Options

Additional Options

Paths

Review Options

Verify the NetBIOS name assigned to the domain and change it if necessary

The NetBIOS domain name: DV-SQLNET

Laissez les chemins par défaut pour le dossier de la base de données et du fichier journal à **C:\Windows\NTDS** et le dossier **C:\Windows\SYSVOL** et cliquez sur **NEXT**.

Review Options cliquez sur **NEXT**

Prerequisites Check : Vérifiez si vous voyez le text suivant : **“All prerequisite checks passed successfully. Click ‘Install’ to begin installation.”** Avant de cliquer sur **INSTALL**.

Prerequisites Check

TARGET SERVER
DV-DC

Deployment Configuration

Domain Controller Options

DNS Options

Additional Options

Paths

Review Options

Prerequisites Check

Installation

Results

✔ All prerequisite checks passed successfully. Click 'Install' to begin installation.
 Show more

Prerequisites need to be validated before Active Directory Domain Services is installed on this computer

[Rerun prerequisites check](#)

[View results](#)

adapter, both IPv4 and IPv6 static IP addresses should be assigned to both IPv4 and IPv6 Properties of the physical network adapter. Such static IP address(es) assignment should be done to all the physical network adapters for reliable Domain Name System (DNS) operation.

⚠ A delegation for this DNS server cannot be created because the authoritative parent zone cannot be found or it does not run Windows DNS server. If you are integrating with an existing DNS infrastructure, you should manually create a delegation to this DNS server in the parent zone to ensure reliable name resolution from outside the domain "DV-SQLNET.COM". Otherwise, no action is required.

i Prerequisites Check Completed

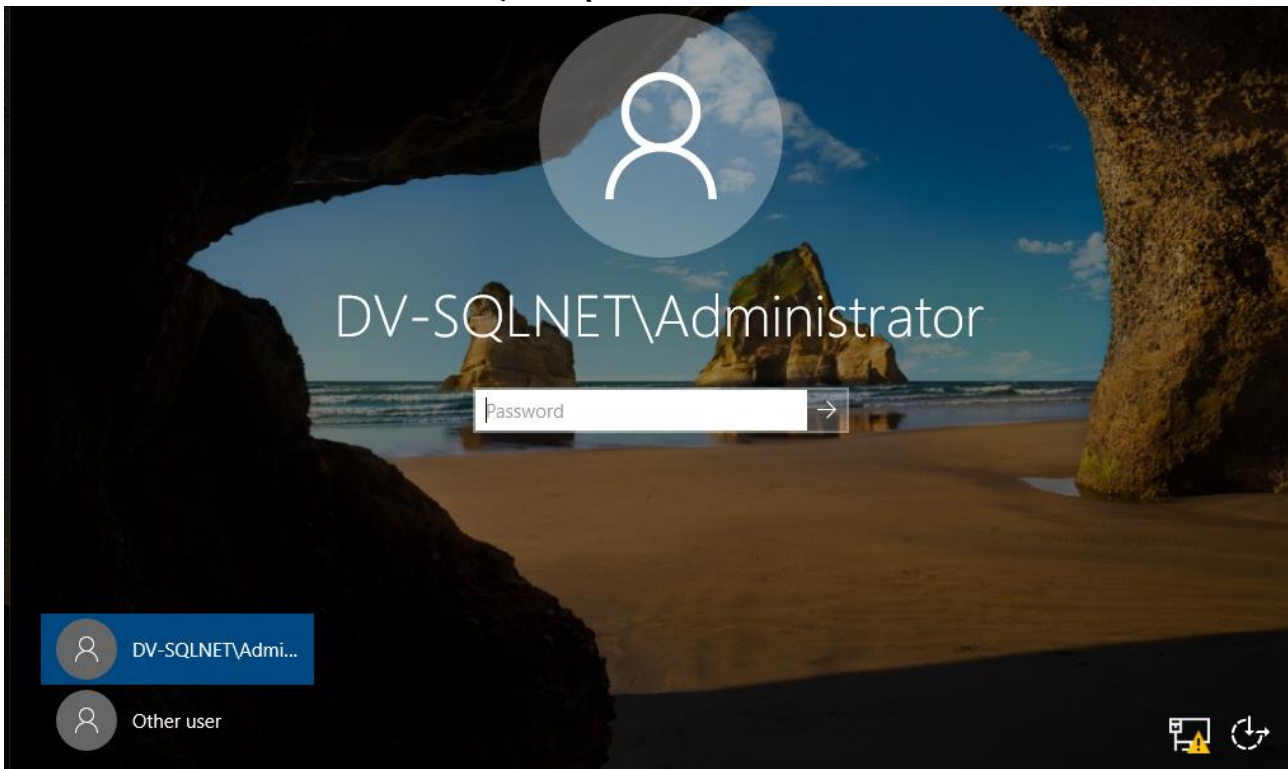
✔ All prerequisite checks passed successfully. Click 'Install' to begin installation.

⚠ If you click Install, the server automatically reboots at the end of the promotion operation.

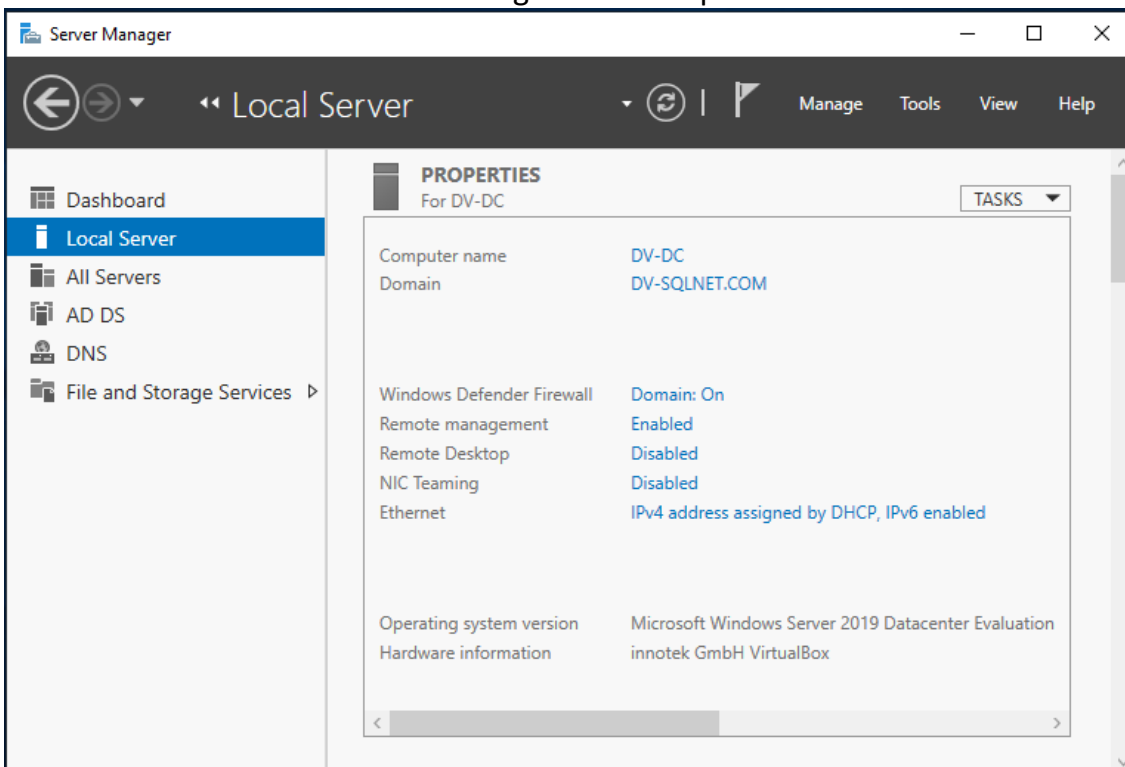
[More about prerequisites](#)

< Previous
Next >
Install
Cancel

L'installation va continuer et un message puis redémarrer automatiquement toute seule.
Se connecter à nouveau et là on voit très bien que l'écran change. Nous maintenant donc
Dans un Domain Controller : **DV-SQLNET\Administrator**

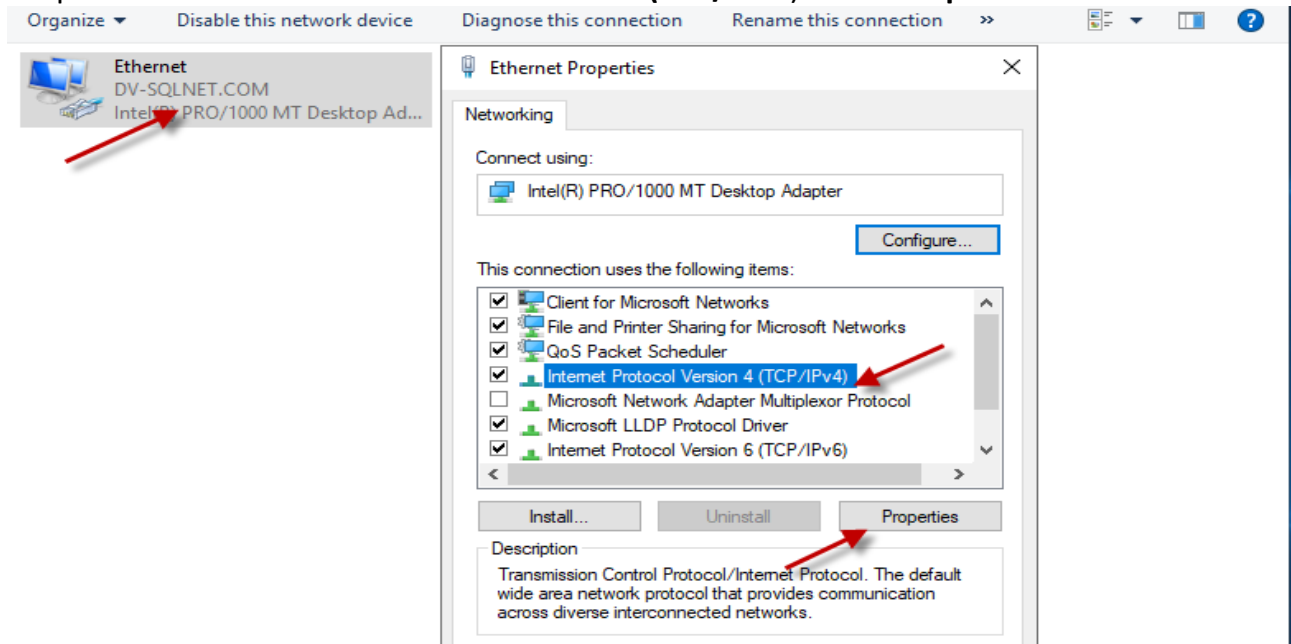


Bravo ! Si vous avez réussi jusqu'ici. Entrez votre mot de passe pour vous connecter et dans le Local Server du Server Manager. Vérifiez que vous avez ce même écran.

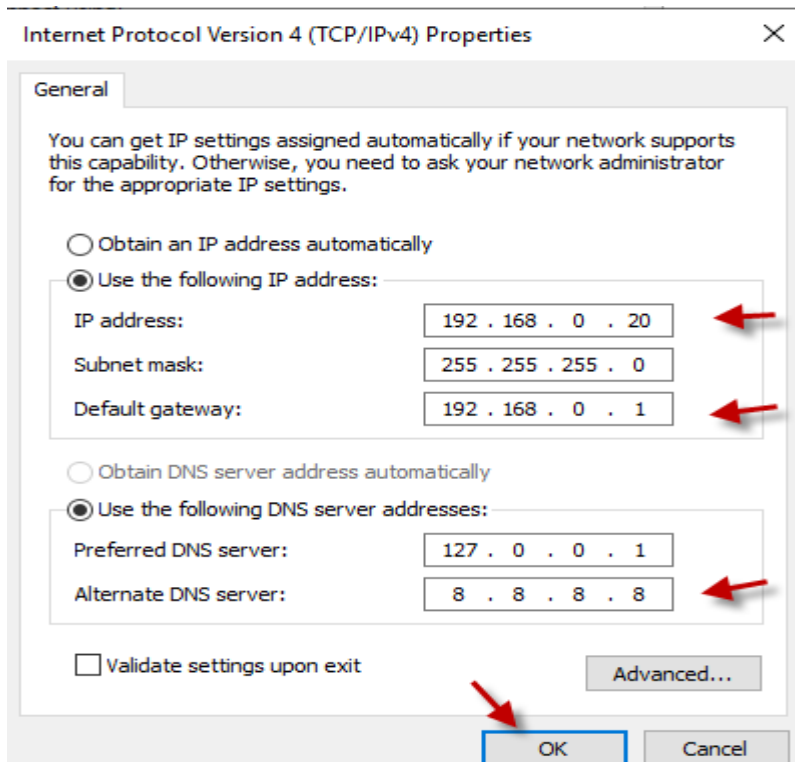


Nous allons assigner des Adresses IP static au DV-DC du domain DV-SQLNET.COM

Cliquez sur **IPv4 address assigned by DHCP, IPv6 Enabled** puis Click Droit sur **Ethernet** puis **Properties** et sur **Internet Protocol Version 4 (TCP/IPv4)** et sur **Properties**.

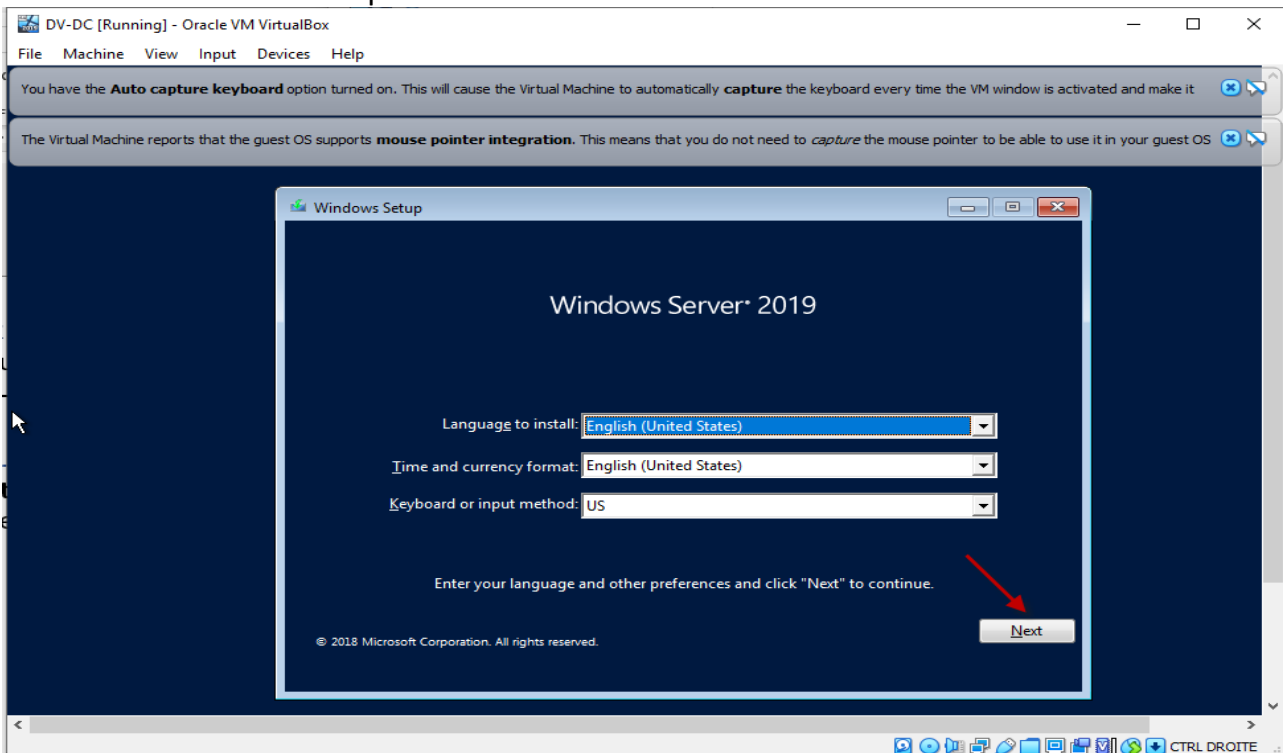


Choisissez **Use the Following IP address** et tapez les valeurs suivants comme à l'écran et **OK** et **CLOSE**.



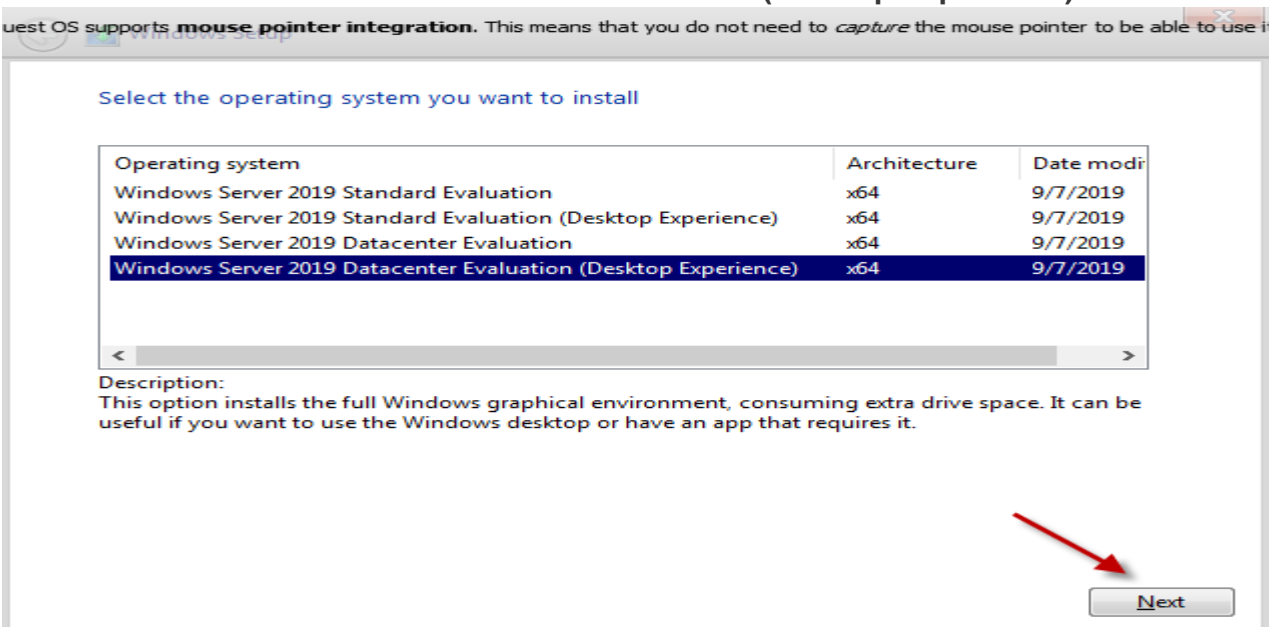
2.2.3 Installation et Configuration de Windows-Server-2019-VM1

Sélectionnez **DV-DC** et cliquez sur le bouton vert **START**



Laissez les choix par défaut et cliquez sur **NEXT** puis sur **INSTALL NOW**.

Choisir **Windows Server 2019 Datacenter Evaluation (Desktop Experience)**. Et **NEXT**



Accepter les termes et choisir l'option **Custom: Install Windows only (advanced)**

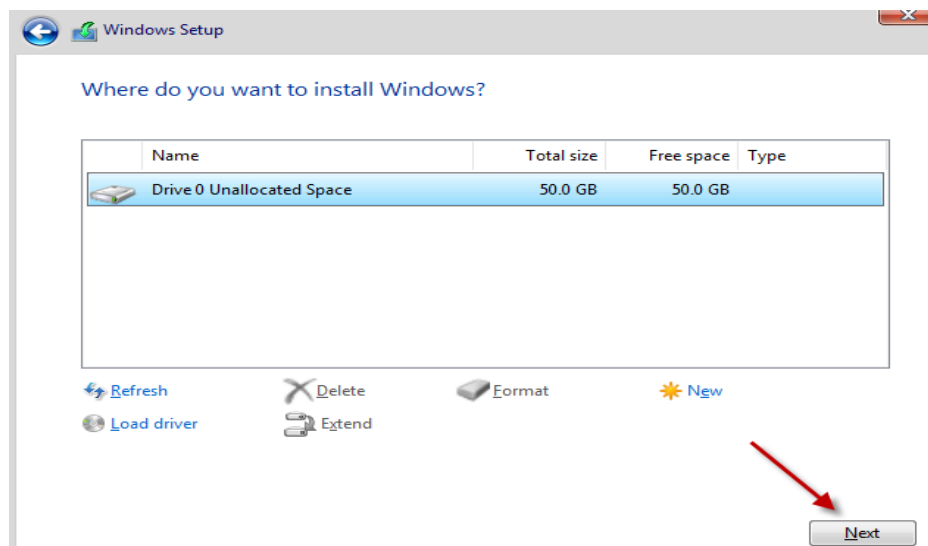
Which type of installation do you want?

Upgrade: Install Windows and keep files, settings, and applications

The files, settings, and applications are moved to Windows with this option. This option is only available when a supported version of Windows is already running on the computer.

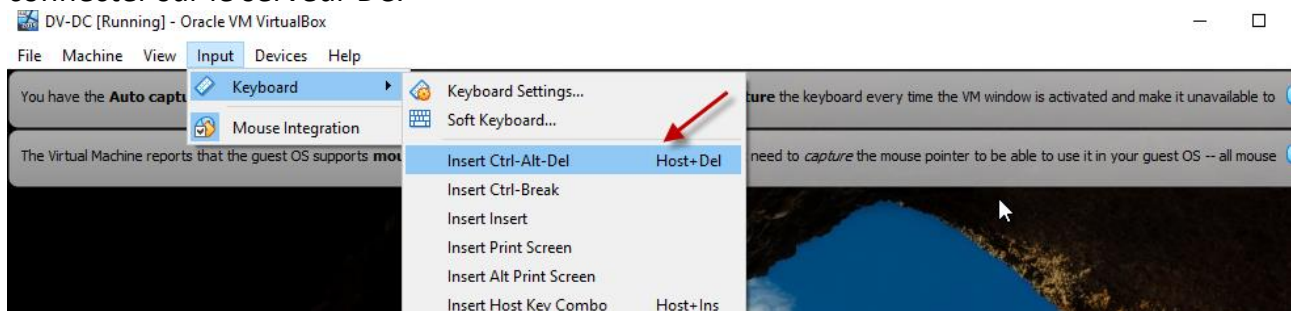
Custom: Install Windows only (advanced)

The files, settings, and applications aren't moved to Windows with this option. If you want to make changes to partitions and drives, start the computer using the installation disc. We recommend backing up your files before you continue.



Et laissez terminer l'installation puis choisir le mot de passe **Robert123** pour le Compte Administrateur et cliquez sur Finish.

Une fois terminé, appuyez sur le menu **Input>Keyboard>Insert Ctrl+Alt+Del** pour se connecter sur le serveur DC.

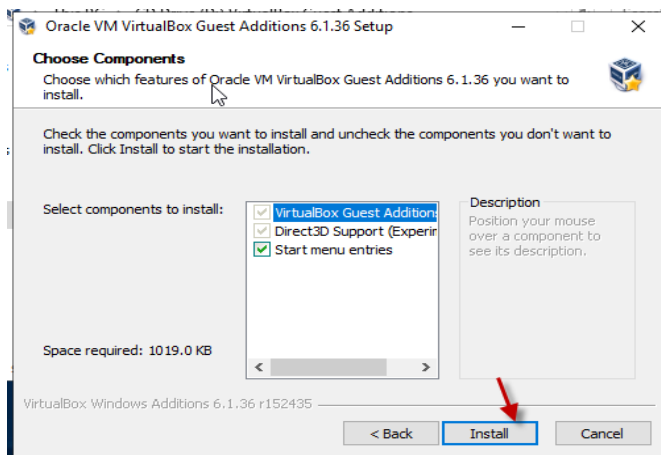
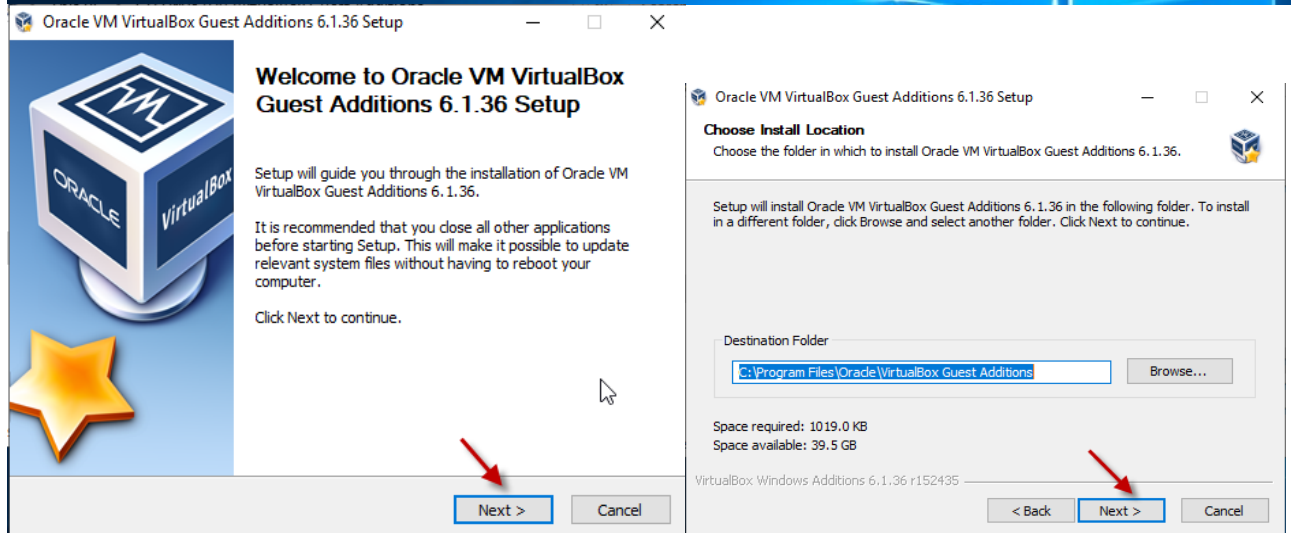
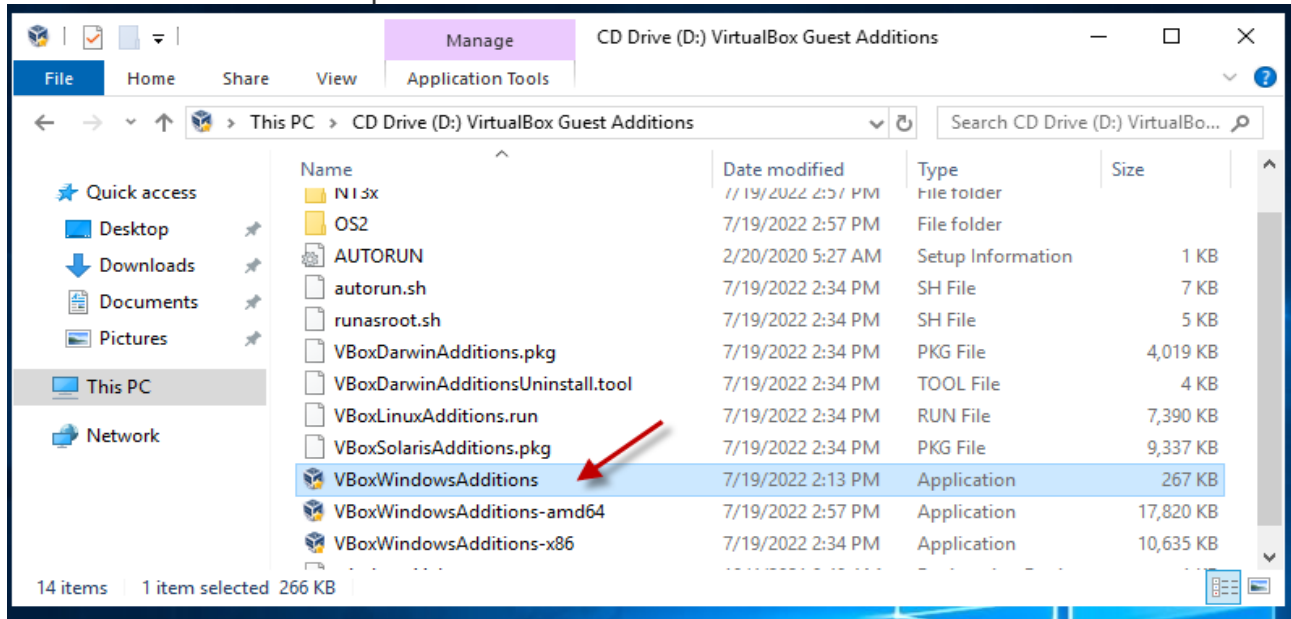


Entrez votre mot de passe : **Robert123** puis Cliquez sur **Yes** dans la boîte de dialogue de confirmation du réseau pour confirmer que vous souhaitez que votre PC soit détectable par d'autres PC et périphériques sur ce réseau.

L'installation du serveur Win 2019 Server est installé nous allons maintenant installer le VirtualBox Guest Additions comme précédemment dans le DV-DC.

Installer **Oracle VM VirtualBox Guest Additions** à partir du menu **Devices > Insert Guest Additions CD image**. Puis appuyez sur le bouton **Win+E** du clavier.

Sélectionnez This PC dans le Windows Explorer qui s'ouvre puis cliquez 2 fois sur **CD Drive (D:) VirtualBox Guest Additions** pour l'ouvrir. Puis cliquez 2 fois sur **VBoxWindowsAdditions** pour lancer l'installation.



Cliquez sur **Finish** et le Windows-Server-2019-VM1 va redémarrer.
Se connecter à Nouveau.

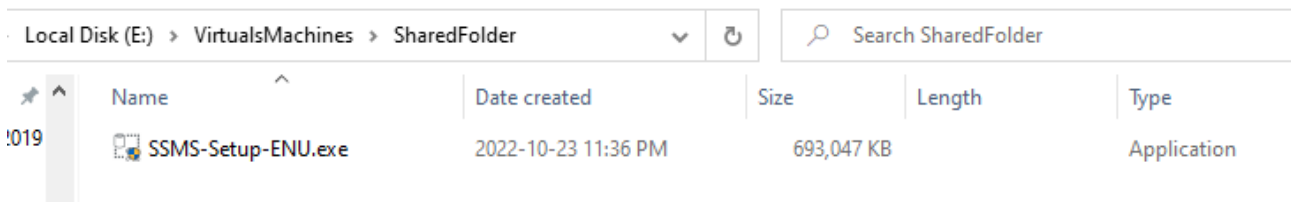
2.2.4 Télécharger le SQL Server Management Studio (SSMS) sur votre PC

Nous allons donc télécharger l'utilitaire de gestion de SQL Server dans notre ordinateur local PC (Laptop ou Desktop) et non dans les machines virtuelles.

Lien URL: <https://aka.ms/ssmsfullsetup>

Sauvez le fichier **SSMS-Setup-ENU.exe** dans un dossier de votre PC.

Ensuite allez **couper et coller** ce fichier **SSMS-Setup-ENU.exe** dans le dossier **SharedFolder** créé plutôt. Cela nous permettra d'avoir accès à ce fichier à partir de notre machine virtuelle.



Et maintenant on passe à l'installation de MS SQL Server 2019 dans notre serveur Windows-Server-2019-VM1 machine virtuelle.

3. Installation et configuration de base (MS SQL Server 2019)

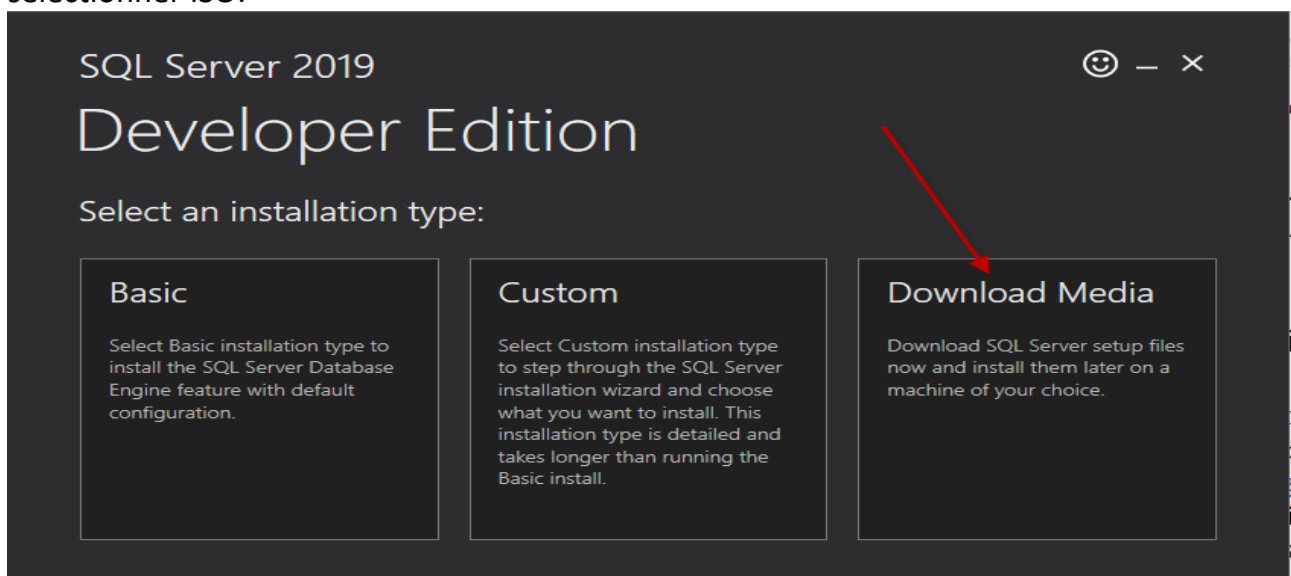
3.1 Installation de SQL Server 2019 et de SSMS

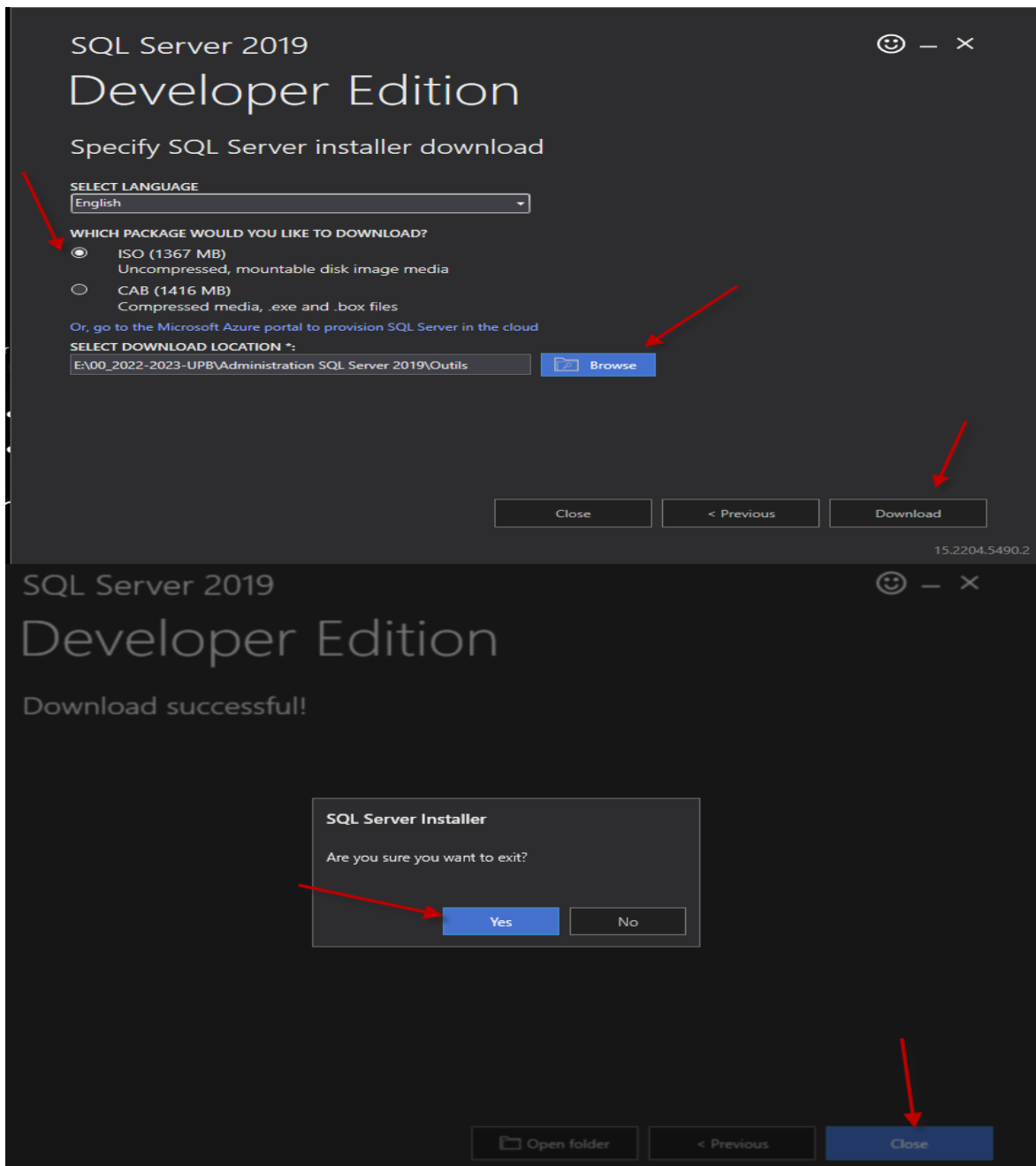
Télécharger le fichier **SQL2019-SSEI-Dev.exe** pour lancer l'installation :

<https://go.microsoft.com/fwlink/p/?linkid=866662>

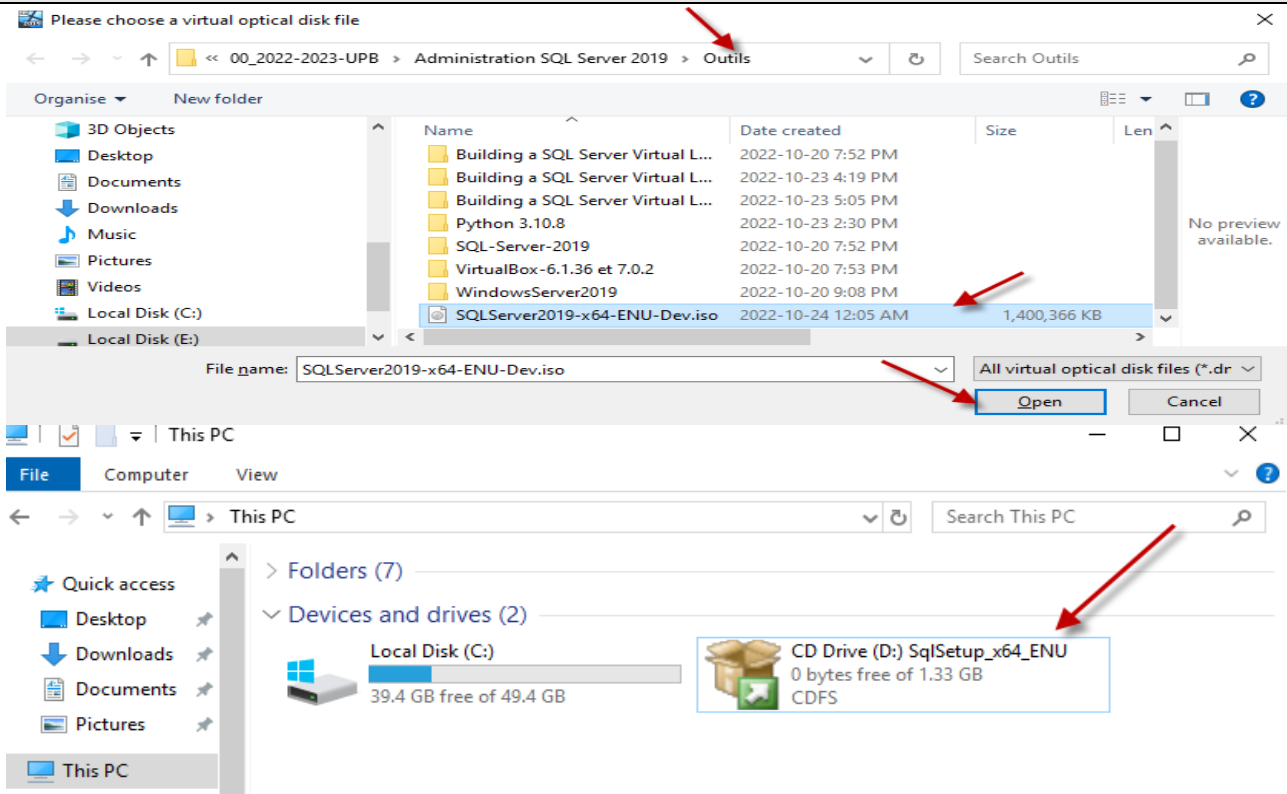
Sauvegarder ce fichier dans votre dossier **Outils** créé au préalable.

Cliquez 2 fois dessus pour commencer l'installation mais choisir Download Media et puis sélectionner ISO.





Dans le menu Device > Optical Drive, choisir Choose a disk file et aller choisir le fichier **SQLServer2019-x64-ENU-Dev.iso** téléchargé dans votre dossier **Outils**.



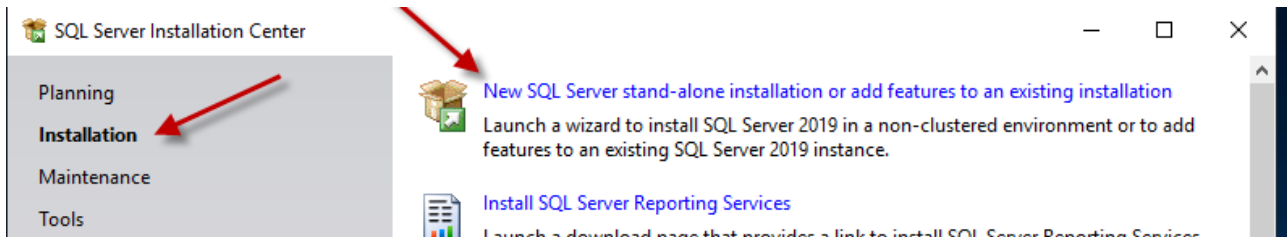
Cliquez 2 fois puis sur **Setup** pour lancer l'installation de SQL Server 2019

Le SQL Server Installation Center s'ouvre :

Cliquez à gauche sur Installation en dessous de Planning puis cliquez sur:

New SQL Server stand-alone installation or add features to an existing installation

Pour commencer installer.



Cliquez sur NEXT et acceptez les termes puis sur NEXT et encore NEXT.

Cliquez encore sur **NEXT** dans **Install Rules** (à gauche). Ne pas tenir compte de l'avertissement Windows Firewall.

Dans la sélection des Fonctionnalités (Feature Selection) cocher les cases suivantes :

- ✓ Database Engine Services
- ✓ SQL Server Replication
- ✓ Client Tools Connectivity
- ✓ Client Tools Backwards Compatibility
- ✓ Client Tools SDK
- ✓ SQL Client Connectivity SDK

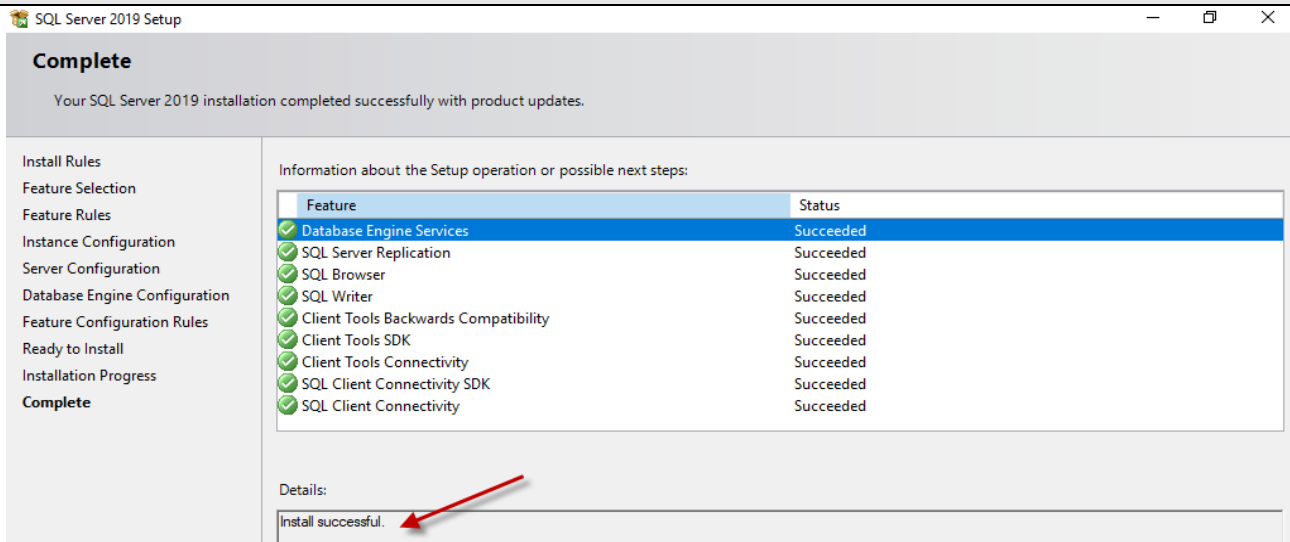
Cliquez sur NEXT puis encore sur NEXT dans **Instance Configuration**
 Ensuite configurez les services SQL Server selon l'écran suivant :

Cliquez sur NEXT pour continuer.

Cliquez sur **Add Current User** pour ajouter un Administrateur SQL (DBA) puis sur NEXT

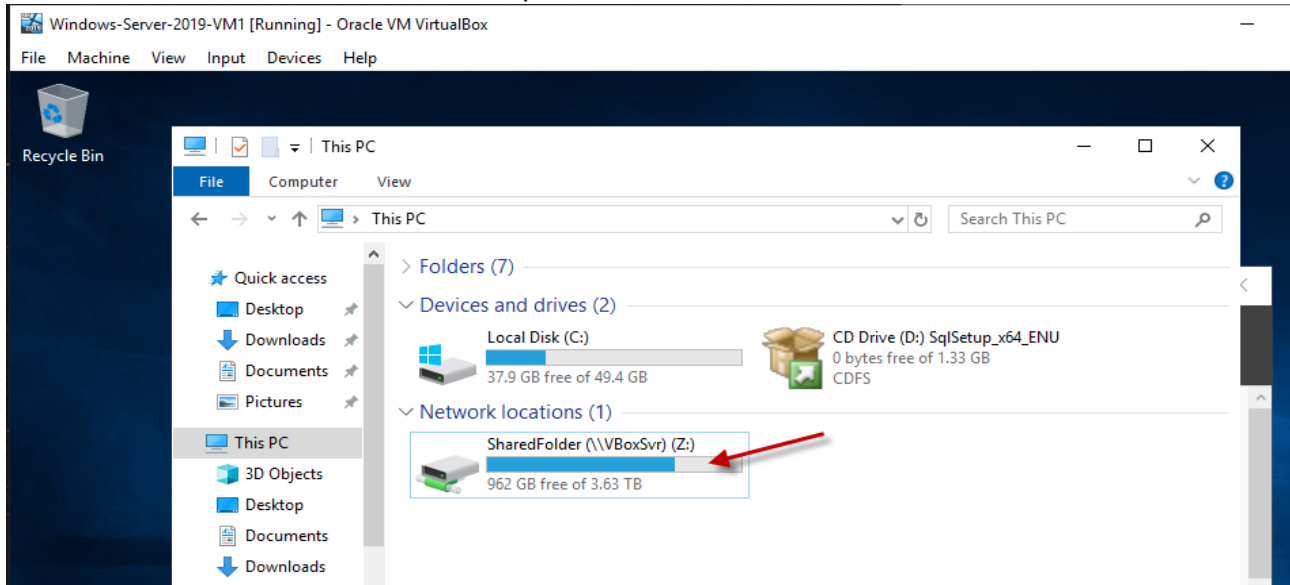
Cliquez maintenant sur **Install**.

Le fichier de configuration se trouve : C:\Program Files\Microsoft SQL Server\150\Setup Bootstrap\Log\20221023_174002\ConfigurationFile.ini

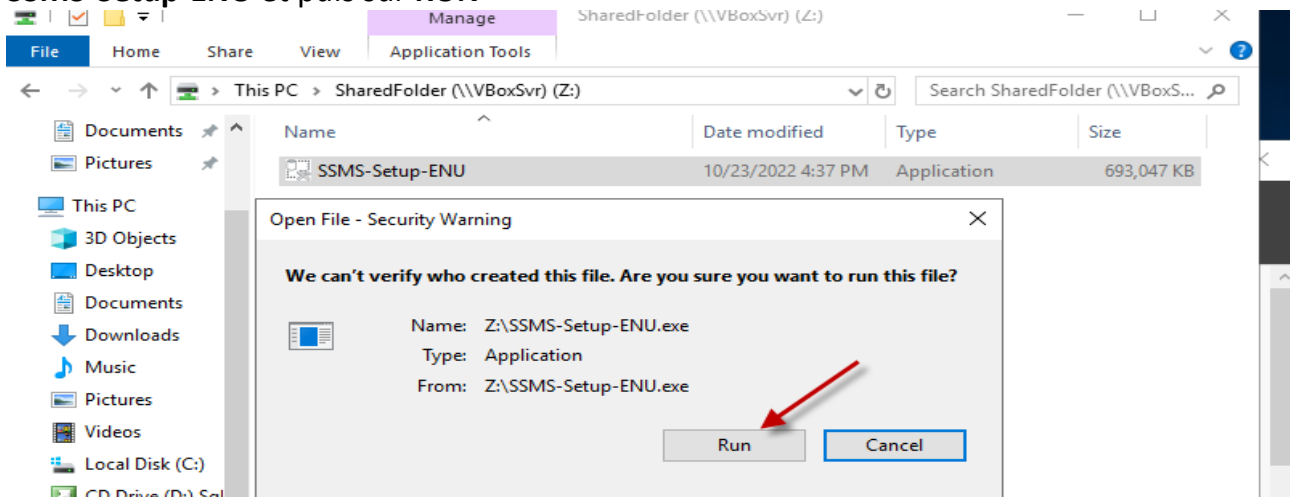


Installez maintenant l'utilitaire **SSMS**.

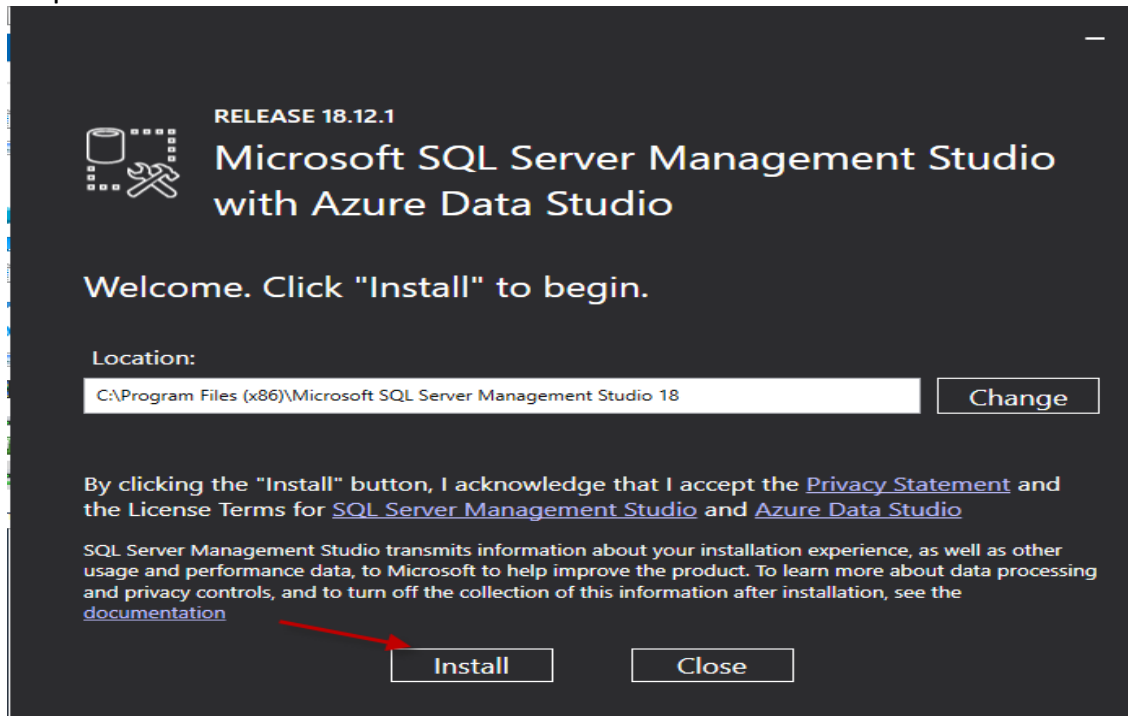
Allez dans le dossier **SharedFolder** à partir de This PC de la machine Virtuelle VM1



Cliquez 2 fois sur **SharedFolder(\\VBoxSvr)(Z:)** puis 2 fois sur le fichier d'installation **SSMS-Setup-ENU** et puis sur **RUN**



Cliquez sur Install



Cliquez à la fin sur Restart

Créer un Raccourci du SSMS sur le Bureau (Desktop) ou sur la barre des tâches.

4. Configuration avancée du réseau et des 2 Serveurs SQL Server 2019

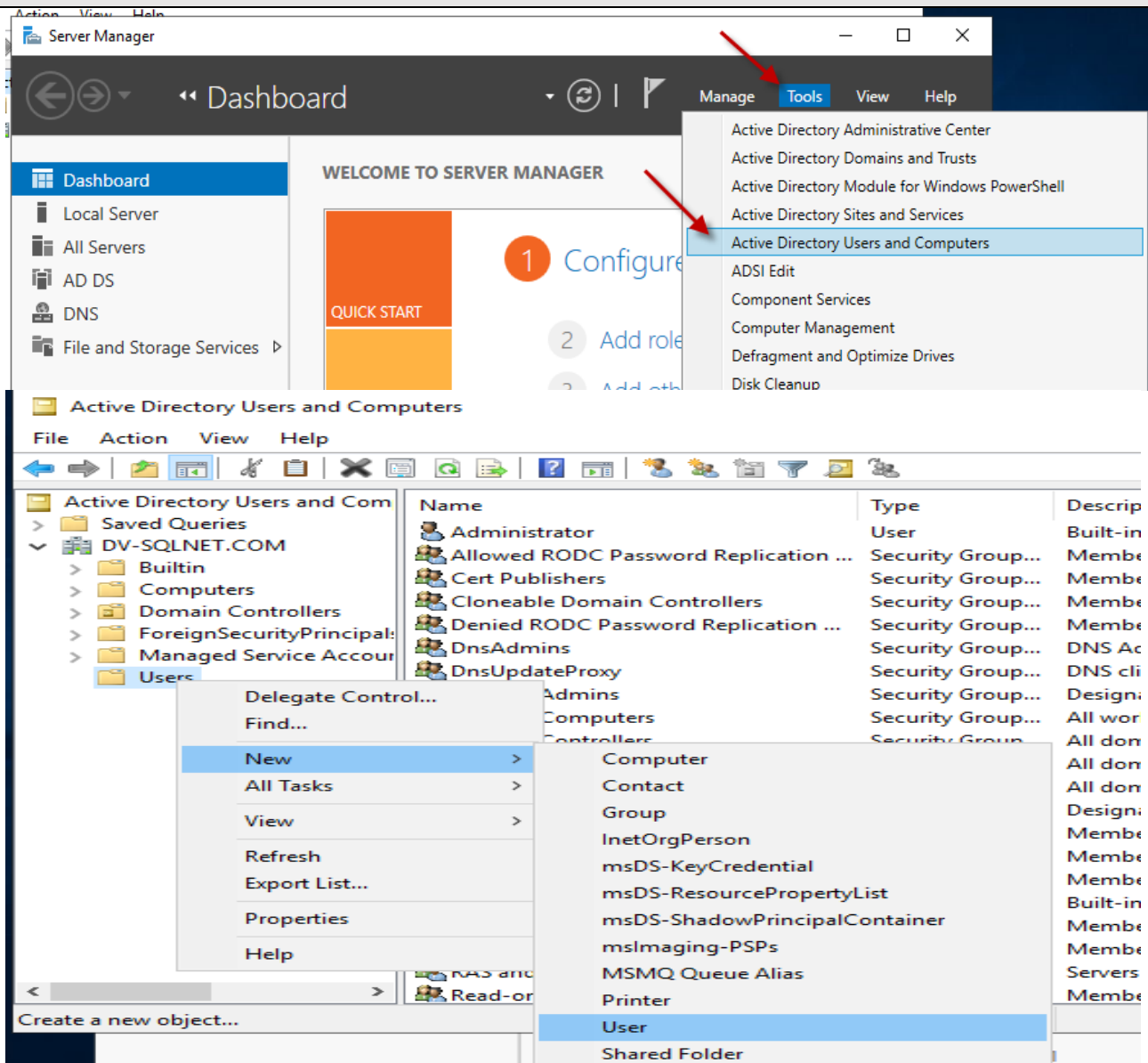
4.1 Création des comptes de Domain pour les services SQL Server

Comme **meilleure pratique de sécurité**, Microsoft recommande de créer des comptes individuels pour tous les services et de suivre le principe du moindre privilège (POLP). En d'autres termes, créez des comptes pour chacun des services requis et accordez les autorisations minimales nécessaires pour que les comptes effectuent leur travail. Pas plus.

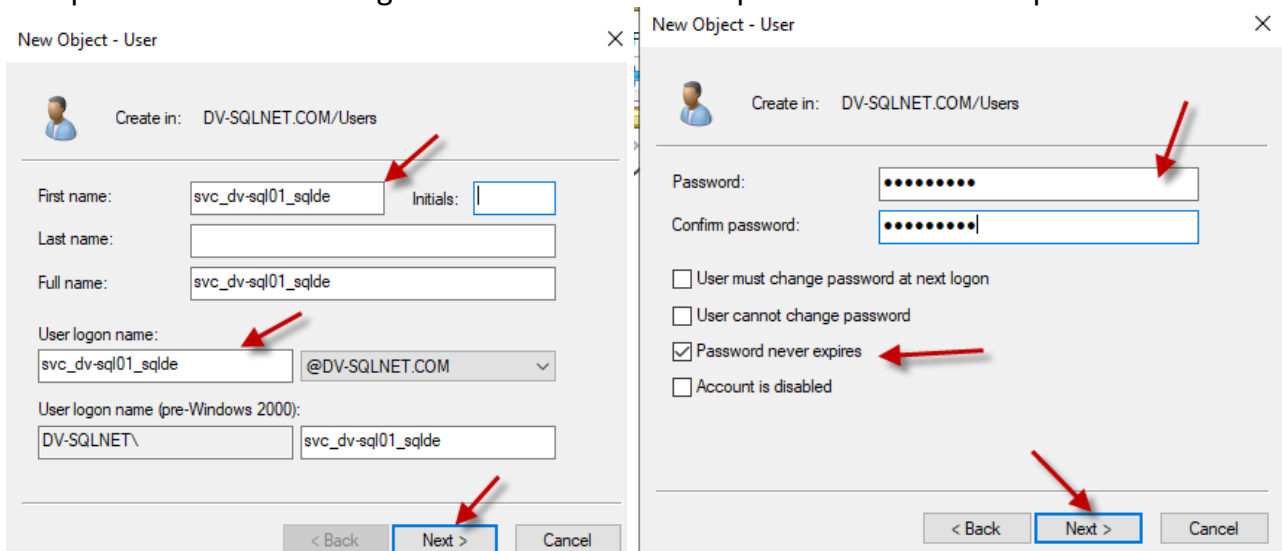
Nous allons créer des comptes de service SQL Server et SQL Server Agent pour chacune des 2 instances SQL Server, définir le mot de passe pour qu'il n'expire pas, créer un groupe de sécurité, puis ajouter tous les comptes de service au groupe de sécurité.

SQL Server	SQL Server Service Account	SQL Server Agent Service Account
DV-SQL01	svc_dv-sql01_sqlde	svc_dv-sql01_sqlagent
DV-SQL02	svc_dv-sql02_sqlde	svc_dv-sql02_sqlagent

Connectez vous sur le Domain Controller : DV-DC



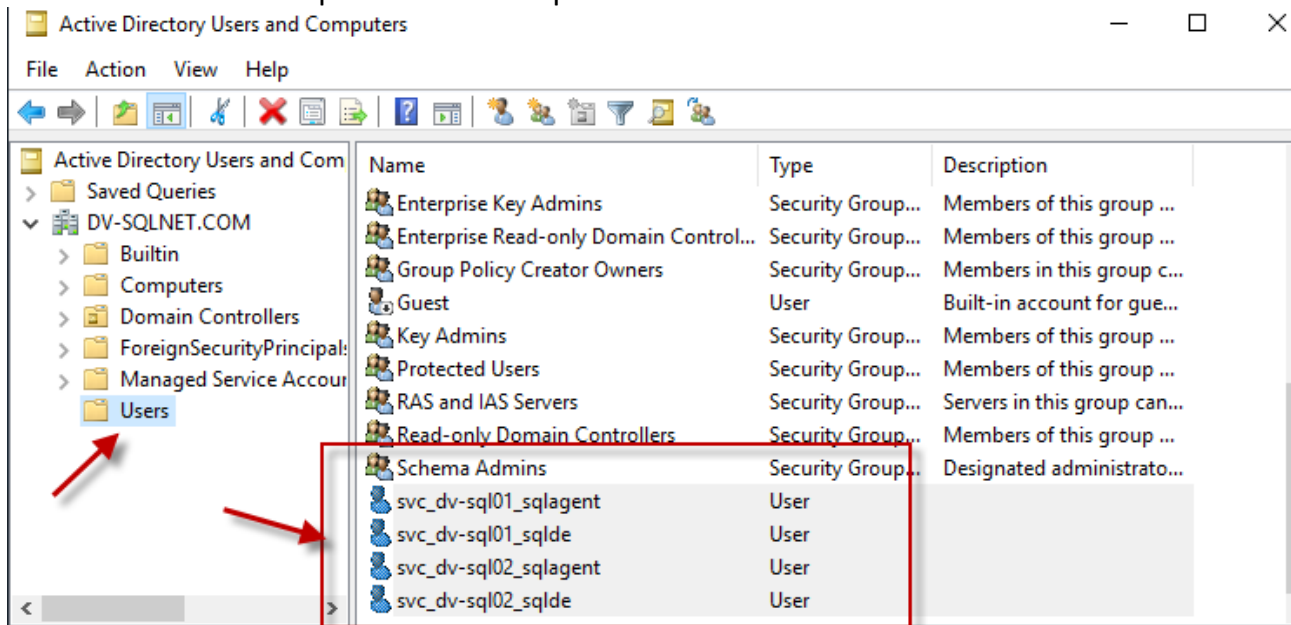
Remplir les détails des usagers comme le montre le prochain écran et cliquer NEXT:



Entrez le mot de passe : Robert123 et cochez **Password Never Expires**

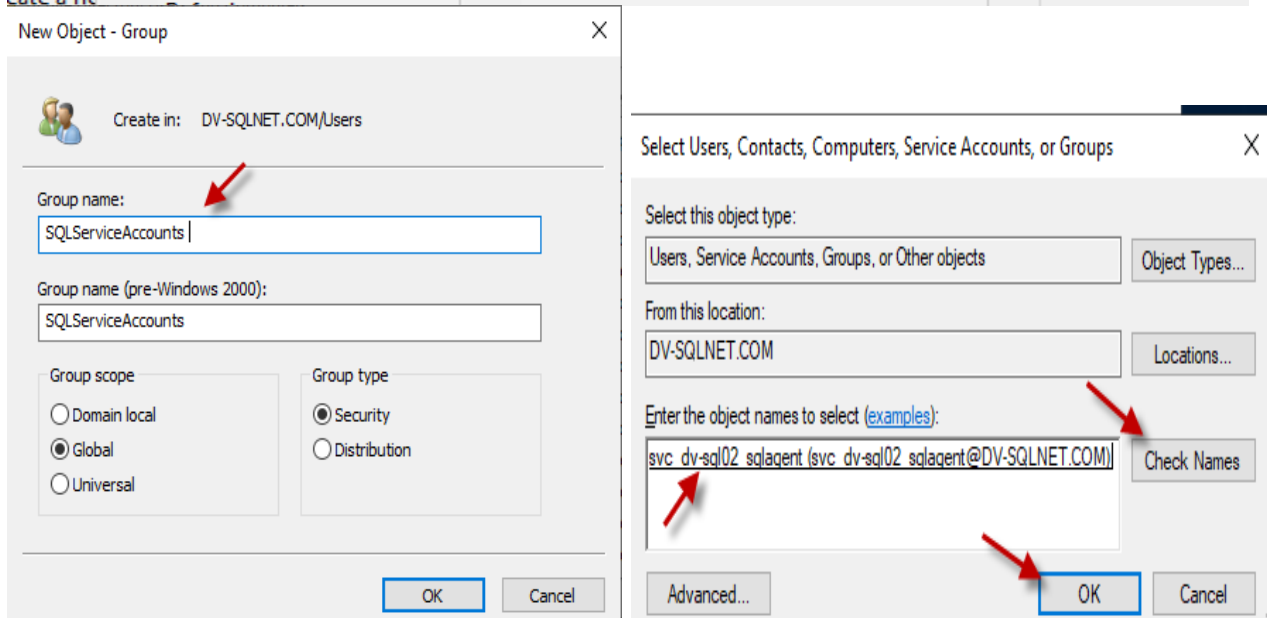
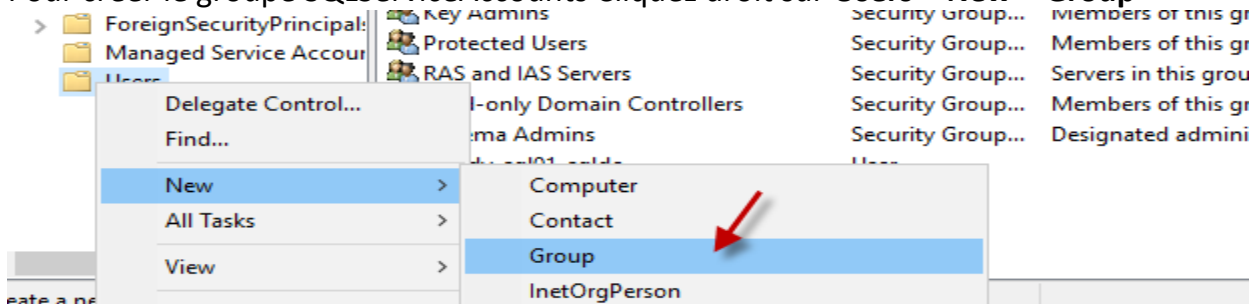
Cliquez sur **NEXT** puis sur **FINISH**

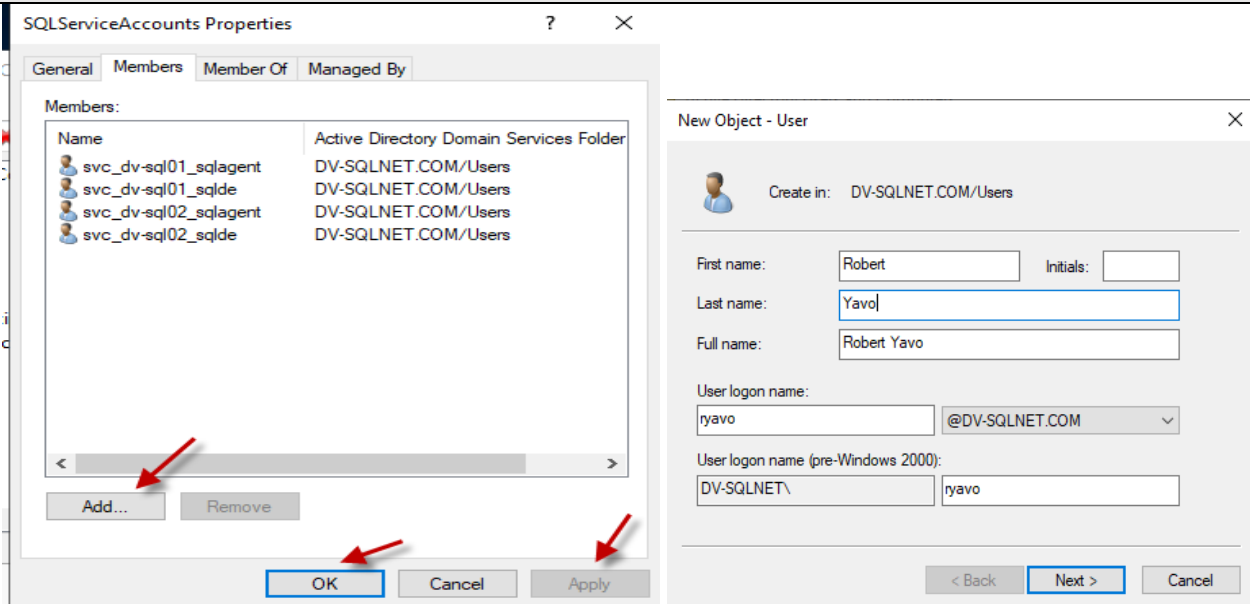
Faites la même chose pour le 2^{ème} compte dans le tableau ci-haut. Vous aurez ceci :



Nous allons maintenant regrouper les utilisateurs de domaine pour les comptes de service, nous allons créer un groupe d'utilisateurs de domaine, **SQLServiceAccounts**, puis ajouter les comptes de service au groupe.

Pour créer le groupe SQLServiceAccounts Cliquez droit sur **Users** > **New** > **Group**





Cliquez sur **OK** lorsque tous les 4 usagers ont été ajoutés.

N'oubliez pas de créer le compte pour le DBA qui est Robert Yavo avec **Password\$** comme mot de passe et username est **ryavo**.

4.2 Configuration du Serveur DV-SQL01 de la Windows-Server-2019-VM1

Nous allons dans cette partie continuer avec le TD-1 pour configurer le Server Windows-Server-2019 de la machine VM1 pour qu'il puisse communiquer avec le Domain Controller DV-DC. Pour réaliser cette tâche nous devrions changer le Nom de l'ordinateur puis, le nom du Domaine puis lui donner une adresse IP statique dans le même réseau.

Ensuite, nous devrions nous devrions lancer l'Utilitaire SSMS pour se connecter au domaine DV-DC avec les noms d'utilisateurs déjà créés dans le tableau de la section 2.2.2

SQL Server	SQL Server Service Account	SQL Server Agent Service Account
DV-SQL01	svc_dv-sql01_sqlde	svc_dv-sql01_sqlagent
DV-SQL02	svc_dv-sql02_sqlde	svc_dv-sql02_sqlagent

Se connecter sur la machine virtuelle VM1 (Windows-Server-2019-VM1) entant qu'Administrateur.

Cliquez sur Local Server puis sur Ethernet pour changer le Computer Name comme on l'a vu dans la configuration du Domain Controller DV-DC.

Donnez les Adresses IP Static, Le Subnet Mask, Le Default Gateway et le DNS Server, comme le montre l'écran qui suit.

Internet Protocol Version 4 (TCP/IPv4) Properties

General

You can get IP settings assigned automatically if your network supports this capability. Otherwise, you need to ask your network administrator for the appropriate IP settings.

☐ Obtain an IP address automatically

☒ Use the following IP address:

IP address: 192 . 168 . 0 . 21

Subnet mask: 255 . 255 . 255 . 0

Default gateway: 192 . 168 . 0 . 1

☐ Obtain DNS server address automatically

☒ Use the following DNS server addresses:

Preferred DNS server: 192 . 168 . 0 . 20

Alternate DNS server: 8 . 8 . 8 . 8

☐ Validate settings upon exit

Advanced...

OK Cancel

Puis cliquez sur OK

Ensuite Ajouter une description : Windows Server 2019 pour DV-SQL01

System Properties

Computer Name Hardware Advanced Remote

Windows uses the following information to identify your computer on the network.

Computer description: Windows Server 2019 pour DV-SQL01

For example: "IIS Production Server" or "Accounting Server".

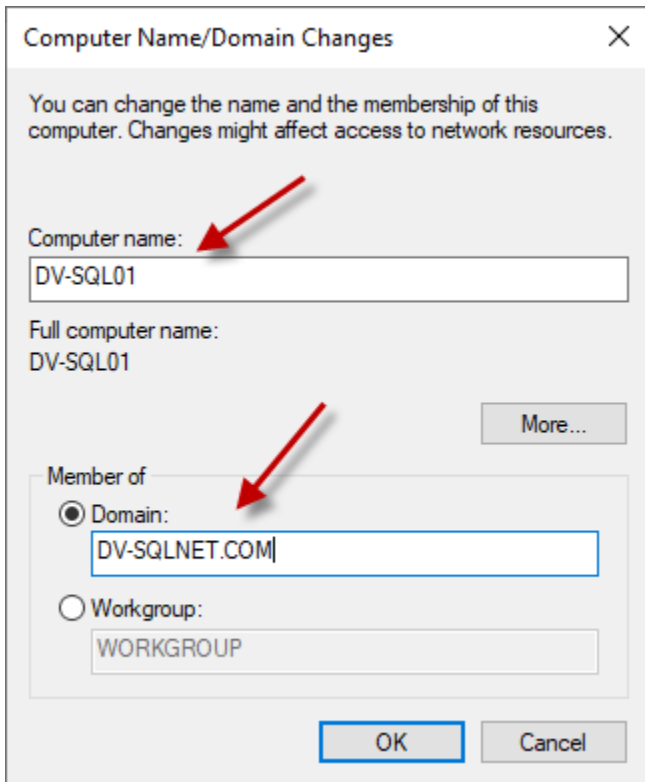
Full computer name: WIN-U7QP1374H4

Workgroup: WORKGROUP

To rename this computer or change its domain or workgroup, click Change.

Change...

Et configurer cette machine pour être membre du Domain DV-SQLNET.COM donné dans le DV-DC plus haut. Modifier aussi le Computer Name à **DV-SQL01** comme le montre l'écran qui suit.



Computer Name/Domain Changes

You can change the name and the membership of this computer. Changes might affect access to network resources.

Computer name: **DV-SQL01**

Full computer name: DV-SQL01

Member of

☒ Domain: **DV-SQLNET.COM**

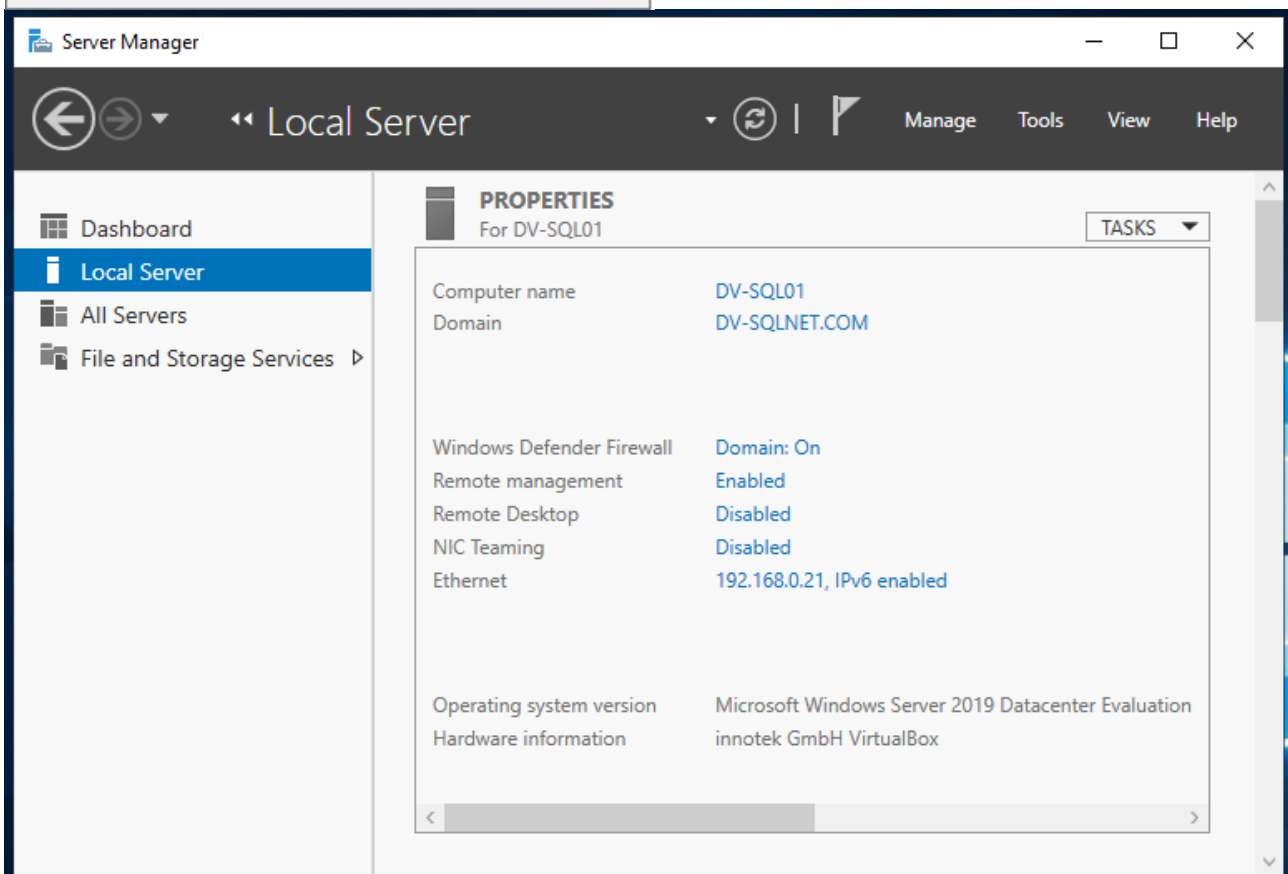
☐ Workgroup: WORKGROUP

OK Cancel

Cliquez sur OK

Cette modification est faite pour que le DV-DC puisse reconnaître cette machine et l'autoriser à se connecter dans le domaine DV-SQLNET.COM.

Redémarrer l'ordinateur virtuelle et reconnecter vous. Vérifiez si la modification a réussie comme dans cet écran qui suit :



Server Manager

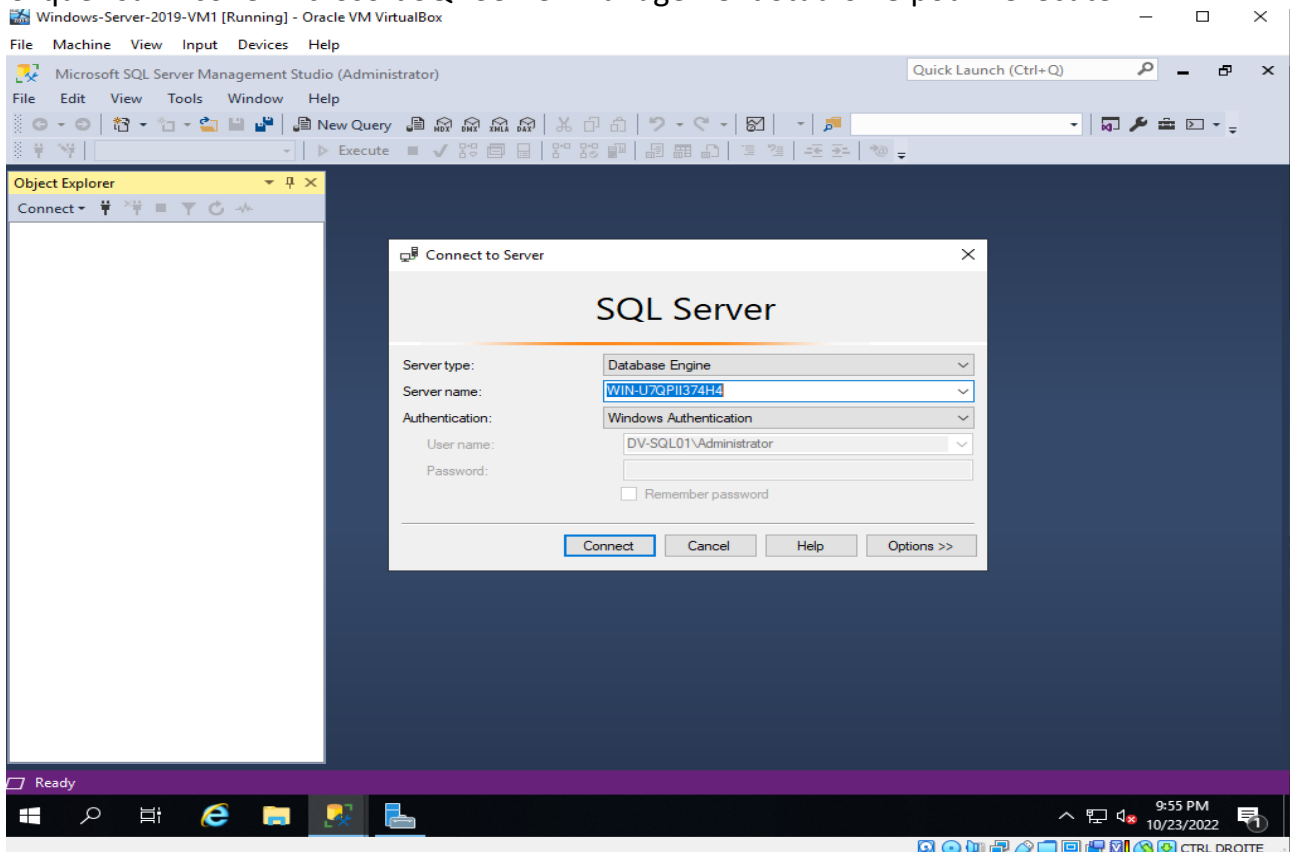
Local Server

PROPERTIES
For DV-SQL01

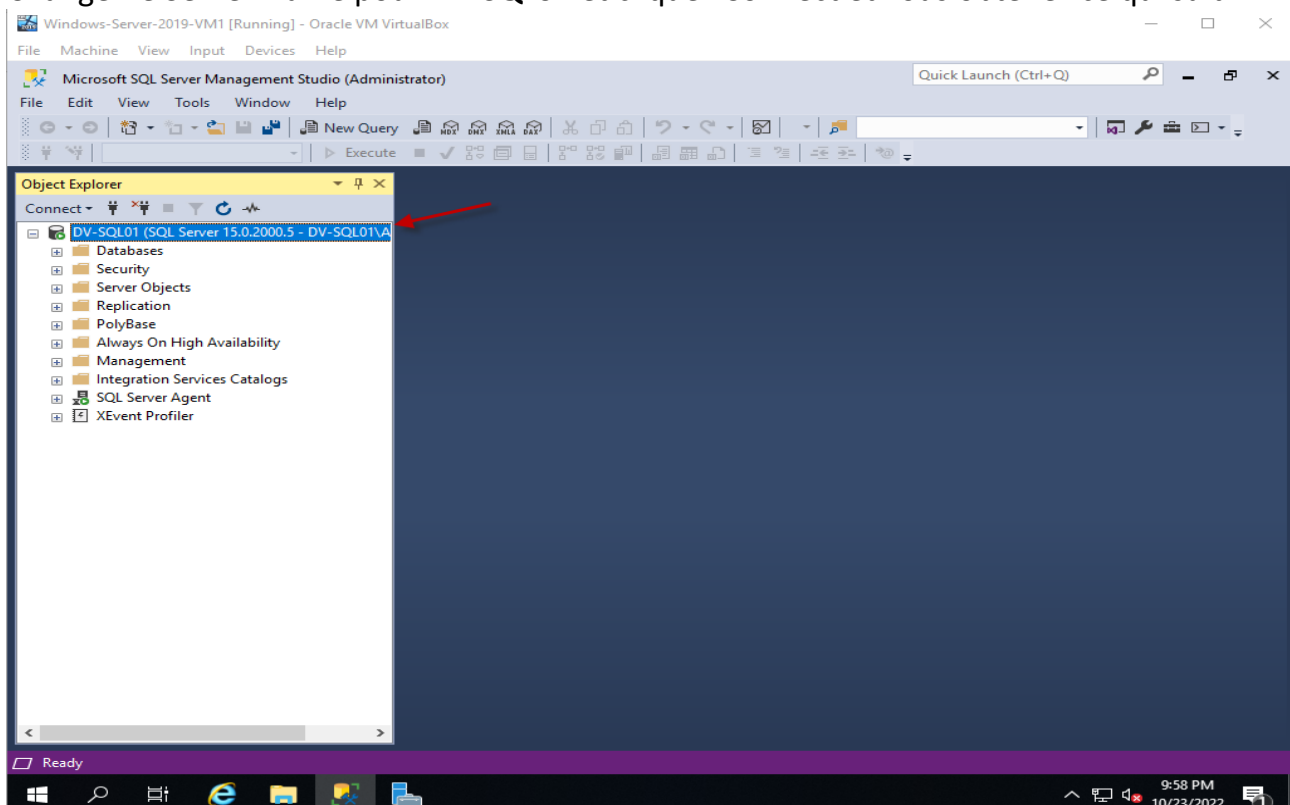
Computer name	DV-SQL01
Domain	DV-SQLNET.COM
Windows Defender Firewall	Domain: On
Remote management	Enabled
Remote Desktop	Disabled
NIC Teaming	Disabled
Ethernet	192.168.0.21, IPv6 enabled
Operating system version	Microsoft Windows Server 2019 Datacenter Evaluation
Hardware information	innotek GmbH VirtualBox

4.3 Connection au Moteur de la Base de données SQL Server (DV-SQL01) avec SSMS

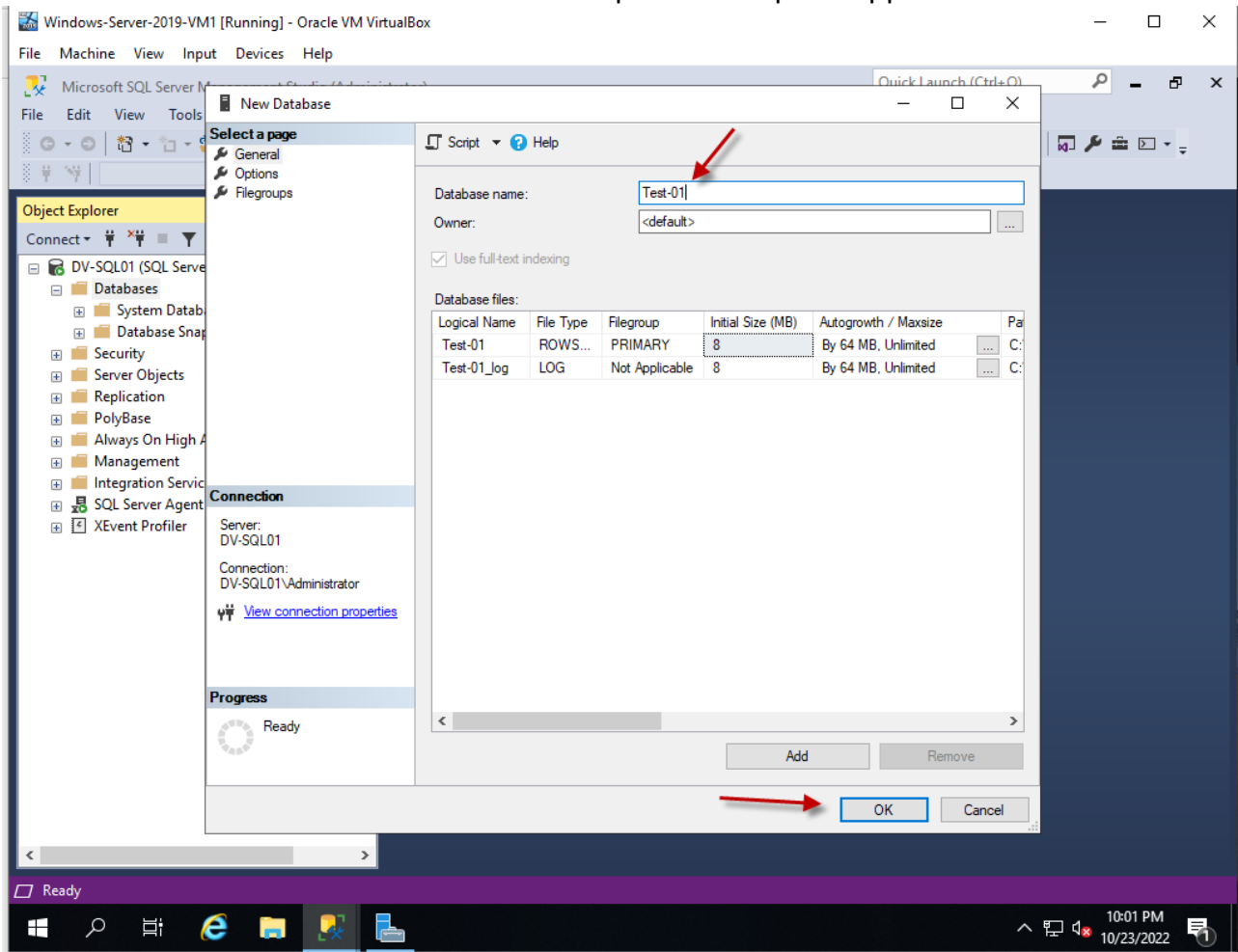
Cliquez sur l'icône Microsoft SQL Server Management Studio 18 pour l'exécuter.



Changer le Server Name pour **DV-SQL01** et cliquez Connect et vous obtenez ce qui suit :



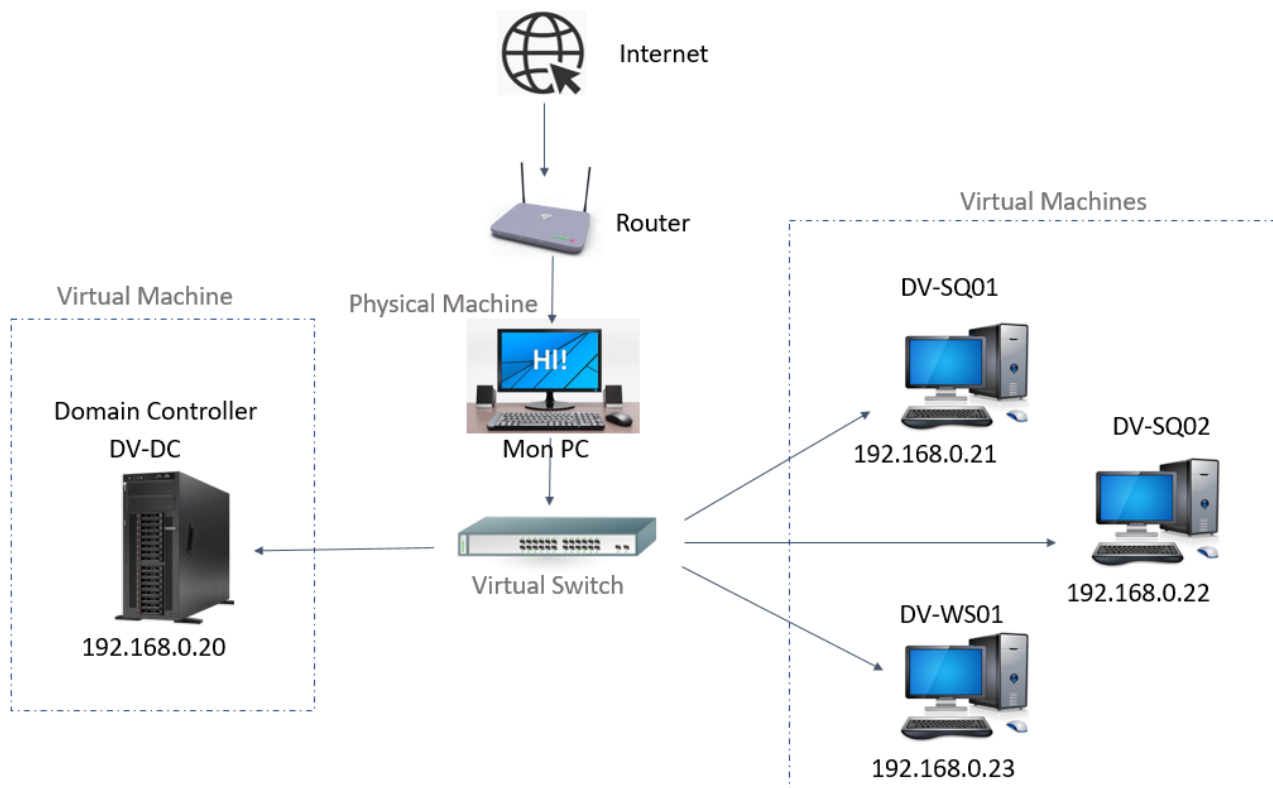
Créer une nouvelle Base de données Test-01 pour tester puis supprimer la.



Fermez l'utilitaire SSMS.

4.4 Configuration Avancée du Réseau Virtuel

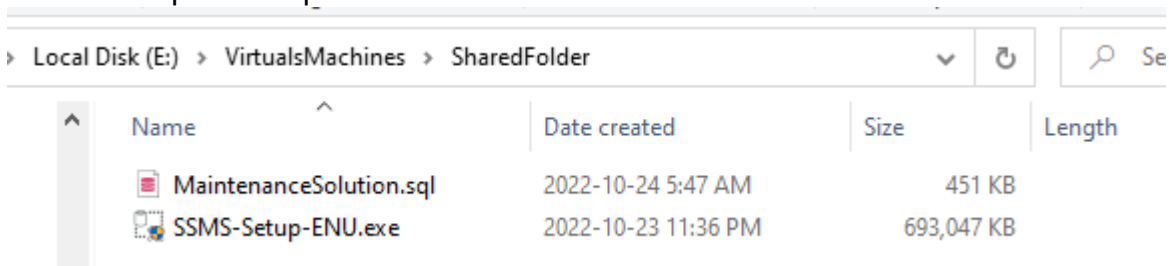
4.4.1 Architecture du Laboratoire du réseau Virtuel



Il s'agit d'installer et de configurer cette architecture de réseaux

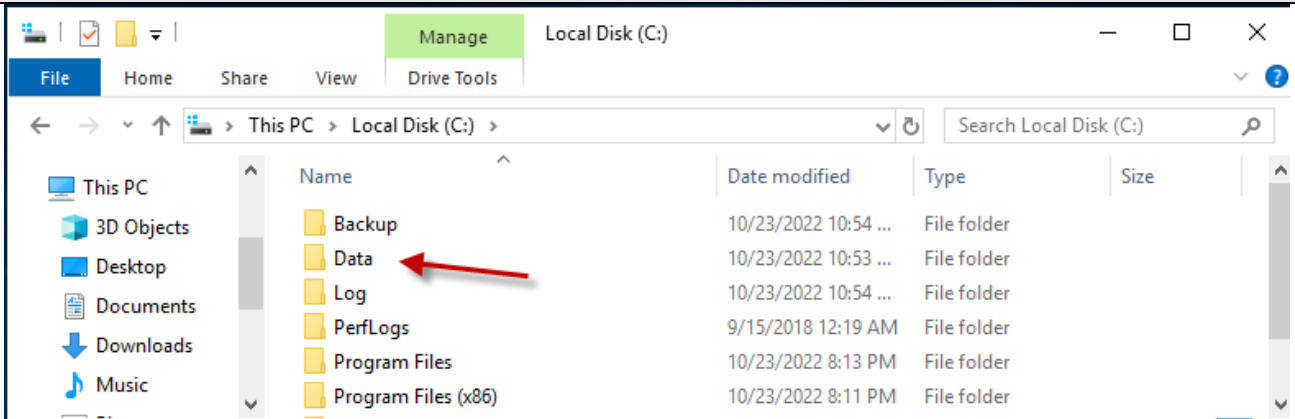
4.4.2 Déployer un utilitaire de SQL Server Maintenance Solution

Télécharger l'utilitaire Ola Hallengren's SQL Server Maintenance Solution à partir du lien suivant : <https://ola.hallengren.com/scripts/MaintenanceSolution.sql> et le sauver dans votre PC et puis le copier-coller dans le dossier **SharedFolder**.

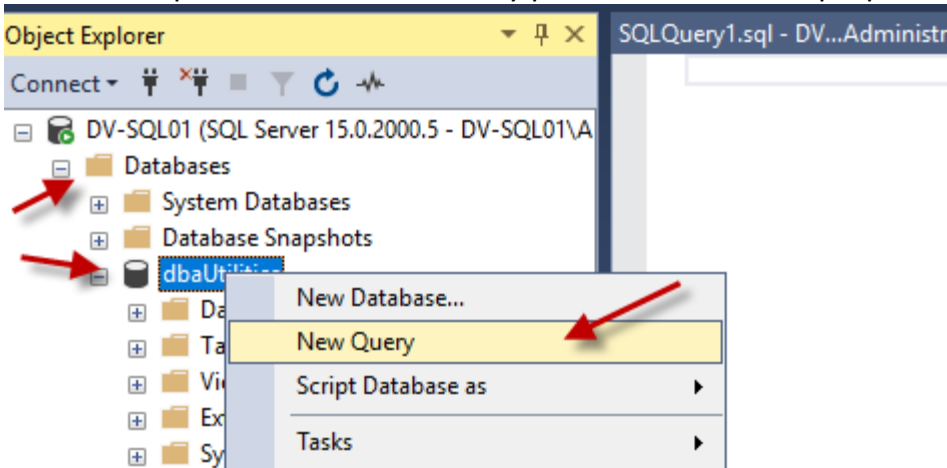


Création des dossiers de données, de journaux et de sauvegarde SQL dans la machine virtuelle

Se connecter sur le DV-SQL01 et cliquer sur This PC > Local Disk (C:) et créer 3 dossiers appelés **Data**, **Log** et **Backup**.



Ouvrir SSMS et Créer une base de données appelée **dbaUtilities** pour le DBA qui contiendra ses scripts. En cliquant sur Databases > New Database et taper le nom dbaUtilities puis créer un New Query pour exécuter le script qui suit l'écran :



```
-- dbaUtilities database
-- Increase the maxsize for data and log files as needed
USE master;
GO
CREATE DATABASE dbaUtilities
ON
( NAME = dbaUtilities_data,
  FILENAME = 'C:\Data\dbaUtilities_data.mdf',
  SIZE = 50MB,
  MAXSIZE = 100MB,
  FILEGROWTH = 10MB )
LOG ON
( NAME = dbaUtilities_log,
  FILENAME = 'C:\Log\dbaUtilities_log.ldf',
  SIZE = 25MB,
  MAXSIZE = 50MB,
  FILEGROWTH = 5MB )
GO
```

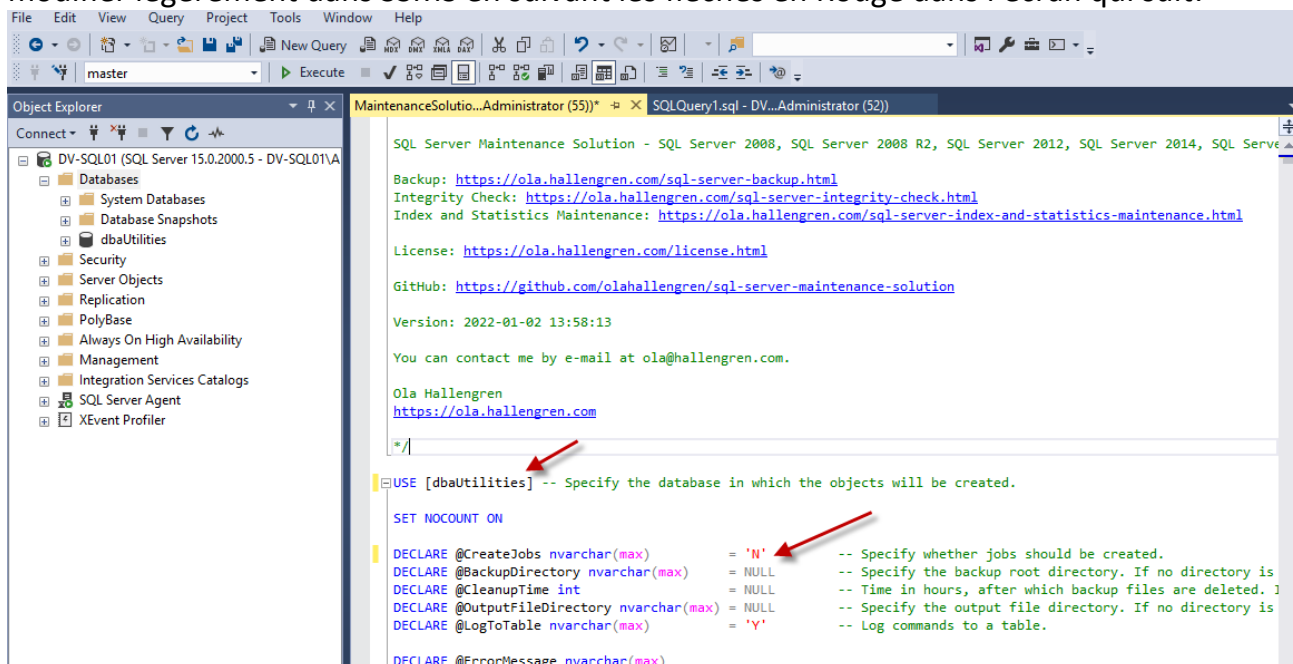

Exécutez la requête en appuyant sur F5 du clavier

Vérifiez que les fichiers dbaUtilities_data et dbaUtilities.log sont créés et contiennent des données.

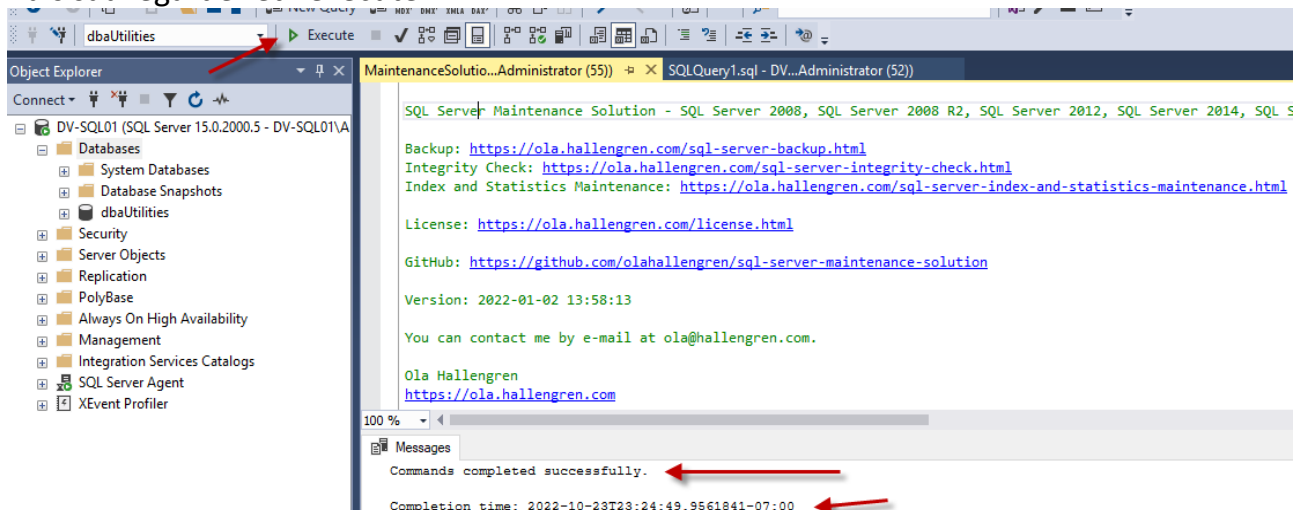
This PC > Local Disk (C:) > Data					Search Data
Name	Date modified	Type	Size		
dbaUtilities_data	10/23/2022 11:13 ...	SQL Server Databa...	51,200 KB		

This PC > Local Disk (C:) > Log					Search Log
Name	Date modified	Type	Size		
dbaUtilities_log	10/23/2022 11:13 ...	SQL Server Databa...	25,600 KB		

Ouvrir le fichier **MaintenanceSolution.sql** qui se trouve dans SharedFolder pour la modifier légèrement dans SSMS en suivant les flèches en Rouge dans l'écran qui suit.



Puis sauvegarder et l'exécuter.



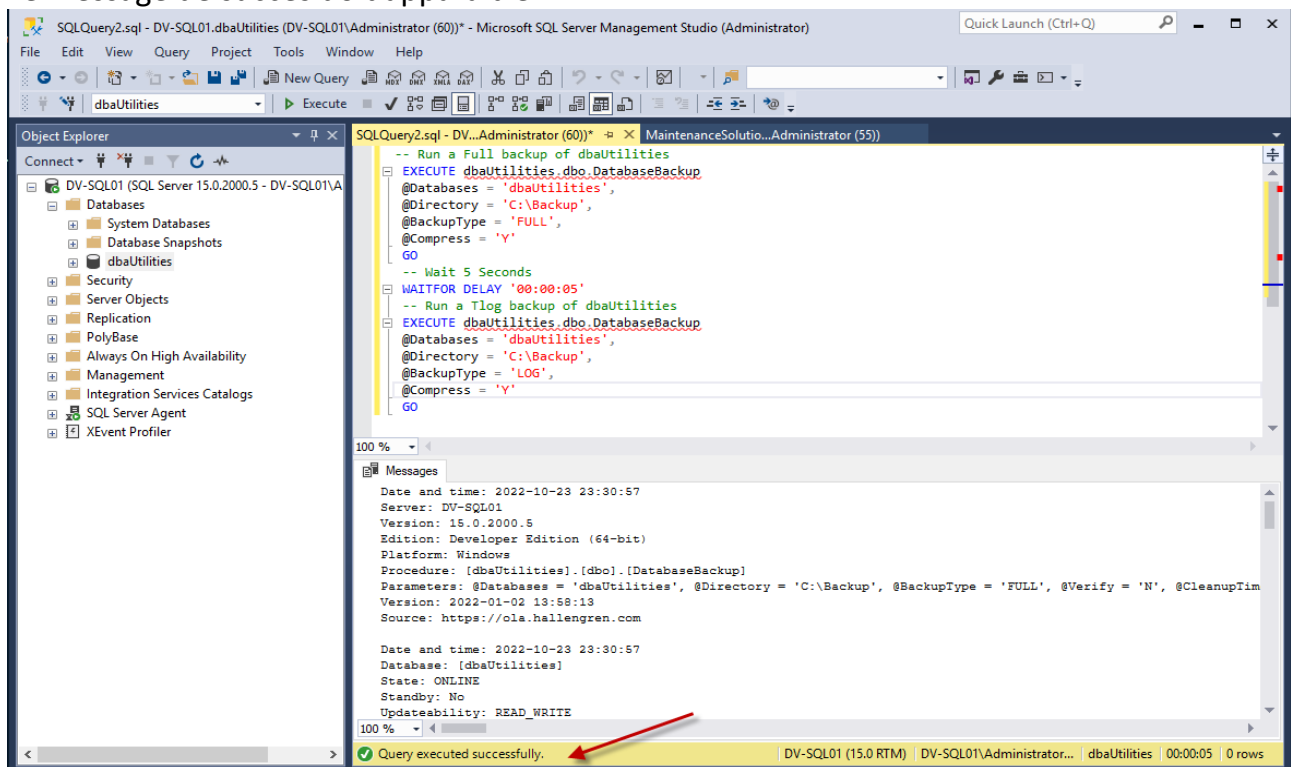
Exécutez le Script suivant pour exécuter les backups Full et Tlog de la BD dbaUtilities.

```
-- Run a Full backup of dbaUtilities
EXECUTE dbaUtilities.dbo.DatabaseBackup
@Databases = 'dbaUtilities',
@Directory = 'C:\Backup',
@BackupType = 'FULL',
@Compress = 'Y'
GO

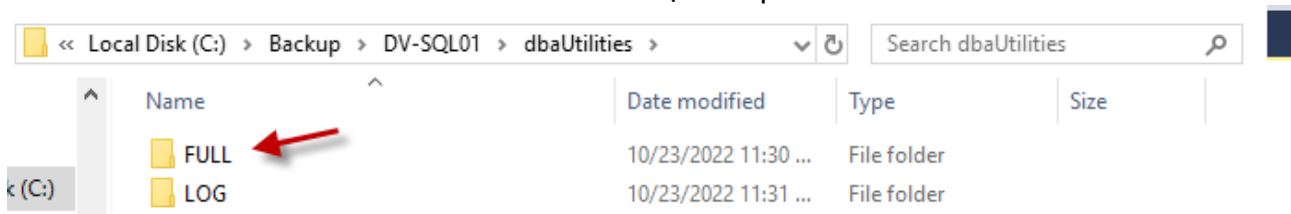
-- Wait 5 Seconds
WAITFOR DELAY '00:00:05'

-- Run a Tlog backup of dbaUtilities
EXECUTE dbaUtilities.dbo.DatabaseBackup
@Databases = 'dbaUtilities',
@Directory = 'C:\Backup',
@BackupType = 'LOG',
@Compress = 'Y'
GO
```

Le message de succès doit apparaître :



Vérifier les dossiers **dbaUtilities** et **Full** dans C:\Backup

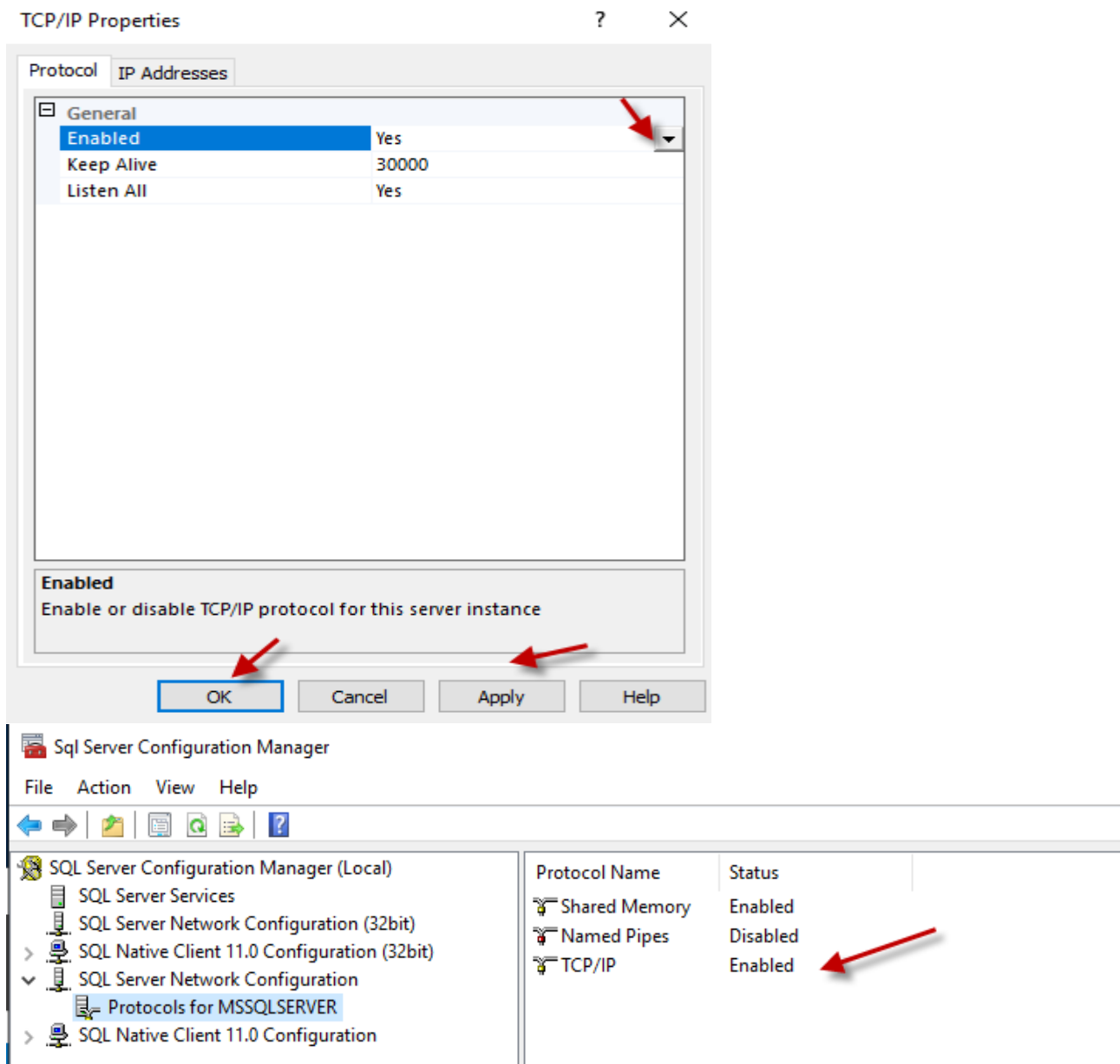


Vous devriez voir ce fichier backup : **DV-SQL01_dbaUtilities_FULL_20221023_232945.bak**

4.4.3 Activer TCP/IP pour SQL Server

Le protocole TCP/IP est requis pour les communications à distance et il est désactivé par défaut dans SQL Server 2019. Vous devez l'activer et redémarrer le service SQL Server pour qu'il prenne effet.

Ouvrir le SQL Server Network Configuration (**Start > Microsoft SQL Server 2019 > SQL Server 2019 Configuration Manager**)



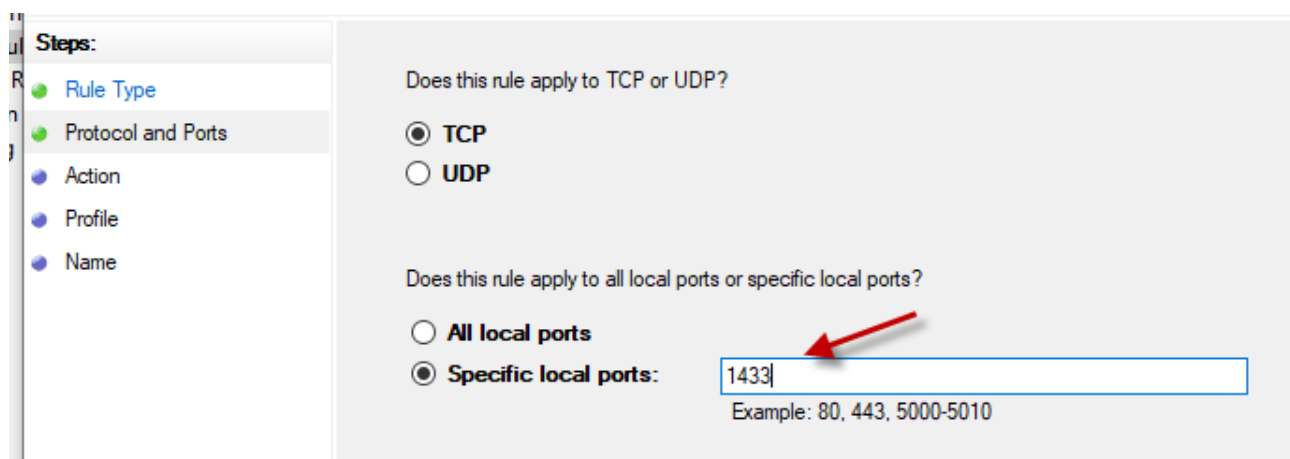
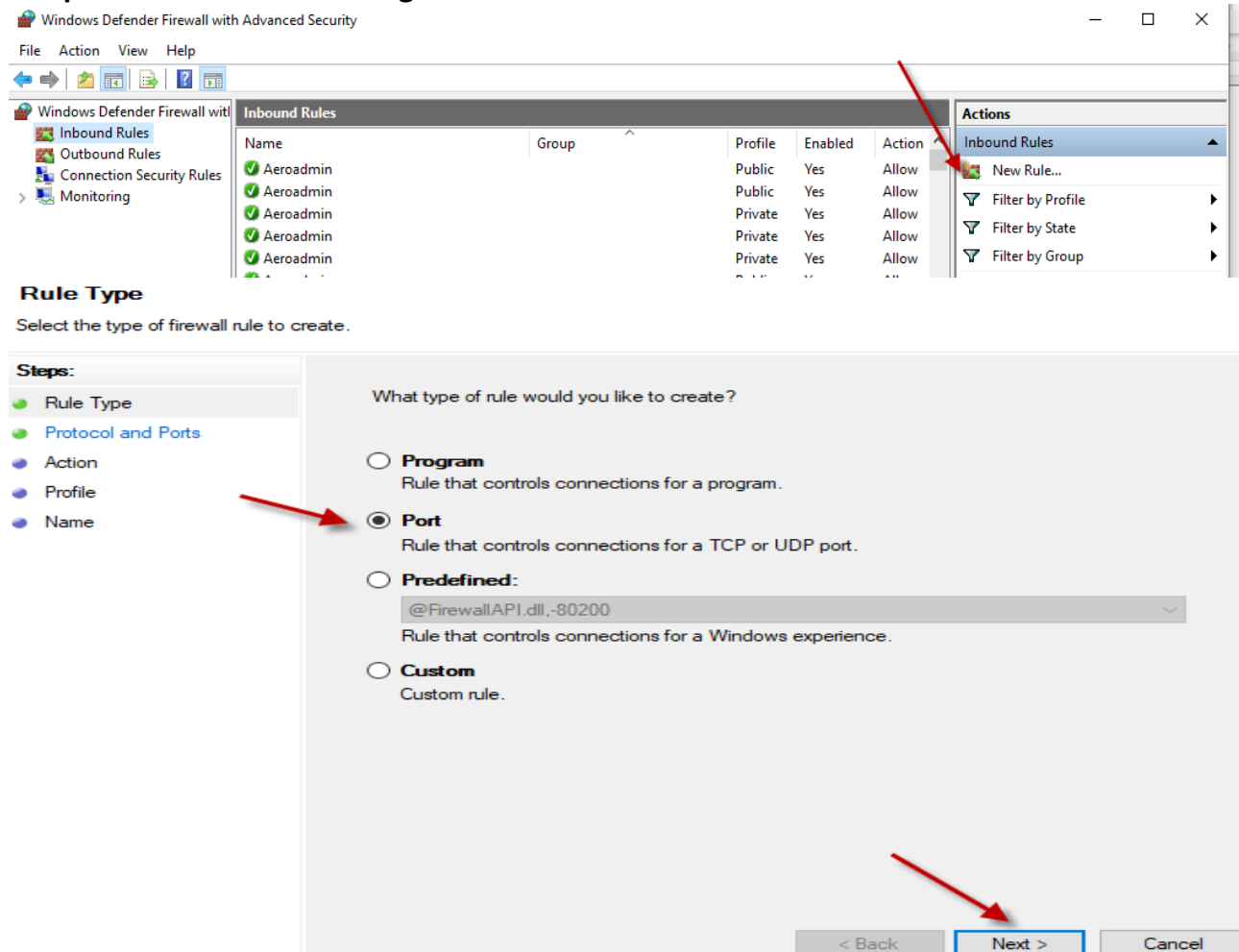
Cela activera TCP/IP pour votre instance SQL Server. Ensuite, nous allons configurer la règle entrante pour le pare-feu Windows sur la machine virtuelle afin d'autoriser les connexions entrantes via le port 1433.

4.4.4 Configurer le Firewall de Windows dans la machine virtuelle

Il faut créer une règle entrante dans le pare-feu Windows pour le port 1433 :

- Ouvrir **Control Panel > System and Security > Windows Defender Firewall**.

- Cliquer sur **Advanced settings > Inbound Rules > New Rule**



en
Rul
d R
on
g

Steps:

- Rule Type
- Protocol and Ports
- Action
- Profile
- Name

What action should be taken when a connection matches the specified conditions?

☒ **Allow the connection**
This includes connections that are protected with IPsec as well as those are not.

☐ **Allow the connection if it is secure**
This includes only connections that have been authenticated by using IPsec. Connections will be secured using the settings in IPsec properties and rules in the Connection Security Rule node.
[Customize...](#)

☐ **Block the connection**

< Back **Next >** Cancel

Rul
d R
on
g

Steps:

- Rule Type
- Protocol and Ports
- Action
- Profile
- Name

When does this rule apply?

☒ **Domain**
Applies when a computer is connected to its corporate domain.

☒ **Private**
Applies when a computer is connected to a private network location, such as a home or work place.

☒ **Public**
Applies when a computer is connected to a public network location.

Name

Specify the name and description of this rule.

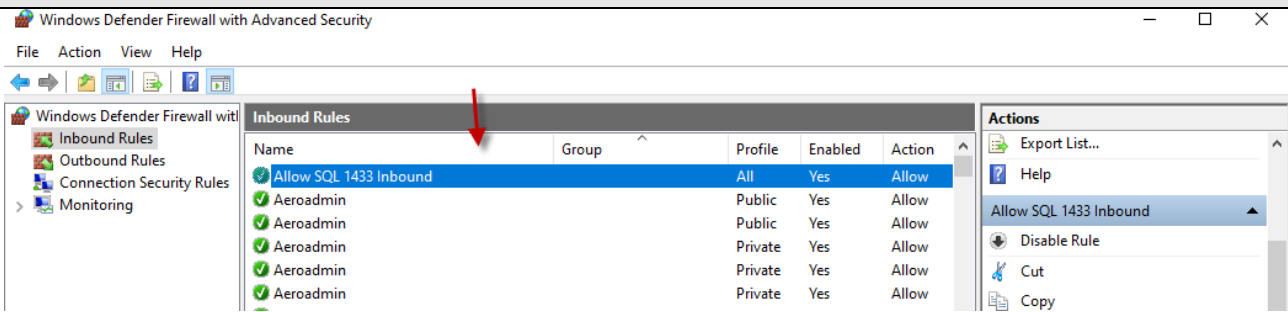
Steps:

- Rule Type
- Protocol and Ports
- Action
- Profile
- Name

Name:

Description (optional):

< Back **Finish** Cancel



Remarque : les instances nommées de SQL Server utilisent des ports dynamiques. Le port dynamique pour SQL Server peut changer à chaque redémarrage du moteur de base de données. Pour les instances nommées, vous pouvez créer une exception dans le pare-feu pour **sqlservr.exe**.

Pour plus de détails: <https://learn.microsoft.com/en-us/sql/sql-server/install/configure-the-windows-firewall-to-allow-sql-server-access?view=sql-server-ver15>

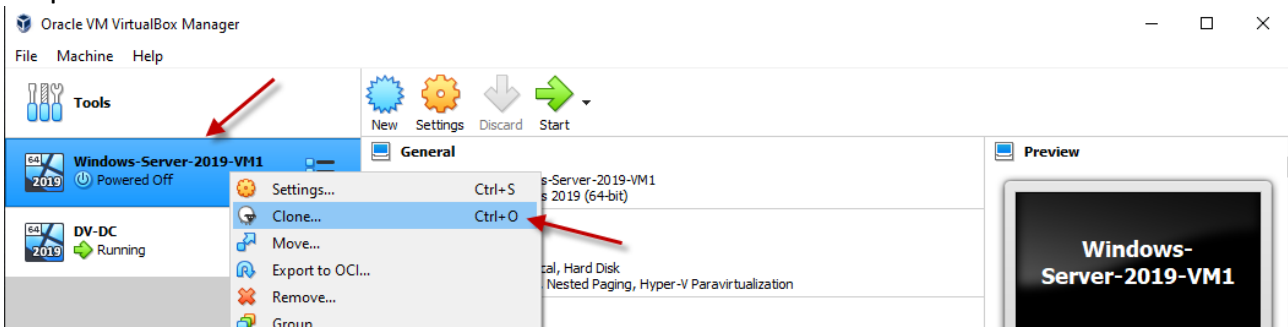
Fermez complètement la machine virtuelle DV-SQL01 de Windows-Server-2019-VM1.

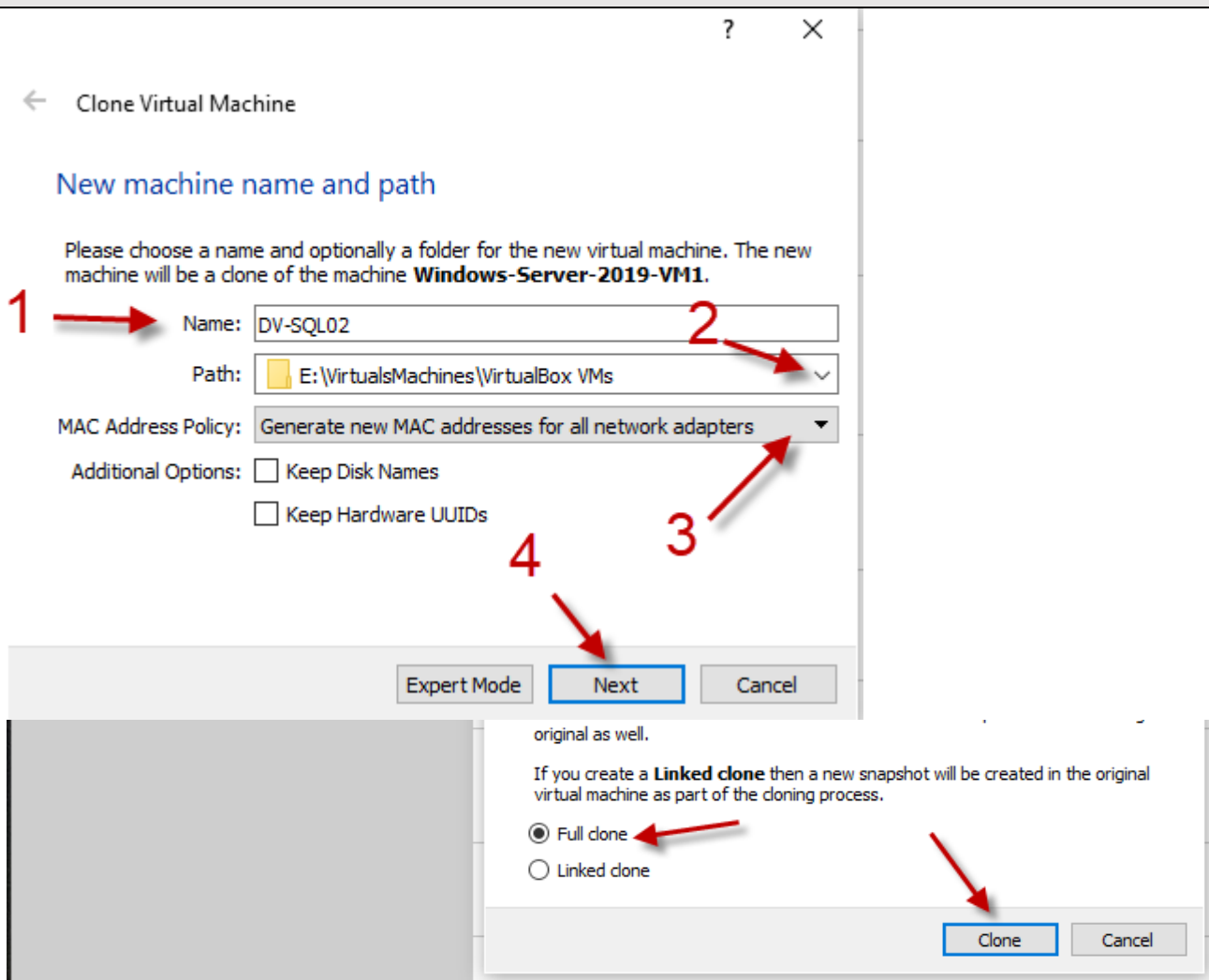
4.5 Configuration d'une nouvelle machine virtuelle et Serveur SQL DV-SQL02

Nous allons utiliser une autre méthode plus simple pour créer la machine virtuelle et ensuite la configurer.

Plutôt que de répéter les étapes de la création d'une machine virtuelle dans Oracle VM VirtualBox à la configuration du pare-feu Windows sur la machine virtuelle pour créer deux machines virtuelles supplémentaires, nous allons **cloner la machine** virtuelle que nous avons créée pour créer la nouvelle machine virtuelle **DV-SQL02**.

Cliquez dans le menu **Machine > Clone** ou utilisez le bouton droit de la souris.





← Clone Virtual Machine

New machine name and path

Please choose a name and optionally a folder for the new virtual machine. The new machine will be a clone of the machine **Windows-Server-2019-VM1**.

1 → Name: DV-SQL02

2 → Path: E:\VirtualsMachines\VirtualBox VMs

MAC Address Policy: Generate new MAC addresses for all network adapters

Additional Options: ☐ Keep Disk Names ☐ Keep Hardware UUIDs

4 → Next

original as well.

If you create a **Linked clone** then a new snapshot will be created in the original virtual machine as part of the cloning process.

☒ Full clone

☐ Linked clone

Clone

5. Tester les connexions et les communications des Serveurs

Suivez les instructions du professeur pour effectuer les tests de communications entre le Domain Controller et les machines virtuelles et les Serveurs de Bases de données.

A2 – Construire et exécuter des requêtes avancées

1. Rappels sur la conception d'un SGBDR

1.1 Définition

Le Système de Gestion de Base de Données Relationnelle (SGBDR) est un type de système de gestion de base de données (SGBD) qui stocke les données dans une structure de table basée sur des lignes et colonnes qui relie les éléments de données associés. Un SGBDR comprend des fonctions qui maintiennent la sécurité, l'exactitude, l'intégrité et la cohérence des données. Ceci est différent du stockage de fichiers utilisé dans un SGBD.

1.2 Types de SGBD

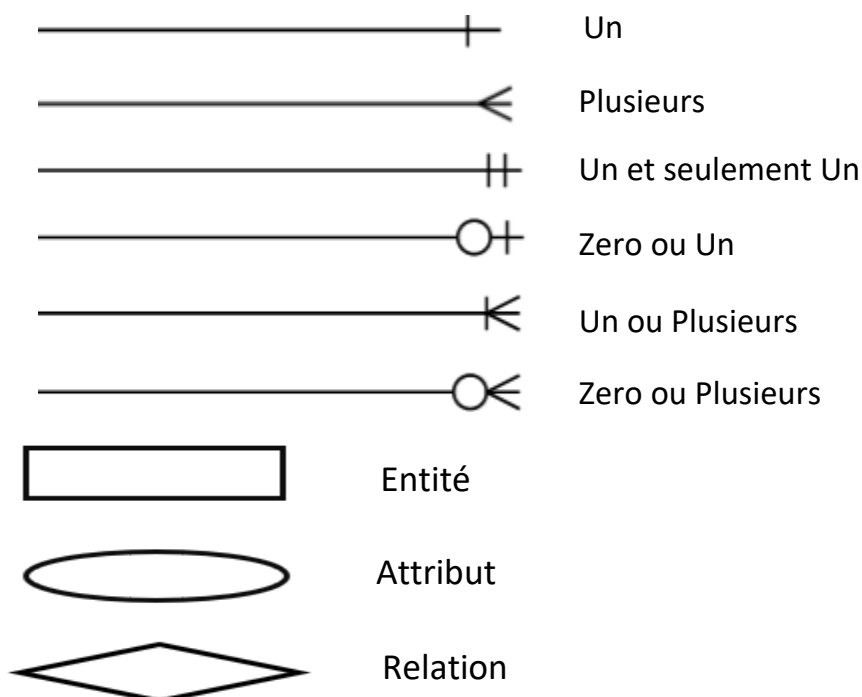
Il existe plusieurs types de systèmes de gestion de bases de données. Voici une liste de sept systèmes de gestion de base de données courants :

- Hierarchical databases
- Network databases
- **Relational databases (Entity-Relational databases) SGBDR**
- Object-oriented databases
- Document databases

Ce qui nous intéresse ici c'est le **Entity-Relational databases** qui est un SGBDR.

1.3 Le Diagramme d'Entité-Relation (ER-Diagram)

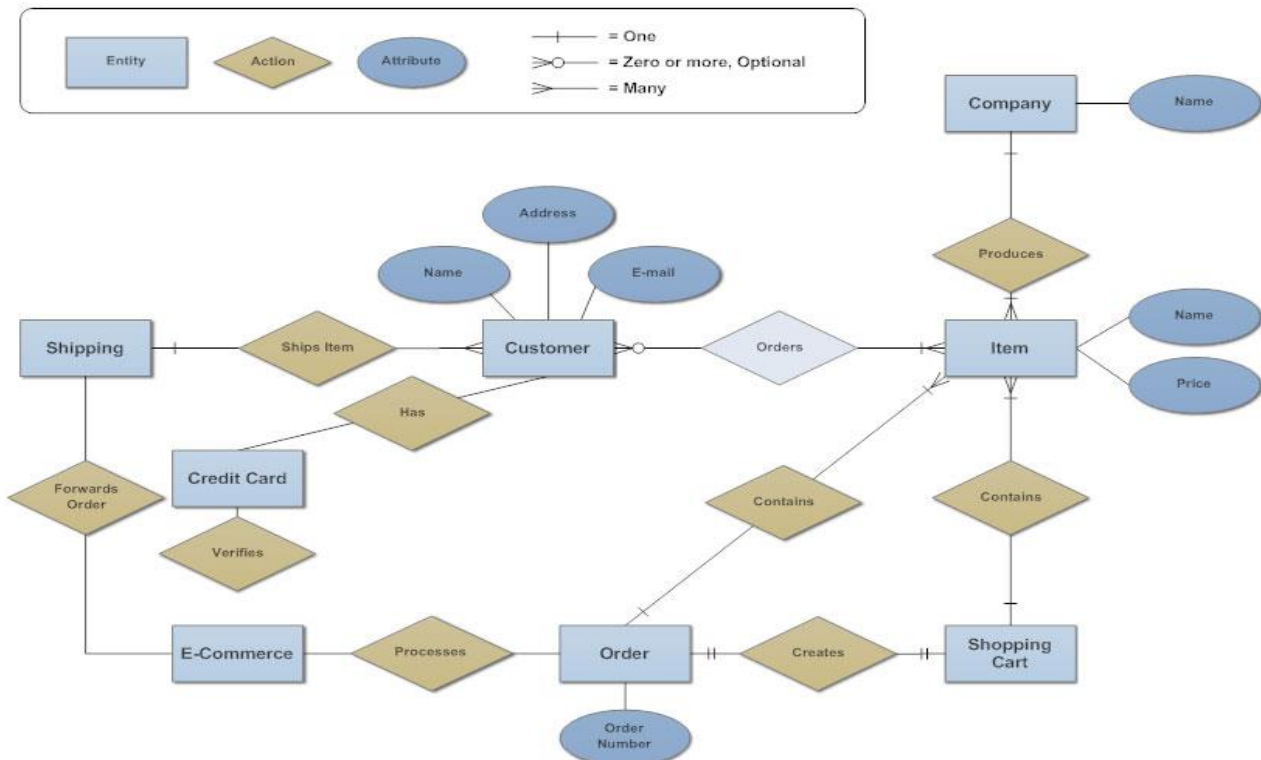
Les représentations graphiques



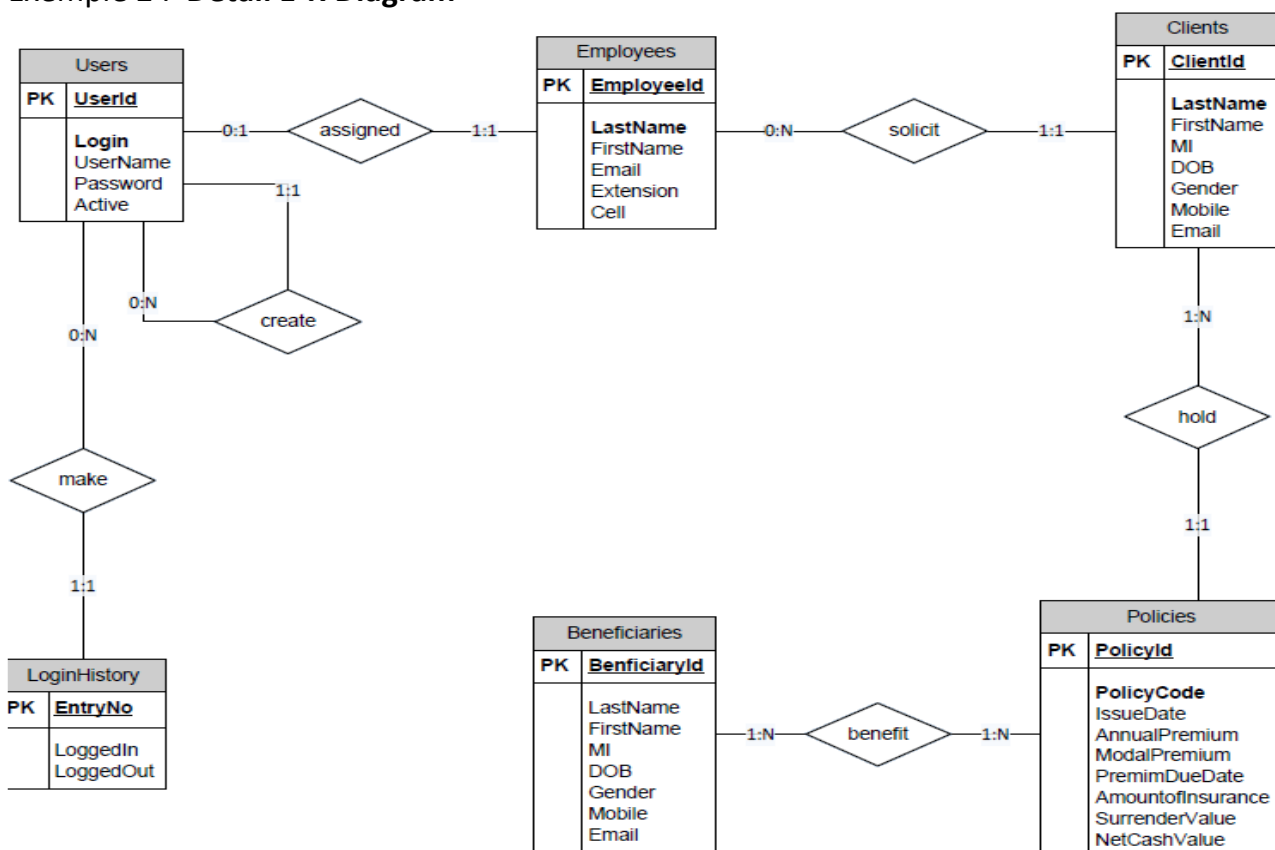
Primary Key (**PK**): Clé primaire et Foreign Key (**FK**): Clé étrangère ou secondaire

Exemple 1 : Simple E-R Diagram

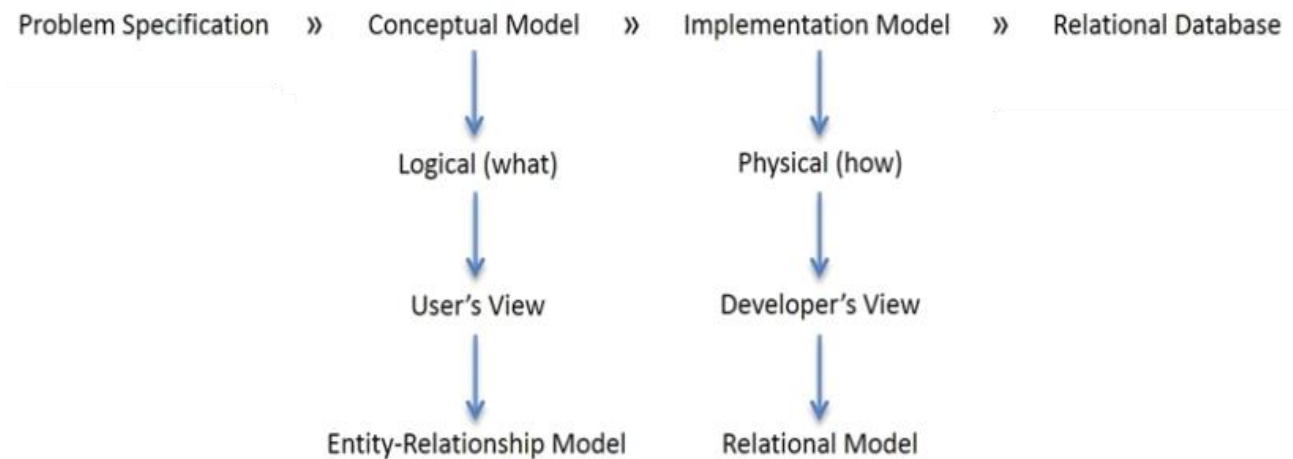
Entity Relationship Diagram - Internet Sales Model



Exemple 2 : Detail E-R Diagram



1.4 Processus de DESIGN d'un SGBDR



De façon générale on part de la Spécification du problème à la Conceptualisation ou Modèle Logique (Ce qui sera représenté) et le Modèle d'Implémentation ou Modèle Physique (Comment ça va se présenter physiquement) dans le Design d'un SGBDR.

L'ensemble du processus de conception d'une base de données relationnelle se fait en 6 étapes.

1. Décrire les entités et les attributs
2. Dédire des relations unaires et binaires
3. Développer (Créer) un Schéma Entité-Relation simplifié
4. Lister les assertions de toutes les relations
5. Créez un diagramme ER détaillé (modèle logique) en utilisant les assertions
6. Transformer le modèle logique en un modèle de relation implémentable (modèle physique)

Étape-01 : Décrire les entités et les attributs

À partir du problème ou des exigences du projet, le développeur de la base de données doit :

- Découvrez les entités
- Donnez des attributs à chaque entité découverte
- Sélectionnez les identifiants, les clés et les clés primaires parmi les attributs de chaque entité découverte

Étape-02 : Découvrir des relations unaires et binaires

Construire la matrice et la remplir. C'est une grille avec le nom des entités pour découvrir les relations unaires ou binaires possibles entre les entités (les tables).

Étape 03 : Créer un diagramme ER simple

A partir de la grille de l'étape 2 on peut concevoir un diagramme ER simple avec les outils de représentations graphiques vus plus haut.

Étape 04 : Répertorier les assertions à l'aide du diagramme E-R simplifié

Lister les optionalités et les cardinalités entre les instances d'entités et leurs relations entre elles. (Zéro à Un, Un à Un, Un à Plusieurs, Plusieurs à Plusieurs)

Étape 05 : Créer le diagramme ER détaillé à l'aide d'assertions

A l'aide des assertions (optionalités et cardinalités) créer le Diagramme ER Détaillé

La convention étant : Optionalité : Cardinalité

Ex : Un client peut faire zéro ou plusieurs paiements => (0:N) entre les tables Clients et Paiements.

Le Diagramme Détaillé est aussi le Modèle Logique (Conception Logique)

Étape 06 Transformez le modèle logique en modèle physique

Cette étape s'est concentrée sur la façon de transformer le diagramme E-R détaillé (modèle logique) créé à l'étape précédente, en un diagramme de modèle relationnel Crow's Foot (modèle physique) qui peut être facilement mis en œuvre sur n'importe quel SGBDR. Il s'agit aussi ici d'arriver au moins à la 3^{ème} forme normale, là où l'intégrité des données est respectée.

Le Modèle Physique est utilisé par le développeur pour l'implémentation physique de ce SGBDR.

Voir en Annexe 1, un autre modèle de meilleures pratiques pour le Design d'une Base de Données Relationnelle en 20 étapes. Mais c'est en version Anglaise.

1.5 Normalisation

La normalisation de base de données est le processus, les étapes ou les techniques que nous utilisons pour structurer une base de données relationnelle sous différentes formes afin d'éliminer les duplications de données (redondance) et d'améliorer son intégrité.

En d'autres termes, il s'agit d'une technique d'organisation des tables de données (entités) en plusieurs tables liées en suivant certains principes et règles afin de minimiser la redondance des données et de maximiser l'intégrité des données.

1.1 Première forme normale (1NF) ?

Pour qu'une table soit dans la première forme normale, elle doit suivre les 4 règles suivantes :

- 1) Il ne doit avoir que des attributs à valeur unique (colonnes à valeur atomique).
- 2) Les valeurs stockées dans une colonne doivent appartenir au même domaine (même genre ou type)
- 3) Toutes les colonnes d'une table doivent avoir des noms uniques.
- 4) Et l'ordre dans lequel les données sont stockées n'a pas d'importance.

1.3 Deuxième forme normale (2NF) ?

Pour qu'une table soit dans la deuxième forme normale,

- 1) Il devrait être dans la première forme normale.
- 2) Et, il ne devrait pas avoir de dépendance partielle.

La dépendance partielle existe lorsque, pour une clé primaire composite, tout attribut de la table dépend uniquement d'une partie de la clé primaire et non de la clé primaire complète.

Pour supprimer la dépendance partielle, nous pouvons diviser la table, supprimer l'attribut qui cause la dépendance partielle et le déplacer vers une autre table

1.4 Troisième forme normale (3NF) ?

Une table est dite en troisième forme normale lorsque :

- 1) Il est sous la deuxième forme normale.
- 2) Et, il n'a pas de dépendance transitive.

1.6 Meilleures pratiques dans le Design d'une Base de données relationnelle

Une bonne conception de base de données est essentielle pour créer des applications évolutives et performantes.

Une base de données est une masse d'informations stockées dans un cadre qui facilite la recherche.

Si une base de données fonctionne bien, des éléments d'information connexes sont automatiquement classés et des détails peuvent être extraits en cas de besoin. Les rapports et les visualisations peuvent être facilement créés.

Suivez ces directives pour éviter les faux pas courants qui peuvent affecter les logiciels à l'avenir.

Avant de définir un seul champ sur un projet, consultez ces 10 meilleures pratiques de conception de base de données qui peuvent considérablement simplifier le processus de développement.

1.6.1. Tenez compte de chaque point de vue lors de la planification

Ne commencez pas à créer une base de données sans la contribution du sponsor du projet et des autres parties prenantes.

Obtenez les besoins et les exigences de ces acteurs.

Obtenez un consensus sur des attentes précises et réfléchissez à la difficulté de former les utilisateurs aux fonctions de recherche.

1.6.2. Choisissez un type de base de données

C'est généralement aussi simple que de choisir entre SQL et NoSQL (bien qu'il existe des types plus spécifiques qui peuvent être appropriés pour certains projets).

Les bases de données SQL sont la norme pour les données structurées, lorsque l'intégrité des données est absolument importante.

Les technologies émergentes telles que l'apprentissage automatique ou l'Internet des objets (IoT) trouvent que la vitesse, l'évolutivité et les exigences fluides des bases de données NoSQL sont mieux adaptées.

L'analyse Web, les réseaux sociaux et certains autres types de bases de données fonctionnent également beaucoup mieux dans le cadre NoSQL. (Prendre la décision le plus tôt possible).

1.6.3. Normalisez vos données

En réalité, la plupart des entreprises fonctionnent aujourd'hui dans un monde hybride de bases de données SQL et NoSQL qui fonctionnent ensemble dans des arrangements complexes.

Avec une structure aussi compliquée, il est essentiel de normaliser les données pour obtenir une redondance minimale.

Éliminez les attributs à valeurs multiples et les attributs répétés, puis commencez par les sous-clés.

1.6.4. Rendre les structures transparentes

La base de données appartient à ses futurs utilisateurs, pas à son créateur, alors concevez en pensant à eux.

Éloignez-vous des raccourcis, des abréviations ou des pluriels. Utilisez des conventions de dénomination cohérentes.

Ne réinventez pas la roue et ne rendez pas les choses difficiles pour ceux qui pourraient avoir besoin de modifier la base de données à un moment donné, ce qui arrivera certainement.

1.6.5. Définissez les contraintes pour maintenir l'intégrité des données

Examinez la gamme complète d'options pour appliquer les règles métier, telles que la clé étrangère, la vérification, la valeur non nulle, etc.

L'application empêchera certaines mauvaises données d'entrer, mais pas toutes.

1.6.6. Documentez tout

Aussi ennuyeux que cela puisse paraître, la documentation est aussi essentielle que les clés primaires.

Prenez soin de documenter la conception, les schémas entité-relation et les déclencheurs pour les futurs utilisateurs.

1.6.7. Prévoyez d'augmenter le temps de sauvegarde

Avant de plonger trop profondément dans le design, pensez à ce qui se passe lors d'une catastrophe naturelle ou causée par l'homme.

Planifiez le clustering de basculement, les sauvegardes automatiques, la réplication et toute autre procédure nécessaire pour garantir que la structure de la base de données reste intacte.

1.6.8. Gardez la confidentialité primaire

Le GDPR signale une ère de préoccupations croissantes en matière de confidentialité.

Chiffrez les mots de passe et n'affectez pas d'administrateur sans formation à la confidentialité et qualifications bien documentées.

Cela peut être une règle délicate à suivre en raison de la politique du bureau, mais en tant que bonne pratique de sécurité, la base de données doit être aussi fermée que possible.

Les vulnérabilités ont un impact sur l'intégrité des données, qui a un impact sur tout le reste de l'entreprise.

1.6.9. Optimisez pour la vitesse

Créez des index pour les requêtes qui seront utilisées régulièrement. Utilisez un analyseur de base de données pour déterminer si un index ou un index clustérisé est nécessaire.

Envisagez d'intégrer des outils efficaces pour accélérer les recherches et la maintenance.

1.6.10. Gardez la base de données sur son propre serveur

Placez la base de données sur un serveur différent du Web pour réduire l'utilisation du processeur.

En plus de libérer des ressources de calcul, cela permet également de garder la base de données hors de portée des utilisateurs non autorisés.

2. Requêtes SQL avancées (LDD, LMD, LCD, LCT)

2.1 Définition SQL

SQL signifie « Structured Query Language » c'est-à-dire « Langage d'interrogation structuré ». C'est un langage déclaratif, il n'est donc pas à proprement parlé un langage de programmation, mais plutôt une interface standard pour accéder aux bases de données. C'est un langage complet de gestion de bases de données relationnelles qui est composé de quatre sous-ensembles :

- **Le Langage de Définition de Données** (LDD, ou en anglais DDL, Data Definition Language) pour créer et supprimer des objets dans la base de données (tables, contraintes d'intégrité, vues, etc.).
Exemple: CREATE, DROP, ALTER
- **Le Langage de Manipulation de Données** (LMD, ou en anglais DML, Data Manipulation Language) pour la recherche, l'insertion, la mise à jour et la suppression de données. Le LMD est basé sur les opérateurs relationnels, auxquels sont ajoutés des fonctions de calcul d'agrégats et des instructions pour réaliser les opérations d'insertion, mise à jour et suppression.
Exemple: INSERT, UPDATE, DELETE, SELECT
- **Le Langage de Contrôle de Données** (LCD, ou en anglais DCL, Data Control Language) pour gérer les droits sur les objets de la base (création des utilisateurs et affectation de leurs droits).
Exemple: GRANT REVOKE
- **Le Langage de Contrôle de Transaction** (LCT, ou en anglais TCL, Transaction Control Language) pour la gestion des transactions (validation ou annulation de modifications de données dans la BD)
Exemple: COMMIT ROLLBACK

2.2 Langage de Définition de Données (LDD)

SQL est un Langage de Définition des Données (LDD), il permet de créer des tables dans une base de données relationnelle, ainsi que d'en modifier ou en supprimer.

Syntaxes:

Pour créer une base de données : **CREATE DATABASE *nom de la BD*;**

Supprimer une Bases de données : **DROP DATABASE *nom de la BD*;**

2.2.1 Les tables

Syntaxe :

```
CREATE TABLE [database_name.][schema_name.]table_name (  
    column_definition1,  
    column_definition2,  
    .....,  
    table_constraints  
);
```

Principaux types de données

- le type CHAR pour les colonnes qui contiennent des chaînes de longueur constante (CHAR(n)).
- le type VARCHAR pour les colonnes qui contiennent des chaînes de longueurs variables (VARCHAR(n)).
- le type SMALLINT Nombres entiers sur 2 octets.
- le type INTEGER Nombres entiers sur 4 octets.
- le type DECIMAL(n,m) correspond à des nombres décimaux qui ont n chiffres significatifs et m chiffres après la virgule.
- le type DATE réserve 2 chiffres pour le mois et le jour et 4 pour l'année.
- le type TIME pour les heures, minutes et secondes.
- ...

Contraintes d'intégrité

- NOT NULL : valeur null impossible.
- UNIQUE : interdit qu'une colonne contienne deux valeurs identiques.
- PRIMARY KEY : définit la clé primaire de la table.
- FOREIGN KEY : indique que la colonne que l'on définit est une clé étrangère qui fait référence à la colonne de la table tableref (contrainte d'intégrité référentielle).
- CHECK : donne une condition que les colonnes de chaque ligne devront vérifier (plage ou liste de valeurs).
- CASCADE : cascader les suppressions ou Modifications.
- SET NULL : rendre nul les attributs référençant.
- RESTRICT : rejet de la mise à jour c'est l'option par défaut.

Les contraintes peuvent s'exprimer :

- Soit au niveau colonne (contraintes locales) : valables pour une colonne.
- Soit au niveau table (contraintes globales) : valables pour un ensemble de colonnes d'une table.

Les contraintes se définissent :

- Soit lors de la création des tables, dans l'ordre CREATE TABLE;
- Soit après la création des tables, par l'ordre ALTER TABLE permettant certaines modifications de la structure des tables.

EXEMPLE 1

```
CREATE TABLE Clients ( idClient CHAR(6) PRIMARY KEY , nom VARCHAR(30) NOT NULL,  
adresse VARCHAR(30) , numéroTelephone INTEGER  
);
```

```
CREATE TABLE Produit ( idProduit CHAR(6) PRIMARY KEY , nom VARCHAR(30) NOT NULL,  
marque VARCHAR(30) NOT NULL , prix DECIMAL(6,2) ,  
- - contrainte de table  
CHECK (marque IN ( BMW, TOYOTA, PEUGEOT) )  
);
```

```
CREATE TABLE Vente ( idVente CHAR(6) PRIMARY KEY , referenceProduit CHAR(6) , idClient  
CHAR(6) ,  
date DATE NOT NULL ,  
- - contrainte de table  
FOREIGN KEY (referenceProduit) REFERENCES Produit(idProduit) ON DELETE  
CASCADE,  
FOREIGN KEY (idClient) REFERENCES Clients (idClient) ON DELETE CASCADE, );
```

Il est possible de renommer, supprimer ou de modifier la structure d'une table à l'aide des commandes : DROP TABLE et ALTER TABLE

Renommer une table

```
ALTER TABLE nom_table RENAME TO nouveau_nom;
```

Ajout ou modification de colonne

```
ALTER TABLE nom_table {ADD/MODIFY} ([nom_colonne type [contrainte], ...]);
```

Renommer une colonne

```
ALTER TABLE nom_table RENAME COLUMN ancien_nom TO nouveau_nom;
```

Supprimer une colonne

```
ALTER TABLE nom_table DROP COLUMN nom_colonne;
```

Ajout d'une contrainte de table

```
ALTER TABLE nom_table ADD [CONSTRAINT nom_contrainte ]contrainte;
```

Supprimer des contraintes

```
ALTER TABLE nom_table DROP CONSTRAINT nomContrainte;
```

Suppression de données uniquement

```
TRUNCATE TABLE nom_table;
```

EXEMPLE 2

```
CREATE TABLE TestDB.dbo.Student(  
    Id INT IDENTITY PRIMARY KEY,  
    Name VARCHAR(65) NOT NULL,  
    Gender VARCHAR(20),  
    Age INT,  
    Marks INT  
)
```

Note: Pour la création de tables avec SSMS, suivre les indications du professeur.

2.2.2 Les Indexes

Afin d'améliorer les temps de réponses, SQL propose un mécanisme d'index. Un index est associé à une table, mais il est stocké à part. Un index peut ne porter que sur une colonne (index simple) ou sur plusieurs (index multiple). Chaque valeur d'index peut ne désigner qu'une et une seule ligne, on parlera alors d'index unique donc de clef primaire, dans le cas contraire on parlera d'index dupliqué.

Un index est automatiquement créé lorsqu'une table est créée avec la contrainte PRIMARY KEY.

Syntaxes:

```
CREATE INDEX index_nom ON table;
```

index sur une seule colonne :

```
CREATE INDEX index_nom ON table (colonne1);
```

index sur plusieurs colonnes :

```
CREATE INDEX index_nom ON table (colonne1, colonne2);
```

Suppression

```
DROP INDEX nom_index;
```

Désactivation

```
ALTER INDEX index_name  
ON table_name  
DISABLE;
```

Réactivation ou Reconstruction

```
ALTER INDEX index_name  
ON table_name  
REBUILD;
```

On peut aussi utiliser DBCC DBREINDEX (table_name, index_name); pour reconstruire un index.

Exécutez cette commande pour voir le résultat.

```
SELECT * FROM USER_INDEXES;
```

2.3 Langage de Manipulation de Données (LMD)

SELECT, INSERT, UPDATE et DELETE sont les 4 commandes de manipulation des données en SQL.

2.3.1 Interroger une base de données

La commande SELECT permet de rechercher des données à partir de plusieurs tables; le résultat est présenté sous forme d'une table réponse.

Syntaxe de base

```
SELECT [ALL | DISTINCT] NomColonne1,... | * FROM NomTable1,... WHERE Condition
```

- ALL : toutes les lignes.
- DISTINCT : pas de doublons.
- * : toutes les colonnes.
- La clause AS : SELECT Compteur AS Ctp FROM Vehicule

Expression des restrictions

Les conditions peuvent faire appel aux opérateurs suivants :

- Opérateurs logiques : AND, OR, NOT, XOR
- Compareurs de chaînes : IN, BETWEEN, LIKE
- Opérateurs arithmétiques : +, -, /, %
- Compareurs arithmétiques : =, <, >, >=, <=, <> — Fonctions numérique : abs, log, cos, sin, mod, power,...
- Fonctions sur chaîne : length, concat,...

EXEMPLE 1

```
SELECT * FROM produit WHERE (prix>1000) AND (prix <=3000)
```

```
SELECT * FROM produit WHERE prix BETWEEN 1000 AND 3000
```

```
SELECT * FROM produit WHERE Marque IN ('TOYOTA','BMW')
```

Différentes clauses de SELECT :

```
SELECT * FROM table
```

```
WHERE condition
```

```
GROUP BY expression
```

```
HAVING condition
```

```
{ UNION | INTERSECT | EXCEPT }
```

```
ORDER BY expression LIMIT count
```

```
— GROUP BY
```

GROUP BY est utilisé pour grouper plusieurs résultats sur un groupe de résultat.

EXEMPLE 2

```
SELECT COUNT (*)  
FROM produit  
GROUP BY marque
```

— HAVING

La clause HAVING permet de spécifier une condition de restriction des groupes. Elle sert à éliminer certains groupes, comme WHERE sert à éliminer des lignes.

```
SELECT COUNT (*)  
FROM produit  
GROUP BY marque  
HAVING COUNT (*) > 40
```

— ORDER BY

ORDER BY permet de trier les lignes dans un résultat d'une requête SQL. Il est possible de trier les données sur une ou plusieurs colonnes, par ordre ascendant ou descendant (ASC|DESC).

Exemple

```
SELECT *  
FROM produit  
ORDER BY marque DESC
```

— UNION

UNION permet de concaténer les résultats de deux requêtes ou plus. Pour l'utiliser il est nécessaire que chacune des requêtes à concaténer retournes le même nombre de colonnes, avec les mêmes types de données et dans le même ordre.

Exemple

```
SELECT marque FROM produit  
UNION  
SELECT marque FROM Vente
```

— INTERSECT

INTERSECT permet d'obtenir l'intersection des résultats de deux requêtes (récupérer les enregistrements communs à 2 requêtes). Cela permet de trouver s'il y a des données similaires sur 2 tables distinctes.

Exemple

```
SELECT marque FROM produit  
INTERSECT  
SELECT marque FROM Vente
```

— Expression des jointures

Le produit cartésien s'exprime simplement en incluant plusieurs tables après la clause FROM. La condition de jointure est exprimée après WHERE

Exemple

```
SELECT P.IdProduit, P.nom  
FROM produit P , vente V  
WHERE P.IdProduit = V.referenceProduit
```

— Requête imbriquée

SQL permet l'imbrication de sous-requêtes au niveau de la clause WHERE. Les sous-requêtes sont utilisées :

-dans des prédicats de comparaison : (=, <>, <, <=, >, >=) - dans des prédicats : IN, EXISTS, ALL ou ANY.

Exemple

```
SELECT C.nom  
FROM client C  
WHERE idClient IN (  
SELECT V.IdClient  
FROM vente V  
WHERE referenceProduit IN (  
SELECT idProduit  
FROM produit P  
WHERE P.nom = 'p1' ))
```

— Le prédicat EXISTS

Il permet de tester si le résultat d'une sous-requête est vide ou non.

Exemple

```
SELECT P.nom  
FROM produit P  
WHERE NOT EXISTS ( SELECT *  
FROM vente V  
WHERE V.referenceProduit = P.IdProduit )
```

— Le prédicat ALL ou ANY

Ils permettent de tester si un prédicat de comparaison est vrai pour tous (ALL) ou au moins un (ANY) des résultats d'une sous-requête.

Exemple

```
SELECT C.nom  
FROM client C, vente V, produit P  
WHERE C.idClient = V. idClient AND P.idProduit = V.referenceProduit AND  
P.prix >= ALL  
( SELECT PR.prix
```



```
FROM client CL, vente VT, produit PR
WHERE CL.idClient = VT. idClient AND PP.idProduit = VT.referenceProduit
AND PR.nom = 'c1' )
```

2.3.2 Modifier une base de données

Les commandes INSERT, UPDATE et DELETE permet la modification d'une base de données.

— La commande INSERT :

La commande INSERT permet d'ajouter de nouvelles lignes à une table

Syntaxe :

```
INSERT INTO NomTable (colonne1,colonne2,colonne3,...)
```

```
VALUES (Valeur1,Valeur2,Valeur3,...);
```

Insertion par une sélection

```
INSERT INTO NomTable (colonne1,colonne2,...)
```

```
SELECT colonne1,colonne2,... FROM NomTable2
```

WHERE condition Exemple :

```
INSERT INTO Clients (idClient, nom, adresse, numéroTelephone ) VALUES
```

```
(c214, hamiche, cité Hamadi N114, 0654874125); Remarque :
```

Les valeurs inconnues prennent la valeur NULL

— La commande UPDATE :

La commande UPDATE permet de changer des valeurs d'attributs de lignes existantes.

Syntaxe

```
UPDATE NomTable SET Colonne = Valeur Ou Expression
```

[WHERE condition] Exemple :

```
UPDATE Clients SET numéroTelephone = 05874521 WHERE idClient = c214; Remarque :
```

L'absence de clause WHERE signifie que les changements doivent être appliqués à toutes les lignes de la table cible.

— La commande DELETE :

La commande DELETE permet d'enlever des lignes dans une table.

Syntaxe

```
DELETE FROM NomTable
```

[WHERE condition] Exemple :

```
DELETE FROM Clients WHERE idClient = 'c214'; Remarque :
```

L'absence de clause WHERE signifie que toutes les lignes de la table cible sont enlevées

2.4 Langage de Contrôle de Transaction (LCT)

Une transaction est une unité logique de traitement qui est soit complètement exécutée, soit complètement abandonnée. Une transaction fait passer la BD d'un état cohérent à un autre état cohérent. Une transaction est terminée : soit par COMMIT, soit par ROLLBACK.

La commande COMMIT

La commande COMMIT termine une transaction avec succès; toutes les mises à jour de la transaction sont validées. On dit que la transaction est validée.

La commande ROLLBACK

La commande ROLLBACK termine une transaction avec échec; toutes les mises à jour de la transaction sont annulées (tout se passe comme si la transaction n'avait jamais existé). On dit que la transaction est annulée.

Exemple :

2.5 Langage de Contrôle de Données (LCD)

2.5.1 Contrôle des droits d'accès (Access Rights)

La commande GRANT permet de passer des droits d'accès à un utilisateur ou un groupe d'utilisateurs

Syntaxe

GRANT privilèges ON table TO bénéficiaire
[WITH GRANT OPTION]

Les privilèges qui peuvent être passés sont :

- soit ALL (tous les privilèges)
- soit une liste de privilèges parmi :
 - SELECT
 - INSERT
 - UPDATE [(liste de colonnes)] : l'omission de la liste de colonnes signifie toutes les colonnes
 - DELETE

Le bénéficiaire peut être :

- soit PUBLIC (tous les utilisateurs)
- soit un utilisateur ou un groupe d'utilisateurs

L'option WITH GRANT OPTION permet de passer un privilège avec le droit de le transmettre.

La commande REVOKE permet de retirer des droits à un utilisateur ou groupe d'utilisateurs.

Syntaxe

REVOKE privilèges ON table FROM bénéficiaire

2.5.2 Les rôles

- Regroupement de privilèges pour des groupes d'utilisateur
- Facilitent la gestion des autorisations des privilèges objet en évitant les ordres GRANT
- Un rôle par défaut est donné à un utilisateur
- Un utilisateur peut posséder plusieurs rôles mais il n'est connecté qu'avec un seul à la fois
- On peut donner un mot de passe pour certains rôles

Création / Modification d'un rôle

```
{CREATE|ALTER} ROLE nom_role {NOT IDENTIFIED | IDENTIFIED {BY mot_de_passe|}  
EXTERNALLY};
```

Remplissage et attribution d'un rôle

```
GRANT {privilege1 | role1} TO nom_role;  
GRANT {privilege2 | role2} TO nom_role;  
GRANT ROLE nom_role TO user;
```

Rôle par défaut ou activation

```
SET ROLE nom_role [IDENTIFIED BY mot_de_passe ];
```

Suppression / Révocation d'un rôle

```
DROP ROLE nom_role; REVOKE ROLE nom_role FROM user;
```

Exemple

```
CREATE ROLE employeur;  
GRANT SELECT, INSERT,UPDATE(adresse, numéroTelephone) ON clients TO employeur;  
GRANT SELECT, INSERT ON produit TO employeur;  
GRANT ALL ON vente TO employeur;  
GRANT ROLE employeur TO Said, Salah, Karim;
```

2.5.3 Les vues (Views)

Nous allons dans un premier temps, voir les notions de bases des vues de façon générale et dans une autre séance nous allons aller en profondeur dans les notions avancées des vues et des rapports comme indiqué dans le Syllabus A5.

Définition

La vue est une requête stockée dans le dictionnaire de données, sur laquelle l'utilisateur peut interroger comme il le fait sur des tables. C'est une table virtuelle qui apparaît à l'utilisateur comme une table réelle mais qui n'utilise pas la mémoire physique. Elle est calculée dynamiquement, chaque fois que l'utilisateur effectue une requête dessus.

Les vues assurent l'indépendance logique, elles peuvent être utilisées pour cacher et/ou protéger des données sensibles, ou pour montrer des données statistiques.

Création et suppression d'une VUE :

Syntaxe :

```
CREATE VIEW NomVue(NomColonne1,...) AS SELECT NomColonne1,..  
FROM NomTable  
WHERE Condition
```

Suppression :

```
DROP VIEW nom_vue;
```

RENOME :

```
RENAME VIEW nom_vue TO nouveau_nom;
```

Intérêt des vues

— Indépendance logique

Le concept de vue permet d'assurer une indépendance des applications vis-à-vis des modifications du schéma (Assurer l'indépendance du schéma externe).

— Simplification d'accès

Les vues simplifient l'accès aux données en permettant par exemple une prédéfinition des jointures et en masquant ainsi à l'utilisateur l'existence de plusieurs tables (Création de résultats intermédiaires pour des requêtes complexes).

Exemple : La vue qui calcule les moyennes générales pourra être consultée par la requête :

```
SELECT * FROM Moyennes
```

— Confidentialité des données

Une vue permet d'éliminer des lignes sensibles et/ou des colonnes sensibles dans une table de base (éviter de divulguer certaines informations).

2.6 Connexions à distance

L'Administrateur de Base de données (DBA) peut manuellement ou automatiquement, à partir d'un ordinateur accéder aux bases de données et à les manipuler à distance.

Voyons quelques outils de connexion à distance.

Lab1 : Créez une nouvelle machine virtuelle contenant Windows 10 Professionnel et connectez-le au réseau du domaine du Laboratoire Virtuel créé plus haut.

Créez un fichier sql qui contient une requête pour lister le contenu de toutes les tables créées dans les exemples et à partir d'un script Batch ou PowerShell exécuter ce fichier sql à distance. Et le résultat doit être contenu dans un fichier texte (resultat.log)

Vérifiez les résultats.

Références

- <https://learn.microsoft.com/en-us/sql/sql-server/?view=sql-server-ver16>
- <https://www.w3schools.com/sql>
- https://www.w3schools.com/sql/sql_ref_sqlserver.asp
- <https://www.w3schools.in/dbms/database-normalization/>
- https://en.wikipedia.org/wiki/Database_normalization#Satisfying_3NF
- <https://www.kodyaz.com/sql-server-2019/list-of-transact-sql-functions-on-sql-server-2019.aspx>
- <https://www.javatpoint.com/sql-server-create-table>

Annexes

Annexe 1 – Best practices of Relational Database Design (20 steps)

Another best practices steps suggested in CodeBalance Software Engineering Blog:

Source: <http://codebalance.blogspot.com/2011/07/20-database-design-best-practices.html>

1. Use well defined and consistent names for tables and columns (e.g. School, StudentCourse, CourseID ...).
2. Use singular for table names (i.e. use StudentCourse instead of StudentCourses). Table represents a collection of entities, there is no need for plural names.
3. Don't use spaces for table names. Otherwise you will have to use '{', '[', '"' etc. characters to define tables (i.e. for accesing table Student Course you'll write "Student Course". StudentCourse is much better).
4. Don't use unnecessary prefixes or suffixes for table names (i.e. use School instead of TblSchool, SchoolTable etc.).
5. Keep passwords as encrypted for security. Decrypt them in application when required.
6. Use integer id fields for all tables. If id is not required for the time being, it may be required in the future (for association tables, indexing ...).

7. Choose columns with the integer data type (or its variants) for indexing. varchar column indexing will cause performance problems.
8. Use bit fields for boolean values. Using integer or varchar is unnecessarily storage consuming. Also start those column names with "Is".
9. Provide authentication for database access. Don't give admin role to each user.
10. Avoid "select *" queries until it is really needed. Use "select [required_columns_list]" for better performance.
11. Use an ORM (object relational mapping) framework (i.e. hibernate, iBatis ...) if application code is big enough. Performance issues of ORM frameworks can be handled by detailed configuration parameters.
12. Partition big and unused/rarely used tables/table parts to different physical storages for better query performance.
13. For big, sensitive and mission critical database systems, use disaster recovery and security services like failover clustering, auto backups, replication etc.
14. Use constraints (foreign key, check, not null ...) for data integrity. Don't give whole control to application code.
15. Lack of database documentation is evil. Document your database design with ER schemas and instructions. Also write comment lines for your triggers, stored procedures and other scripts.
16. Use indexes for frequently used queries on big tables. Analyser tools can be used to determine where indexes will be defined. For queries retrieving a range of rows, clustered indexes are usually better. For point queries, non-clustered indexes are usually better.
17. Database server and the web server must be placed in different machines. This will provide more security (attackers can't access data directly) and server CPU and memory performance will be better because of reduced request number and process usage.
18. Image and blob data columns must not be defined in frequently queried tables because of performance issues. These data must be placed in separate tables and their pointer can be used in queried tables.
19. Normalization must be used as required, to optimize the performance. Under-normalization will cause excessive repetition of data, over-normalization will cause excessive joins across too many tables. Both of them will get worse performance.
20. Spend time for database modeling and design as much as required. Otherwise saved(!) design time will cause (saved(!) design time) * 10/100/1000 maintenance and re-design time.

Annexe 2 – Liste des Fonctions dans SQL Server 2019

Function Type	SQL Function Name
Aggregate Function	APPROX_COUNT_DISTINCT
Aggregate Function	AVG
Aggregate Function	CHECKSUM_AGG
Aggregate Function	COUNT
Aggregate Function	COUNT_BIG
Aggregate Function	GROUPING
Aggregate Function	GROUPING_ID
Aggregate Function	MAX
Aggregate Function	MIN
Aggregate Function	STDEV
Aggregate Function	STDEVP
Aggregate Function	STRING_AGG
Aggregate Function	SUM
Aggregate Function	VAR
Aggregate Function	VARP
Analytic Function	CUME_DIST
Analytic Function	FIRST_VALUE
Analytic Function	LAG
Analytic Function	LAST_VALUE
Analytic Function	LEAD
Analytic Function	PERCENT_RANK
Analytic Function	PERCENTILE_CONT
Analytic Function	PERCENTILE_DISC
Collation Function	COLLATIONPROPERTY
Collation Function	TERTIARY_WEIGHTS
Configuration Function	@@DATEFIRST
Configuration Function	@@DBTS
Configuration Function	@@LANGID
Configuration Function	@@LANGUAGE
Configuration Function	@@LOCK_TIMEOUT
Configuration Function	@@MAX_CONNECTIONS

Function Type	SQL Function Name
Configuration Function	@@MAX_PRECISION
Configuration Function	@@NESTLEVEL
Configuration Function	@@OPTIONS
Configuration Function	@@REMSERVER
Configuration Function	@@SERVERNAME
Configuration Function	@@SERVICENAME
Configuration Function	@@SPID
Configuration Function	@@TEXTSIZE
Configuration Function	@@VERSION
Conversion Function	CAST and CONVERT
Conversion Function	PARSE
Conversion Function	TRY_CAST
Conversion Function	TRY_CONVERT
Conversion Function	TRY_PARSE
Cryptographic function	ASYMKEY_ID
Cryptographic function	ASYMKEYPROPERTY
Cryptographic function	CERTENCODED
Cryptographic function	CERTPRIVATEKEY
Cryptographic function	DECRYPTBYASYMKEY
Cryptographic function	DECRYPTBYCERT
Cryptographic function	DECRYPTBYKEY
Cryptographic function	DECRYPTBYKEYAUTOASYMKEY
Cryptographic function	DecryptByKeyAutoCert
Cryptographic function	DECRYPTBYPASSPHRASE
Cryptographic function	ENCRYPTBYASYMKEY
Cryptographic function	ENCRYPTBYCERT
Cryptographic function	ENCRYPTBYKEY
Cryptographic function	ENCRYPTBYPASSPHRASE
Cryptographic function	HASHBYTES
Cryptographic function	IS_OBJECTSIGNED
Cryptographic function	KEY_GUID
Cryptographic function	KEY_ID
Cryptographic function	KEY_NAME

Function Type	SQL Function Name
Cryptographic function	SIGNBYASYMKEY
Cryptographic function	SIGNBYCERT
Cryptographic function	SYMKEYPROPERTY
Cryptographic function	VERIFYSIGNEDBYASMKEY
Cryptographic function	VERIGYSIGNEDBYCERT
Cursor Function	@@CURSOR_ROWS
Cursor Function	@@FETCH_STATUS
Cursor Function	CURSOR_STATUS
Data Type Function	DATALENGTH
Data Type Function	IDENT_CURRENT
Data Type Function	IDENT_INCR
Data Type Function	IDENT_SEED
Data Type Function	IDENTITY (Function)
Data Type Function	SQL_VARIANT_PROPERTY
Date & Time Function	@@DATEFIRST
Date & Time Function	@@LANGUAGE
Date & Time Function	CURRENT_TIMESTAMP
Date & Time Function	DATEADD
Date & Time Function	DATEDIFF
Date & Time Function	DATEDIFF_BIG
Date & Time Function	DATEFROMPARTS
Date & Time Function	DATENAME
Date & Time Function	DATEPART
Date & Time Function	DATETIME2FROMPARTS
Date & Time Function	DATETIMEFROMPARTS
Date & Time Function	DATETIMEOFFSETFROMPARTS
Date & Time Function	DAY
Date & Time Function	EOMONTH
Date & Time Function	GETDATE
Date & Time Function	GETUTCDATE
Date & Time Function	ISDATE
Date & Time Function	MONTH
Date & Time Function	SET DATEFIRST

Function Type	SQL Function Name
Date & Time Function	SET DATEFORMAT
Date & Time Function	SET LANGUAGE
Date & Time Function	SMALLDATETIMEFROMPARTS
Date & Time Function	sp_helplanguage
Date & Time Function	SWITCHOFFSET
Date & Time Function	SYSDATETIME
Date & Time Function	SYSDATETIMEOFFSET
Date & Time Function	SYSUTCDATETIME
Date & Time Function	TIMEFROMPARTS
Date & Time Function	TODATETIMEOFFSET
Date & Time Function	YEAR
JSON Function	ISJSON
JSON Function	JSON_MODIFY
JSON Function	JSON_QUERY
JSON Function	JSON_VALUE
Logical Function	CHOOSE
Logical Function	IIF
Mathematical Function	ABS
Mathematical Function	ACOS
Mathematical Function	ASIN
Mathematical Function	ATAN
Mathematical Function	ATN2
Mathematical Function	CEILING
Mathematical Function	COS
Mathematical Function	COT
Mathematical Function	DEGREES
Mathematical Function	EXP
Mathematical Function	FLOOR
Mathematical Function	LOG
Mathematical Function	LOG10
Mathematical Function	PI
Mathematical Function	POWER
Mathematical Function	RADIANS

Function Type	SQL Function Name
Mathematical Function	RAND
Mathematical Function	ROUND
Mathematical Function	SIGN
Mathematical Function	SIN
Mathematical Function	SQRT
Mathematical Function	SQUARE
Mathematical Function	TAN
Metadata Function	@@PROCID
Metadata Function	APP_NAME
Metadata Function	APPLOCK_MODE
Metadata Function	APPLOCK_TEST
Metadata Function	ASSEMBLYPROPERTY
Metadata Function	COL_LENGTH
Metadata Function	COL_NAME
Metadata Function	COLUMNPROPERTY
Metadata Function	DATABASE_PRINCIPAL_ID
Metadata Function	DATABASEPROPERTYEX
Metadata Function	DB_ID
Metadata Function	DB_NAME
Metadata Function	FILE_ID
Metadata Function	FILE_IDEX
Metadata Function	FILE_NAME
Metadata Function	FILEGROUP_ID
Metadata Function	FILEGROUP_NAME
Metadata Function	FILEGROUPPROPERTY
Metadata Function	FILEPROPERTY
Metadata Function	FULLTEXTCATALOGPROPERTY
Metadata Function	FULLTEXTSERVICEPROPERTY
Metadata Function	INDEX_COL
Metadata Function	INDEXKEY_PROPERTY
Metadata Function	INDEXPROPERTY
Metadata Function	NEXT VALUE FOR
Metadata Function	OBJECT_DEFINITION

Function Type	SQL Function Name
Metadata Function	OBJECT_ID
Metadata Function	OBJECT_NAME
Metadata Function	OBJECT_SCHEMA_NAME
Metadata Function	OBJECTPROPERTY
Metadata Function	OBJECTPROPERTYEX
Metadata Function	ORIGINAL_DB_NAME
Metadata Function	PARSENAME
Metadata Function	SCHEMA_ID
Metadata Function	SCHEMA_NAME
Metadata Function	SCOPE_IDENTITY
Metadata Function	SERVERPROPERTY
Metadata Function	STATS_DATE
Metadata Function	TYPE_ID
Metadata Function	TYPE_NAME
Metadata Function	TYPEPROPERTY
Metadata Function	VERSION
Ranking Function	DENSE_RANK
Ranking Function	NTILE
Ranking Function	RANK
Ranking Function	ROW_NUMBER
Replication Function	PUBLISHINGSERVERNAME
Rowset Function	OPENDATASOURCE
Rowset Function	OPENJSON
Rowset Function	OPENQUERY
Rowset Function	OPENROWSET
Rowset Function	OPENXML
Security Function	CERTENCODED
Security Function	CERTPRIVATEKEY
Security Function	CURRENT_USER
Security Function	DATABASE_PRINCIPAL_ID
Security Function	HAS_PERMS_BY_NAME
Security Function	IS_MEMBER
Security Function	IS_ROLEMEMBER

Function Type	SQL Function Name
Security Function	IS_SRVROLEMEMBER
Security Function	LOGINPROPERTY
Security Function	ORIGINAL_LOGIN
Security Function	PERMISSIONS
Security Function	PWDCOMPARE
Security Function	PWDENCRYPT
Security Function	SCHEMA_ID
Security Function	SCHEMA_NAME
Security Function	SESSION_USER
Security Function	SUSER_ID
Security Function	SUSER_NAME
Security Function	SUSER_SID
Security Function	SUSER_SNAME
Security Function	sys.fn_builtin_permissions
Security Function	sys.fn_get_audit_file
Security Function	sys.fn_my_permissions
Security Function	SYSTEM_USER
Security Function	USER_ID
Security Function	USER_NAME
String Function	ASCII
String Function	CHAR
String Function	CHARINDEX
String Function	CONCAT
String Function	CONCAT_WS
String Function	DIFFERENCE
String Function	FORMAT
String Function	LEFT
String Function	LEN
String Function	LOWER
String Function	LTRIM
String Function	NCHAR
String Function	PATINDEX
String Function	QUOTENAME

Function Type	SQL Function Name
String Function	REPLACE
String Function	REPLICATE
String Function	REVERSE
String Function	RIGHT
String Function	RTRIM
String Function	SOUNDEX
String Function	SPACE
String Function	STR
String Function	STRING_AGG
String Function	STRING_ESCAPE
String Function	STRING_SPLIT
String Function	STUFF
String Function	SUBSTRING
String Function	TRANSLATE
String Function	TRIM
String Function	UNICODE
String Function	UPPER
System Function	\$PARTITION
System Function	@@ERROR
System Function	@@IDENTITY
System Function	@@PACK_RECEIVED
System Function	@@ROWCOUNT
System Function	@@TRANCOUNT
System Function	BINARY_CHECKSUM
System Function	CHECKSUM
System Function	COMPRESS
System Function	CONNECTIONPROPERTY
System Function	CONTEXT_INFO
System Function	CURRENT_REQUEST_ID
System Function	CURRENT_TRANSACTION_ID
System Function	DECOMPRESS
System Function	ERROR_LINE
System Function	ERROR_MESSAGE

Function Type	SQL Function Name
System Function	ERROR_NUMBER
System Function	ERROR_PROCEDURE
System Function	ERROR_SEVERITY
System Function	ERROR_STATE
System Function	FORMATMESSAGE
System Function	GET_FILESTREAM_TRANSACTION_CONTEXT
System Function	GETANSINULL
System Function	HOST_ID
System Function	HOST_NAME
System Function	ISNULL
System Function	ISNUMERIC
System Function	MIN_ACTIVE_ROWVERSION
System Function	NEWID
System Function	NEWSEQUENTIALID
System Function	ROWCOUNT_BIG
System Function	SESSION_CONTEXT
System Function	SESSION_ID
System Function	XACT_STATE
System Statistical Function	@@CONNECTIONS
System Statistical Function	@@CPU_BUSY
System Statistical Function	@@IDLE
System Statistical Function	@@IO_BUSY
System Statistical Function	@@PACK_RECEIVED
System Statistical Function	@@PACK_SENT
System Statistical Function	@@PACKET_ERRORS
System Statistical Function	@@TIMETICKS
System Statistical Function	@@TOTAL_ERRORS
System Statistical Function	@@TOTAL_READ
System Statistical Function	@@TOTAL_WRITE
System Statistical Function	fn_virtualfilestats
Text and Image Function	TEXTPTR
Text and Image Function	TEXTVALID
Trigger Function	COLUMNS_UPDATED

Function Type	SQL Function Name
Trigger Function	EVENTDATA
Trigger Function	TRIGGER_NESTLEVEL
Trigger Function	UPDATE()