What is Azure SQL Service?

Azure SQL allows hosting our Microsoft SQL Server workloads on the Azure cloud. We have three major ways of hosting the SQL database on the Azure Cloud.

1) Azure SQL Database: It is a fully managed database service in Azure. It is always running on the current version of the SQL Server database engine and patched OS with 99.99% availability.

2) Azure SQL Managed Instance: It is a fully managed service. It's for those companies who want to migrate their existing SQL workloads to the cloud.

3) SQL Server on Azure VMs: It is when we want to have more control over the SQL server environment.

What is Microsoft Azure SQL Database?

Microsoft Azure SQL Database is a relational database-as-a-service that is reliable and secure, and it gives high performance without having to worry about any infrastructure.

It supports relational, JSON, XML, and spatial data structures.

It offers the Azure PaaS SQL capacities to build applications and websites, while Microsoft manages the infrastructure.

Types Of Deployment Options For The Database:

1) Single Database: A single database at Azure SQL platform is an equivalent of a usually contained SQL Server database, just hosted in the cloud. This database is isolated from other databases and managed via a server. When you assign resources to every single database, they belong to it only, not shared with other databases, under any service tier.

The single database deployment model is a solution for a cloud application that requires a single data source. Also, it is possible to scale up or down resources allocated to the particular database.

2) Elastics Pool: This deployment model relates to multiple databases with shared resources managed altogether via a logical server. You can move a single database into this elastic pool or remove it from there whenever you need. The elastic pool is a solution for several databases, where each database requires resources to work effectively. If you don't know how many resources each particular database will consume and how much you should allocate at once, then the work is challenging. The SQL pool in Azure resolves this issue in the following way: when any separate database sets unique and unpredictable resource requirements, the elastic pool allocates the necessary resources to the target database itself.

3) Managed Instance: The managed instance is a deployment option of Azure SQL providing near 100% compatibility with the latest SQL server on-premises.

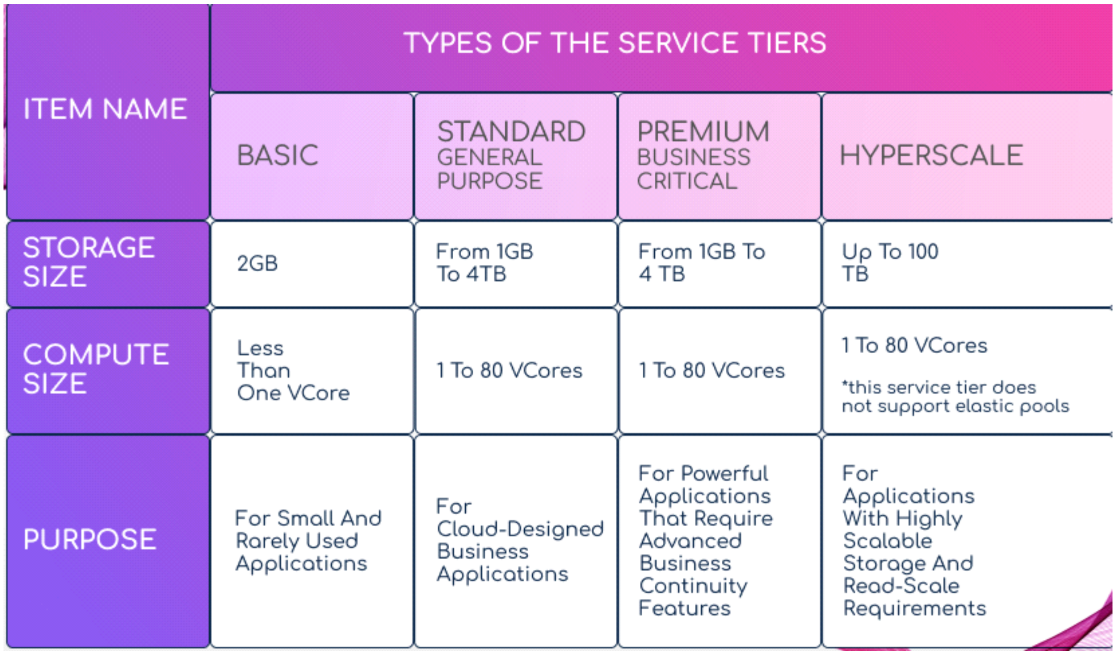
Azure SQL Database Offers Three Service Tiers:

1 The Basic service tier is the simplest option that supports one active operation at a time. It usually suits databases for small and rarely used applications. We won't pay close attention to it and proceed to more popular service tiers

2 The Standard (General Purpose) service tier is a default option for both the Database and Managed Instance in Azure SQL. The Standard service tier can be used by most cloud apps perfectly. The storage size varies from 1GB to 4TB and Azure takes care of all upgrades and patching

3 The Premium (Business Critical) service tier is designed for powerful applications that demand low-latency responses, fast recovery in case of any infrastructure failures, analyzing data loads, etc. If an application is critical for your business, you should go with the Premium service tier. The storage size is the same as for the General Purpose service tier – from 1Gb to 4 TB

4 The Hyperscale service tier is a new option that is present in the vCore model only. It offers much more scalable storage with significantly more power for computing, more than in any other service tier. The size of a database can be up to 100 TB in the Hyperscale service tier and databases don't have a definite maximal size initially. It grows when you need it. Backups and restores are performed much faster and the overall performance is much higher, no matter which data volumes it operates.



ITEM NAME	TYPES OF THE SERVICE TIERS			
	BASIC	STANDARD GENERAL PURPOSE	PREMIUM BUSINESS CRITICAL	HYPERSCALE
STORAGE SIZE	2GB	From 1GB To 4TB	From 1GB To 4 TB	Up To 100 TB
COMPUTE SIZE	Less Than One VCore	1 To 80 VCores	1 To 80 VCores	1 To 80 VCores *this service tier does not support elastic pools
PURPOSE	For Small And Rarely Used Applications	For Cloud-Designed Business Applications	For Powerful Applications That Require Advanced Business Continuity Features	For Applications With Highly Scalable Storage And Read-Scale Requirements

Purchasing models

vCore purchasing model

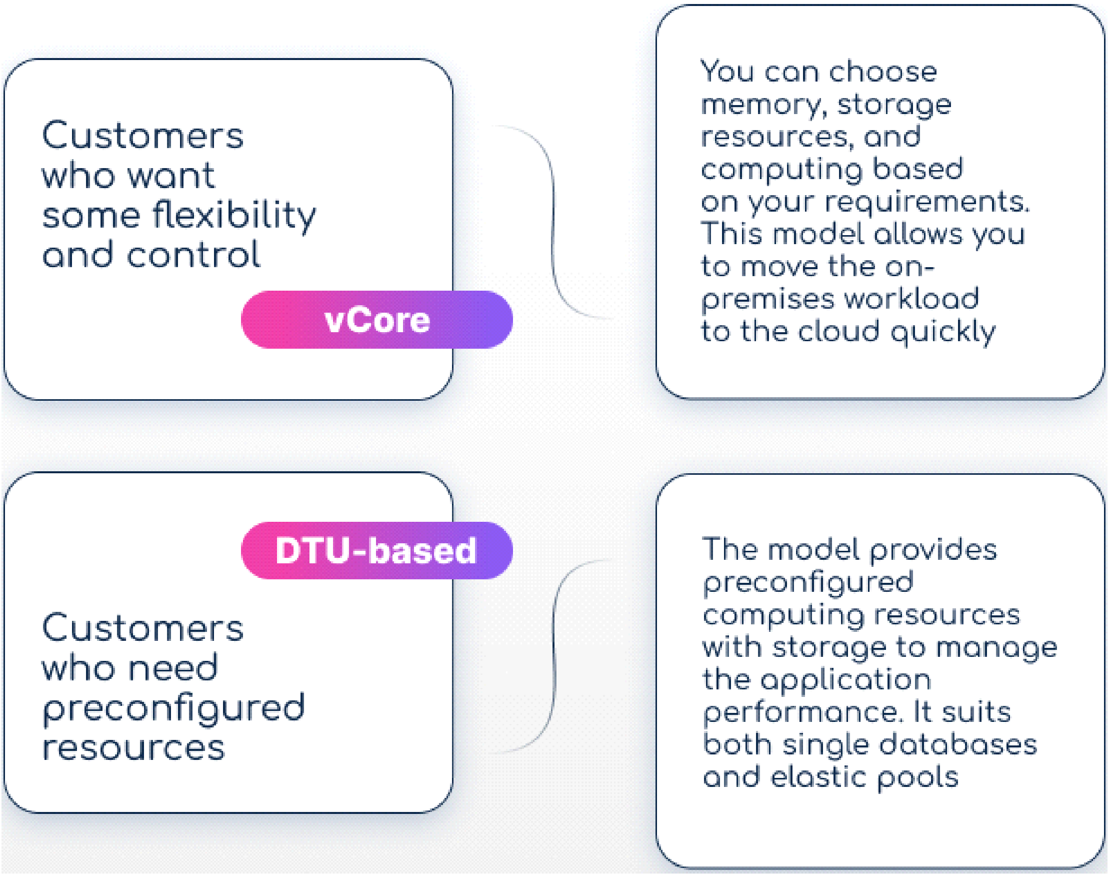
vCore stands for “virtual core” and represents logical CPU. The model is also known as serverless. If you choose this model, you can specify the hardware characteristics you want to have in the cloud, such as memory, storage, the number of cores, etc. This option lets you “transfer” the requirements from on-premise to the cloud, ensuring that you have the required environment to work with databases most efficiently. Also, you can migrate your applications to the cloud easily.

The price will depend on the service tier and your particular requirements for the resources, storage, and hardware configuration. The vCore-based model is available in Standard, Premium, and Hyperscale service tiers.

DTU-based purchasing model

DTU stands for the Database Transaction Unit – a measure for the bundle of computing and storage. The DTU-based purchasing model means that you receive the preconfigured bundle that covers the CPU capabilities and memory under fixed monthly payments. It suits both single databases and elastic pools.

This model is available in Basic, Standard, and Premium service tiers. The Standard and Premium tiers allow you to add more storage. In general, under this model, you get a certain granted level of resources for any database, and it does not matter if there are other databases. Thus, each database will perform predictably.



vCore

Customers who want some flexibility and control

You can choose memory, storage resources, and computing based on your requirements. This model allows you to move the on-premises workload to the cloud quickly

DTU-based

Customers who need preconfigured resources

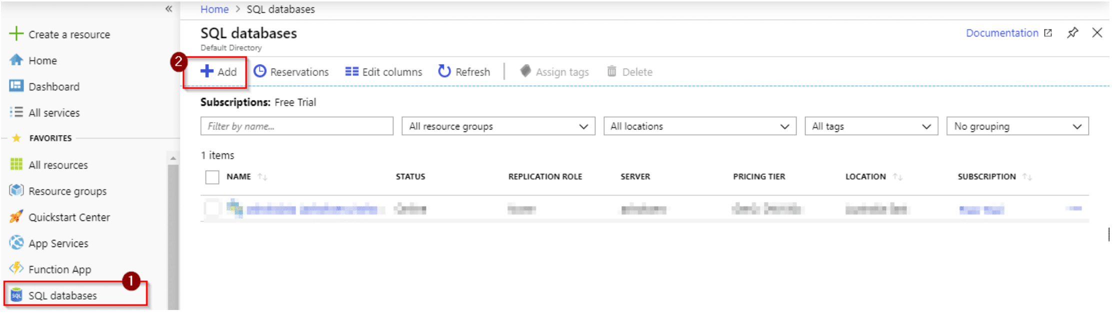
The model provides preconfigured computing resources with storage to manage the application performance. It suits both single databases and elastic pools

Microsoft Azure SQL Database Hands-on

Step 1: Login to the portal first and search for Database under services

1. Select SQL Databases under Database service

2. Select Add to create a new database



Subscriptions: Free Trial

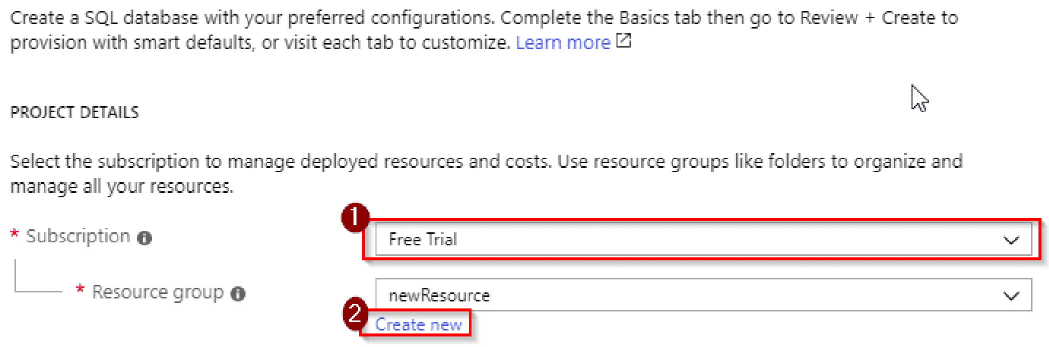
1 items

NAME	STATUS	REPLICATION ROLE	SERVER	PRICING TIER	LOCATION	SUBSCRIPTION
[redacted]	[redacted]	[redacted]	[redacted]	[redacted]	[redacted]	[redacted]

Step 2: You will see the following options on your screen where you will have to choose the type of subscription and resource group

1. If your account is on the free trial, please select Free Trial under Subscription

2. If you do not have any Resource group created, click on Create new



Create a SQL database with your preferred configurations. Complete the Basics tab then go to Review + Create to provision with smart defaults, or visit each tab to customize. Learn more

PROJECT DETAILS

Select the subscription to manage deployed resources and costs. Use resource groups like folders to organize and manage all your resources.

* Subscription: Free Trial

* Resource group: newResource

Step 3: To create a new resource, you have to name it. Here, the resource group is named as SqlResources

Step 4: Once you have created your resource group, you will now have to fill in some details

1. You will have to name your database. Here, it is named as SampleDatabase

2. After naming your database, you will have to create a server. Once you click on Create new, you will be prompted to set up some details, which will be explained in Step 5. Once done, close the pop-up window and continue with further steps

3. You will not need the elastic pool and, hence, select No

4. Once you are done creating a server, you will be able to click on the Configure database, where you will have to select pricing options.

Step 5:Enter a server name as your wish. Here, the name is helloworld1

1. Enter the username as per your choice

2. Enter a strong password containing uppercase alphabets, special characters, and numbers

3. Confirm your password by re-entering it

4. Select the location of your server. Southeast Asia has been selected here

5. Tick the box for Azure Services to access the server

6. Click on Select and close the window

Step 6: After completing Step 5, go to the 4th point of Step 4 for pricing and storage options. You will be redirected to where you can configure it according to your convenience. Select vCore

Step 7: After selecting vCore, you can see options like Compute tier which has two options. Select Serverless and you can select the size of your data from 1 GB to 512 GB. Here, it is kept at the minimum level, 1 GB, and it will cost Rs.11.86 from Rs.13,300 credits which you get for a free account

After configuring your settings, click on preview under Preview Terms.

1. Tick the box

2. Select OK

Step 8: Your changes will be saved, and you will be redirected to the screen with options as in Step 4, but with all the details filled in. Select Next: Additional settings > which is at the bottom and do the following steps:

1. Click on Review + create

Step 9: Once done with the deployment, when you go to SQL Databases, you can see your database that has been created. Select your database and do the next step

Step 10: Now that you have your database created, write a query on your data. Click on Query editor, and you will be prompted to give your log-in details that you set in Step 5

Step 11: After you login with the right credentials, you can see the database name on the left, expanding which you can see the option Tables

Step 12: Now set some security on your database. To do that, select your database from the SQL Databases list and do the following:

1. Copy your server name for future reference

2. Click on the Set server firewall

Step 13:

1. Click on Add client IP

2. You can see the details once you do the first step

3. Save the configuration

Step 14: Now create your own table using Microsoft SQL Server Management.

Once you have installed the software, you have to connect it to your server. For doing that, open the application and you will see a pop-up asking you to enter credentials.

1. Enter the server name that you copied and kept earlier

2. Under the Authentication drop-down menu, select SQL Server Authentication

3. Enter the username that you specified in the portal

4. Enter the correct password and select Connect

Step 15: After getting connected, you can see your server name on the left-hand side. Right-click on the name of your server and select New Query.

Azure SQL Managed Instance

The Azure SQL Database Managed Instance is a new implementation model of Azure SQL Database based on the VCore-based purchasing model.

Native virtual network implementation and connectivity to our on-premises environment using Azure Express Route or VPN Gateway.

SQL endpoint is exposed only through a private IP address, allowing safe connectivity from private Azure or hybrid networks.

It is a Single-tenant environment with dedicated underlying infrastructure (compute, storage).

Advantages of using Managed Instance

Easy lift and shift: Customers can lift and shift their on-premises SQL server to a Managed Instance that offers compatibility with SQL Server on-premises.

Fully managed PaaS: Azure SQL Database Managed Instance is designed for customers looking to migrate a large number of apps from on-premises self-built or ISV provided an environment to fully managed PaaS cloud environment.

New Business model: Competitive, transparent, and frictionless business model

SQL Server on Azure Virtual machine

SQL Server on Azure Virtual Machines enables you to use full versions of SQL Server in the cloud without having to manage any on-premises hardware. SQL Server virtual machines (VMs) also simplify licensing costs when you pay as you go.

Azure virtual machines run in many different geographic regions around the world. They also offer a variety of machine sizes. The virtual machine image gallery allows you to create a SQL Server VM with the right version, edition, and operating system. This makes virtual machines a good option for many different SQL Server workloads.

Key Features of Using SQL Server on Azure Virtual machine

1) Automated updates

SQL Server on Azure Virtual Machines can use Automated Patching to schedule a maintenance window for installing important windows and SQL Server updates automatically.

2) Automated backups

SQL Server on Azure Virtual Machines can take advantage of Automated Backup, which regularly creates backups of your database to blob storage. You can also manually use this technique.

Azure also offers an enterprise-class backup solution for SQL Server running in Azure VMs. A fully-managed backup solution, it supports Always On availability groups, long-term retention, point-in-time recovery, and central management and monitoring.

3) High availability

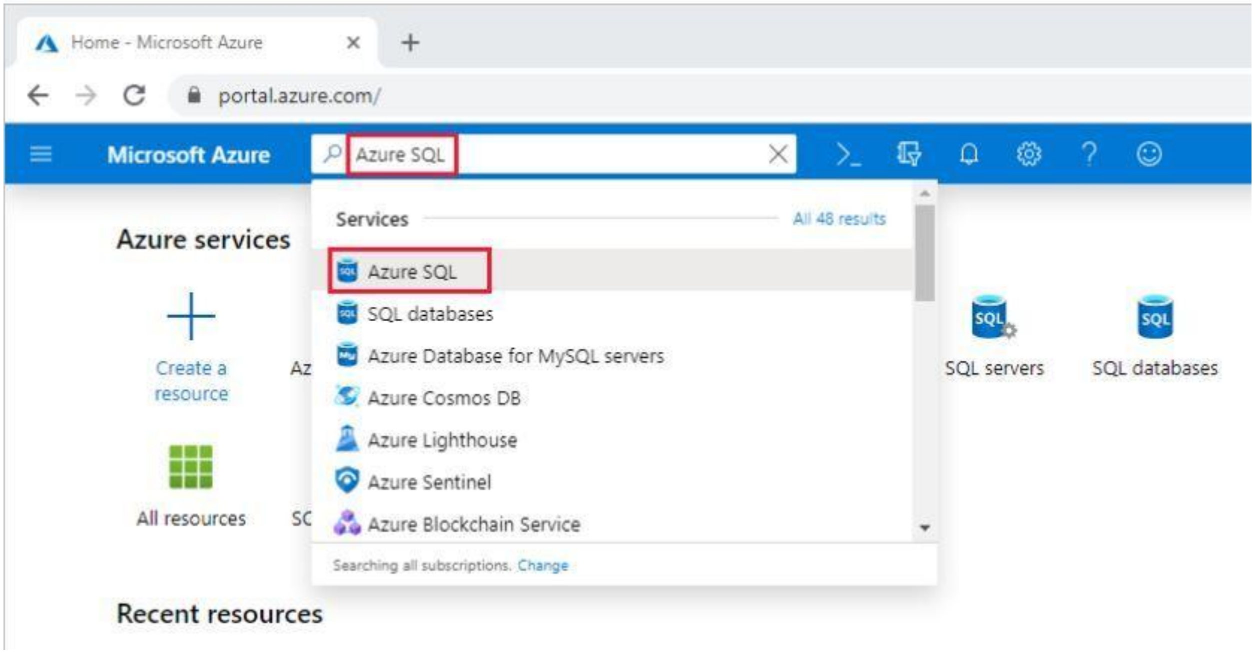
If you require high availability, consider configuring SQL Server Availability Groups. This involves multiple instances of SQL Server on Azure Virtual Machines in a virtual network. You can configure your high-availability solution manually, or you can use templates in the Azure portal for automatic configuration. For an overview of all high-availability options, see High Availability and Disaster Recovery for SQL Server in Azure Virtual Machines.

4) Performance

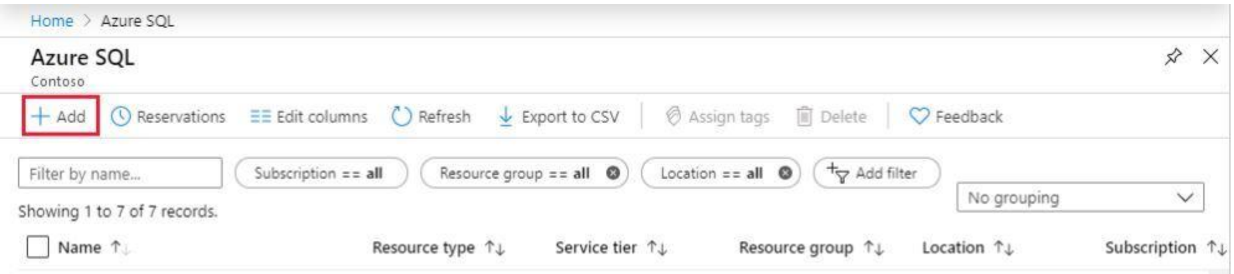
Azure virtual machines offer different machine sizes to meet various workload demands. SQL Server VMs also provide automated storage configuration, which is optimized for your performance requirements.

Create and manage Azure SQL resources with the Azure portal

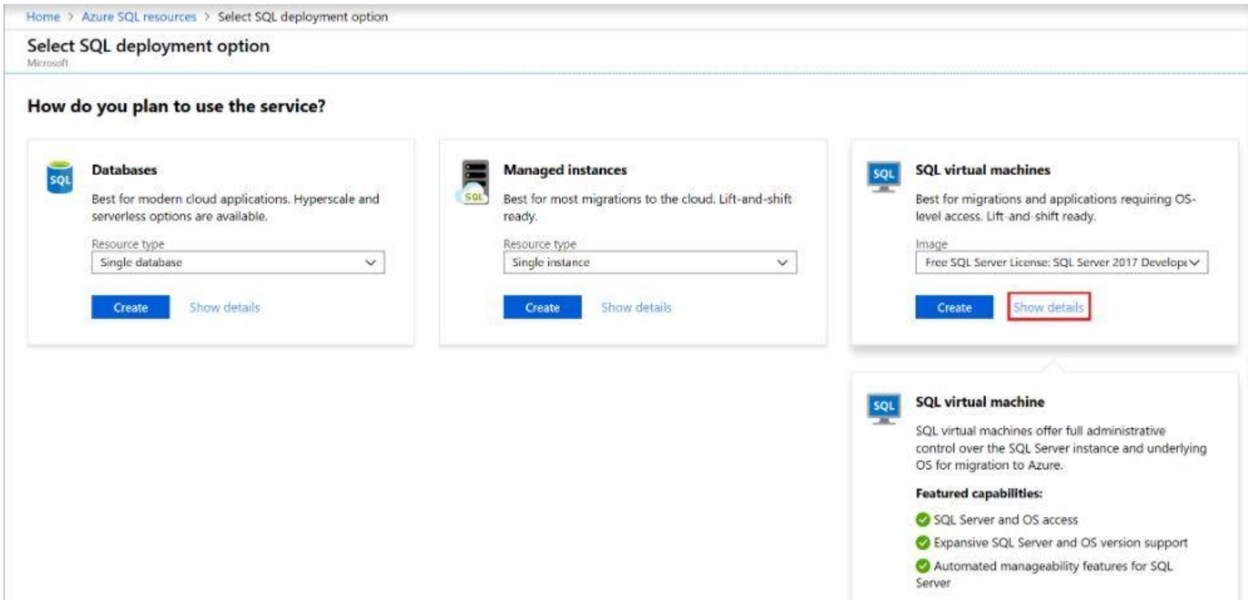
1) To access the Azure SQL resources page, select Azure SQL in the Azure portal menu, or search for and select Azure SQL from any page.



2) To manage existing resources, select the desired item in the list. To create new Azure SQL resources, select + Add.



3) After selecting + Add, view additional information about the different options by selecting Show details on any tile.



Select SQL deployment option

How do you plan to use the service?

Databases

Best for modern cloud applications. Hyperscale and serverless options are available.

Resource type: Single database

Managed instances

Best for most migrations to the cloud. Lift-and-shift ready.

Resource type: Single instance

SQL virtual machines

Best for migrations and applications requiring OS-level access. Lift-and-shift ready.

Image: Free SQL Server License: SQL Server 2017 Develope

SQL virtual machine

SQL virtual machines offer full administrative control over the SQL Server instance and underlying OS for migration to Azure.

Featured capabilities:

- ✓ SQL Server and OS access
- ✓ Expansive SQL Server and OS version support
- ✓ Automated manageability features for SQL Server

4) Provide basic details

On the Basics tab, provide the following information:

1. In the Project Details section, select your Azure subscription and then select Create new to create a new resource group. Type SQLVM-RG for the name.

5) Under Instance details:

1. Type SQLVM for the Virtual machine name.
2. Choose a location for your Region.
3. For the purpose of this quickstart, leave Availability options set to No infrastructure redundancy required. To find out more information about availability options, see Availability.
4. In the Image list, select Free SQL Server License: SQL Server 2017 Developer on Windows Server 2016.
5. Choose to Change size for the Size of the virtual machine and select the A2 Basic offering. Be sure to clean up your resources once you're done with them to prevent any unexpected charges.

6) Under Administrator account, provide a username, such as azureuser and a password. The password must be at least 12 characters long and meet

the defined complexity requirements.

7) Under Inbound port rules, choose Allow selected ports and then select RDP (3389) from the drop-down.

SQL Server settings

On the SQL Server settings tab, configure the following options:

1. Under Security & Networking, select Public (Internet) for SQL Connectivity and change the port to 1401 to avoid using a well-known port number in the public scenario.

2. Under SQL Authentication, select Enable. The SQL login credentials are set to the same user name and password that you configured for the VM. Use the default setting for Azure Key Vault integration. Storage configuration is not available for the basic SQL Server VM image, but you can find more information about available options for other images at storage configuration.
3. 3) Change any other settings if needed, and then select Review + create.

Connect to SQL Server

1. In the portal, find the Public IP address of your SQL Server VM in the Overview section of your virtual machine's properties.

2. On a different computer connected to the Internet, open SQL Server Management Studio (SSMS).

3. In the Connect to Server or Connect to Database Engine dialog box, edit the Server name value. Enter your VM's public IP address. Then add a comma and add the custom port (1401) that you specified when you configured the new VM. For example, 11.22.33.444,1401.

4. In the Authentication box, select SQL Server Authentication.

5. In the Login box, type the name of a valid SQL login.

6. In the Password box, type the password of the login.

7. Select Connect