Creating the Azure SQL Database

Azure SQL Database is a fully managed relational database service provided by Microsoft on the Azure cloud platform. It is based on SQL Server and offers features like high availability, automated backups, scaling, and security, without the need to manage the underlying infrastructure. It allows users to store and manage data in the cloud with minimal administrative effort, providing automatic updates and patching.
Azure SQL Database is designed for cloud applications that require high-performance, scalability, and reliability. It supports a variety of workloads, including web applications, data warehousing, and enterprise solutions. It offers several deployment options, such as single databases, elastic pools (for multiple databases), and managed instances (for near-complete SQL Server compatibility).

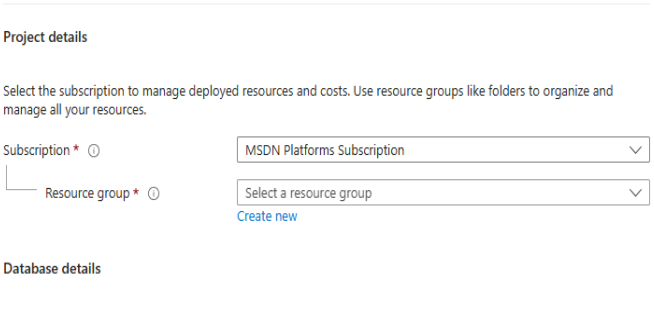
Use Cases of Azure SQL Database

1. **Web and Mobile Applications**: Azure SQL Database is commonly used for storing and managing data for web and mobile applications, offering scalable performance and reliable availability.
2. **Business Applications**: It is ideal for enterprise business applications like customer relationship management (CRM) and enterprise resource planning (ERP) systems that require a robust and scalable database solution.
3. **Data Warehousing and Analytics**: Azure SQL Database can be used to store and analyze large amounts of data for business intelligence, supporting data warehousing solutions and integration with Power BI and Azure Synapse Analytics.
4. **SaaS Applications**: It's widely used for Software-as-a-Service (SaaS) applications, where multi-tenant architectures require secure and scalable database solutions with automatic backups and high availability.
5. **Backup and Disaster Recovery**: Azure SQL Database supports disaster recovery scenarios, enabling automated backups and geo-redundancy for business continuity.

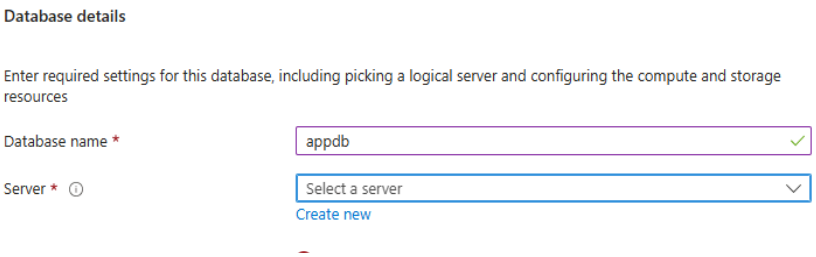
The end goal of this lab is to successfully deploy an Azure SQL Database on the Azure cloud platform. This involves selecting a resource group, providing a unique database name, and configuring a server with SQL authentication. We also choose the appropriate region, cost-effective pricing model, and backup options, while enabling a public endpoint for database access. Additionally, a firewall rule is set to allow access from the current client device. After reviewing the configuration, we proceed to create and deploy the SQL database, ensuring a fully functional, scalable, and secure database solution in the cloud.

To begin with the lab

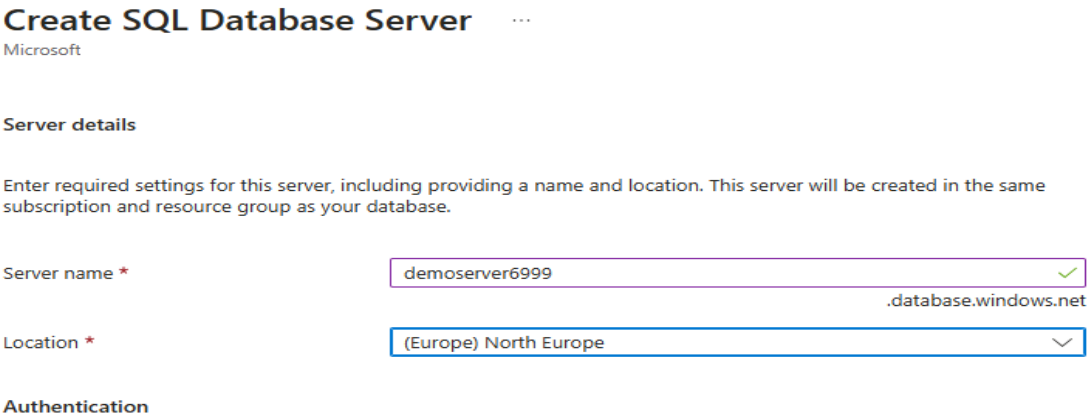
1. In previous lab we deploy a virtual machine. In this lab, we are going to deploy a SQL database. Go to the SQL Databases section in the Azure portal and click on Create.
2. Select an existing resource group or create a new one.



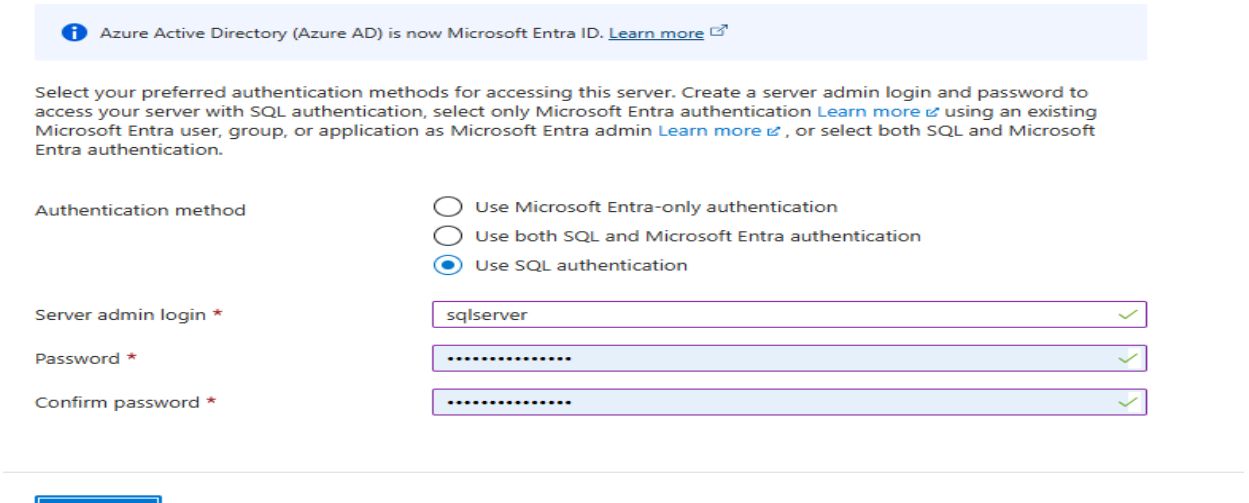
3. Provide a name for the database.



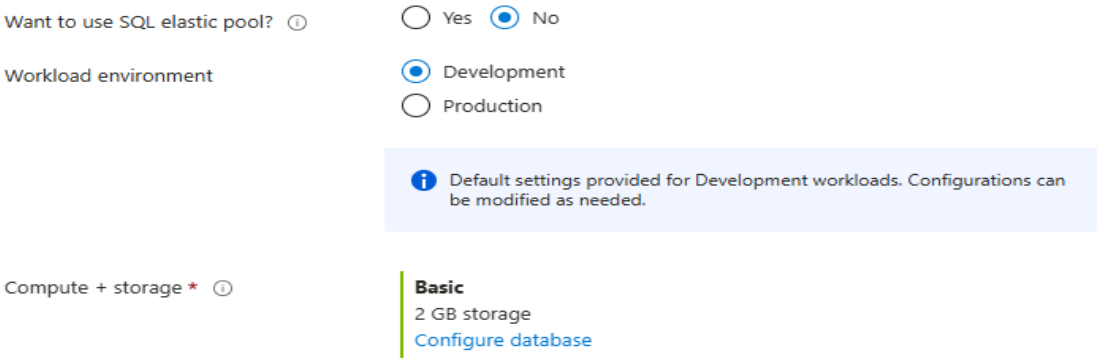
4. Choose a unique server name.
5. Select the Azure region (e.g., North Europe) to keep all resources in one location.



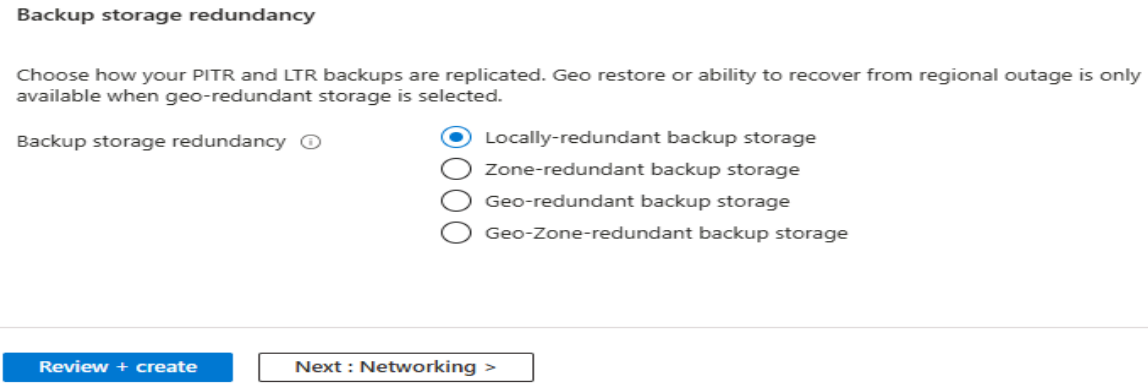
6. Choose SQL authentication and provide the server admin login credentials (username and password) for accessing the database server.



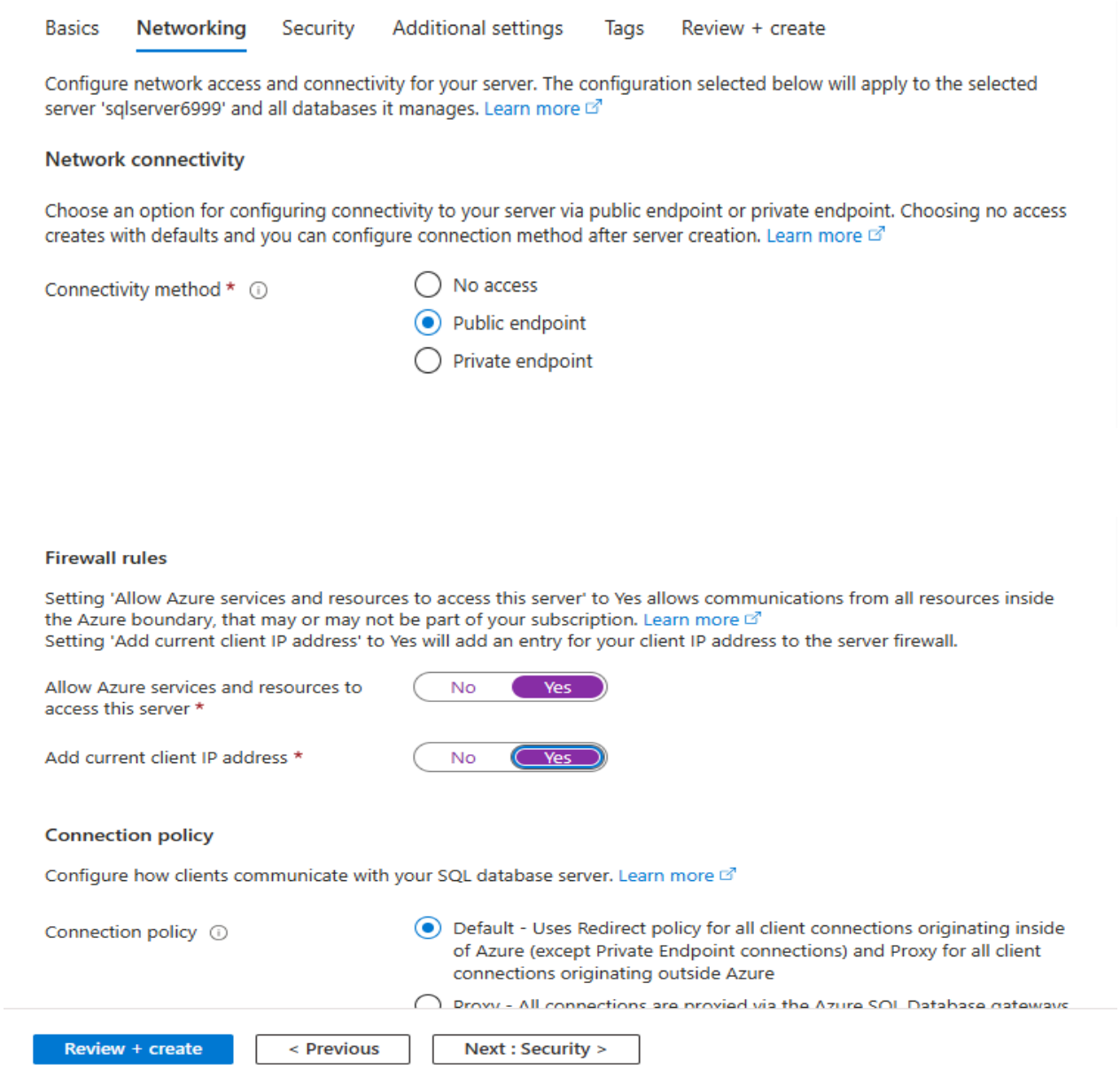
7. Basic Model is selected for cost efficiency, as it offers the least expensive option.



8. Select Locally-redundant backup storage to save on backup storage costs.



9. Choose a public endpoint to allow access to the database.
10. Set a firewall rule to allow the **current client IP address** (the device used to connect to the database).



11. After configuring security and additional settings, click on **Review + Create**.
12. Review the configuration, then click **Create** to begin the deployment.

Your deployment is complete

Deployment name	: Microsoft.SQLDatabase.newDatabase...	Start time	: 12/17/2024, 1:30:06 PM
Subscription	: MSDN Platforms Subscription	Correlation ID	: 57ec337d-e4bc-4ea3-bb56-95e819b57...
Resource group	: Demorg		

Deployment details

Next steps

Go to resource