Student's Full Name : ROHIT SURYA AVB

Course Title : Introduction to Cloud Computing; Cloud Data Platform

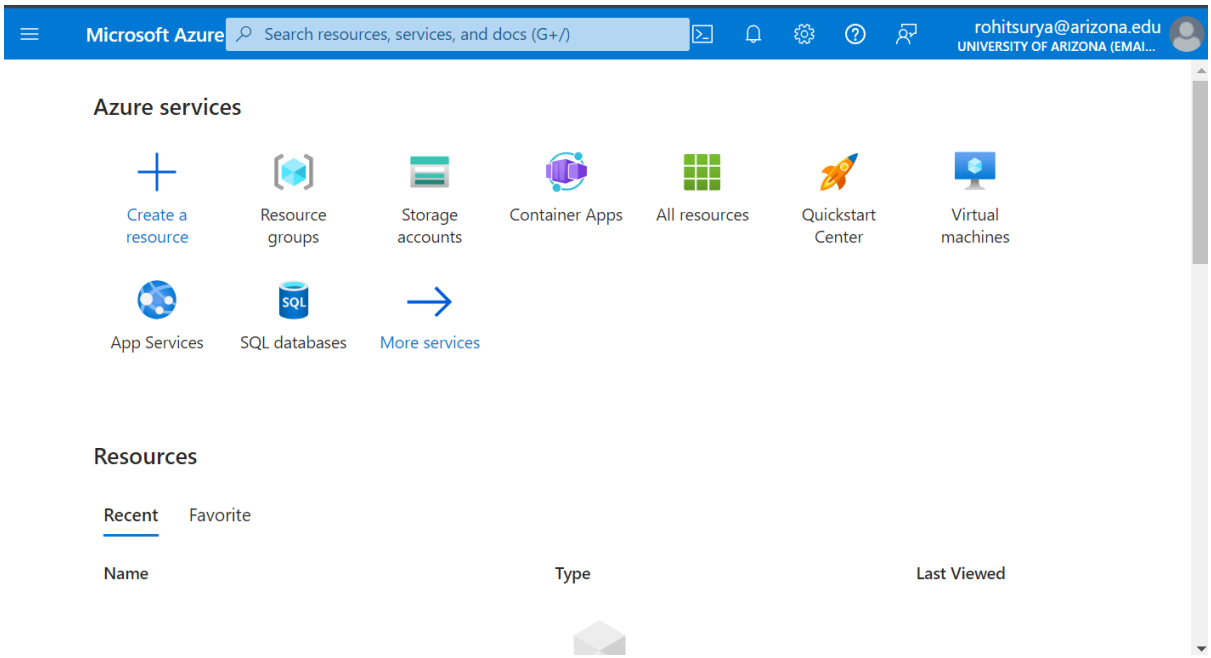
Term name and year : Fall 2023

Submission Week : *Assessment Week 6*

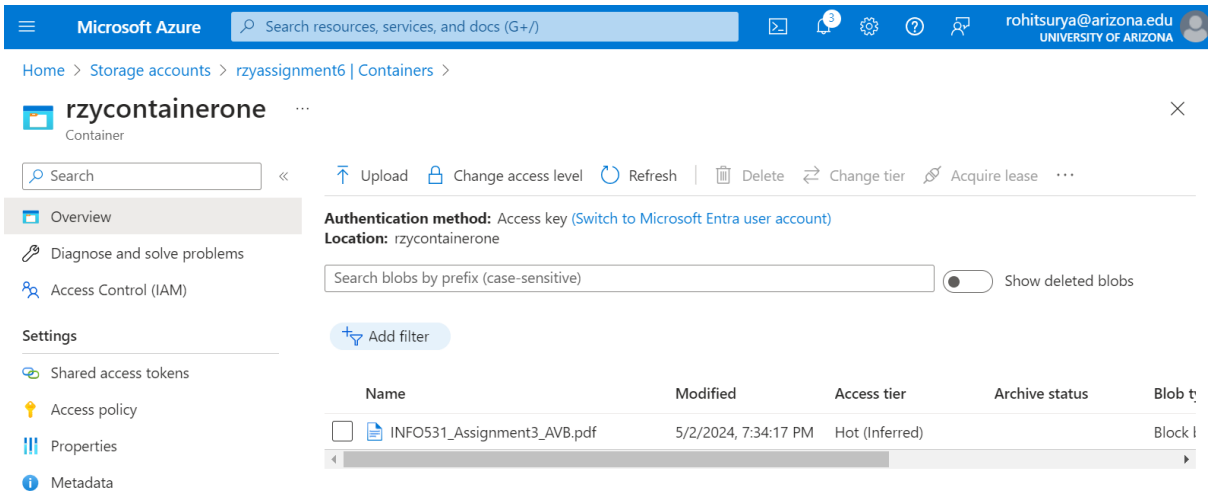
Instructor's Name : Dr.Nayem Rahman

Date of Submission : *2nd May 2024*

Q1. Create your Azure free account

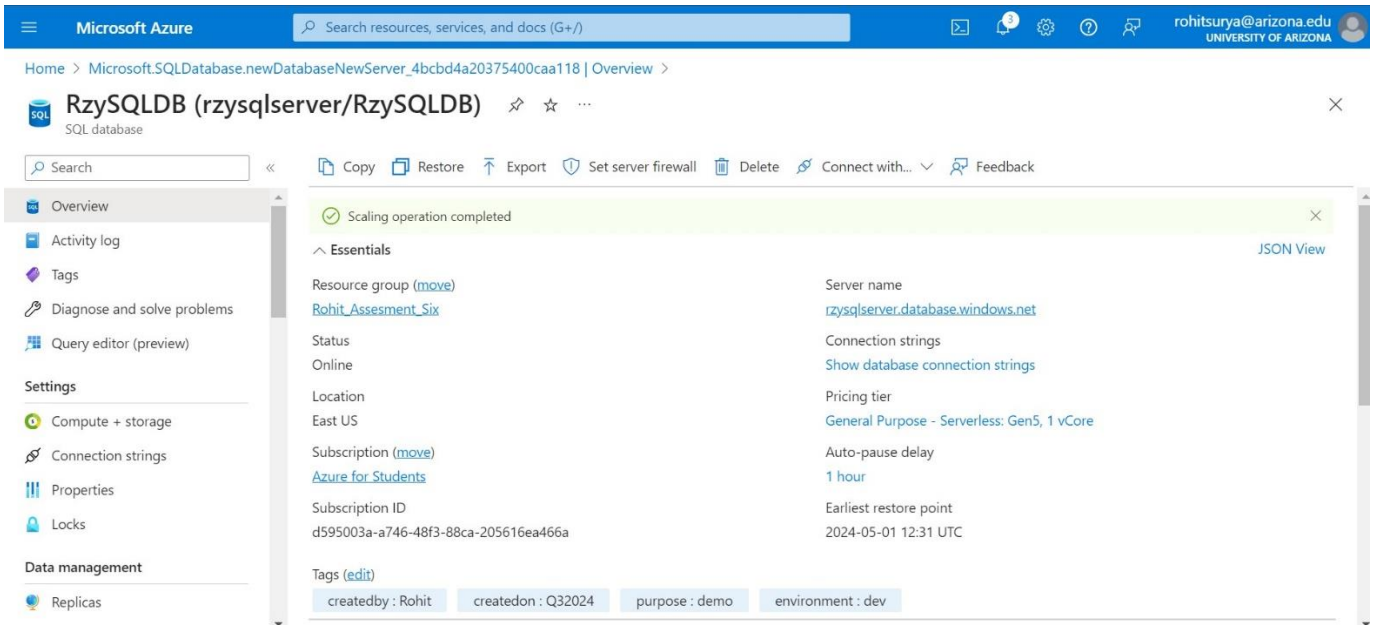


Q2. Create a container in Azure Storage, and to upload and download block blobs in that container.



I have created a new container and also uploaded a mock file for checks.

Q3. Create a SQL database on Azure

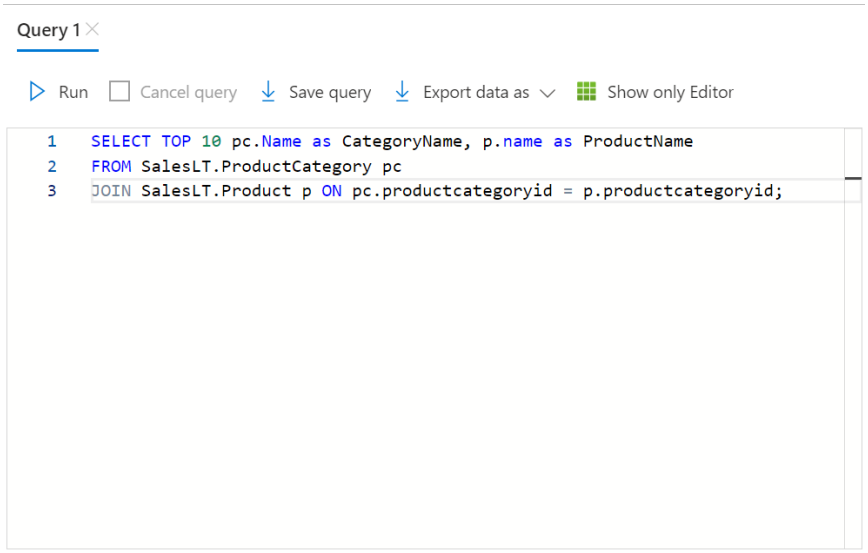


Q4. Summarize SQL database creation in Azure

Answer :

Creating a single SQL database in Azure Portal involves a series of straightforward steps that exemplify the simplicity of cloud-based database management. Through this process, I have gained insights into setting up and configuring a database in a cloud environment. Starting from the Azure Portal, I navigated through creating a new SQL database. The database creation form allowed me to configure various aspects such as database name, the performance tier (which determines the processing power and storage), and additional settings like data collation, which affects how data is sorted and compared in the database.

Q5. Query the SQL database in Azure



Results | Messages

Search to filter items...

CategoryName	ProductName
Road Frames	HL Road Frame - Black, 58
Road Frames	HL Road Frame - Red, 58
Helmets	Sport-100 Helmet, Red
Helmets	Sport-100 Helmet, Black
Socks	Mountain Bike Socks, M
Socks	Mountain Bike Socks, L
Helmets	Sport-100 Helmet, Blue
Caps	AWC Logo Cap
Jerseys	Long-Sleeve Logo Jersey, S
Jerseys	Long-Sleeve Logo Jersey, M

Q6. Describe the SQL database "SalesLT" on Azure.

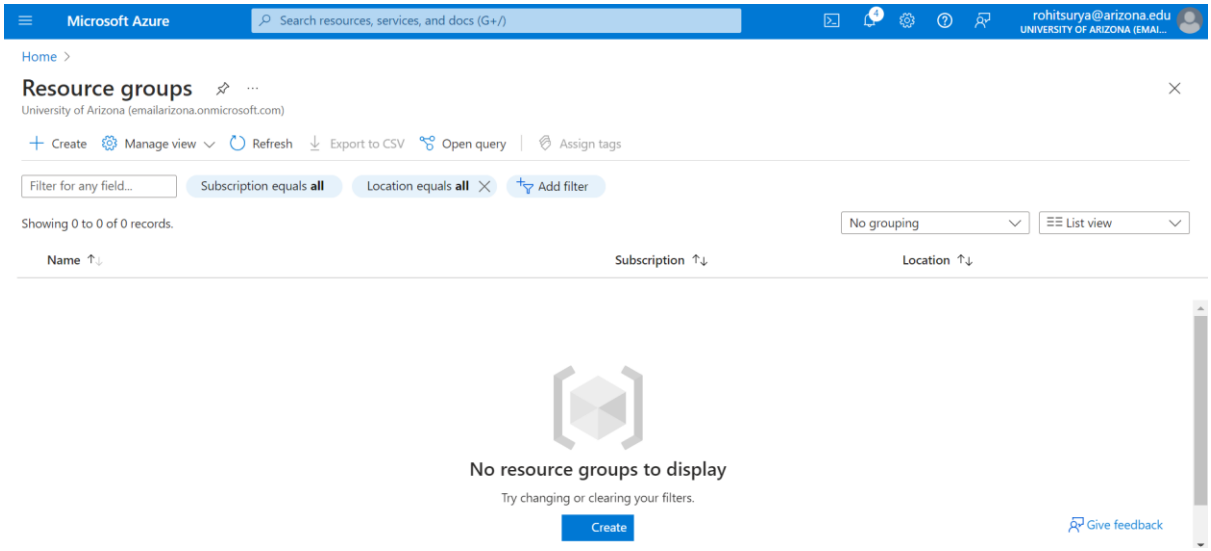
Answer :

The "SalesLT" schema in an Azure SQL Database encompasses several tables such as Address, Customer, Product, ProductCategory, ProductDescription, ProductModel, SalesOrderDetail, and SalesOrderHeader. Each of these tables features a unique primary key column, such as AddressID for Address, CustomerID for Customer, and ProductID for Product, which serves to uniquely identify each record within the respective tables.

The data types present within these tables are varied, typically including integers (int), strings (nvarchar), dates (datetime), monetary values (money), and small integers (smallint). Relationships between these tables are forged through the use of foreign key columns. For example, CustomerID in the SalesOrderHeader table references CustomerID in the Customer table, and ProductID in the SalesOrderDetail table links to ProductID in the Product table. Complex many-to-many relationships are handled by intermediary tables like ProductModelProductDescription, which connect ProductModelID with ProductDescriptionID.

Overall, the SalesLT schema efficiently organizes sales-related data by employing primary keys for record identification, utilizing various data types to handle diverse data formats, and establishing connections between tables through foreign keys to support effective data retrieval and management in the Azure SQL Database.

Q7. Clean up resources in Azure



Q8. Summarize the article on "Understanding Data Store Models" .

Answer :

The article on "Understanding Data Store Models" from Microsoft Learn emphasizes the concept of polyglot persistence, which advocates for using different data stores tailored to specific data types and workload requirements. It discusses various data store models such as relational databases, key/value stores, document databases, graph databases, and more. Each model is suited to particular patterns of data access and manipulation, underscoring the importance of selecting the right data store based on factors like data structure, scalability, and transaction requirements. The guide also touches on multi-model support where a single database can handle multiple data storage models, enhancing flexibility in managing data.

Q9. Summarize the article on "Understanding the differences between NoSQL and relational databases".

Answer :

The article on "Understanding the differences between NoSQL and relational databases" explains that relational databases are structured and use SQL for queries, providing ACID compliance, but lack partition tolerance. NoSQL databases handle high-performance needs with ease-of-use, scalability, and availability, supporting unstructured data but generally not providing ACID guarantees across partitions. The CAP theorem illustrates trade-offs between consistency, availability, and partition tolerance, influencing database choice based on specific application needs.

Q10. Summarize the article on "Understanding Azure Cosmos DB".

Answer :

Azure Cosmos DB is a fully managed, globally distributed database service designed to provide high performance, high availability, and scalable throughput for your applications. It supports multiple data models including key-value, document, graph, and column-family formats, making it versatile for various use cases. Azure Cosmos DB ensures low-latency access and offers comprehensive SLAs on throughput, availability, and consistency. It's highly adaptable, allowing you to scale and replicate data across multiple regions easily, fitting well with global app development needs.

Q11. Summarize the article on "Azure Cosmos DB API for MongoDB".

Answer :

Azure Cosmos DB for MongoDB is a fully managed service that mirrors MongoDB's features, allowing users to apply their existing MongoDB skills and tools. It offers benefits such as automatic scaling, transparent sharding, and consistent, fast performance with a Request Unit (RU)-based cost structure. It supports active-active configurations across multiple regions, enhancing availability and disaster recovery. This service is ideal for applications requiring high scalability, low latency, and real-time analytics without the complexities of managing a database infrastructure.

Q12. Summarize the article on "Nodes and tables in Azure Database for PostgreSQL – Hyperscale".

Answer :

Azure Cosmos DB for PostgreSQL employs a "shared nothing" architecture with a coordinator and multiple worker nodes. It supports five types of tables including distributed, reference, and local tables, each serving different purposes and stored differently across nodes. Distributed tables are partitioned across workers, optimizing cross-machine parallelism, while reference tables are replicated across all nodes. The system uses algorithmic sharding to distribute rows across shards based on a designated distribution column, enhancing query performance and data management efficiency.

Q13. Summarize the article on "Overview - Azure Database for PostgreSQL - Flexible Server".

Answer :

The Azure Database for PostgreSQL Flexible Server is a highly configurable managed database service. It allows for granular control over database management and configuration, offering low-latency, high availability across multiple zones, and cost-effective scalability. This service is suitable for application developments needing extensive customization, managed maintenance windows, and zone-redundant high availability. It supports automated patching, performance scaling, and encrypted backups, making it ideal for enterprise-level deployments requiring robust, secure data handling capabilities.

Q14. Summarize the article on "Azure Database for PostgreSQL - Single Server".

Answer :

The article discusses Azure Database for PostgreSQL - Single Server, a fully managed database service offered by Azure. It provides an overview of its features and functionality, emphasizing its simplicity and ease of use. Azure Database for PostgreSQL - Single Server is suitable for various applications, including web, mobile, and custom applications. It offers features such as automated backups, high availability, and security enhancements. The article highlights the

benefits of using Azure Database for PostgreSQL - Single Server, such as reduced administrative overhead, improved scalability, and reliability. Additionally, it provides guidance on getting started with the service and offers resources for further exploration.

Q15. Summarize the article on "What is Azure Database for PostgreSQL?".

Answer :

The article provides an overview of Azure Database for PostgreSQL, a fully managed database service offered by Microsoft Azure. It explains how Azure Database for PostgreSQL enables users to deploy, manage, and scale PostgreSQL databases in the cloud efficiently. The service supports various PostgreSQL versions and offers features such as automated backups, high availability, and security enhancements to ensure data protection and reliability. The article emphasizes the flexibility and scalability of Azure Database for PostgreSQL, making it suitable for a wide range of applications, from small projects to enterprise-level deployments.

Q16. Summarize the article on "Azure Database for MySQL - Flexible Server".

Answer :

The article discusses Azure Database for MySQL - Flexible Server, a fully managed database service provided by Microsoft Azure. It highlights the key features and benefits of this service, emphasizing its flexibility, scalability, and cost-effectiveness. Azure Database for MySQL - Flexible Server allows users to deploy MySQL databases in the cloud with minimal administrative overhead. It offers automated backups, high availability, and security features to ensure data integrity and reliability. The article also provides guidance on getting started with Azure Database for MySQL - Flexible Server and offers resources for further exploration, making it suitable for a wide range of applications, from small projects to enterprise-level deployments.

Q17. Summarize the article on "Azure Database for MySQL Single Server".

Answer :

The article provides an overview of Azure Database for MySQL Single Server, a fully managed database service offered by Microsoft Azure. It outlines the features and benefits of this service, highlighting its simplicity, reliability, and scalability. Azure Database for MySQL Single Server is designed to meet the needs of various applications, from small projects to large-scale enterprise deployments. It offers features such as automated backups, high availability, and security enhancements to ensure data protection and reliability. The article also provides guidance on getting started with the service and offers resources for further exploration.