



Desenvolvimento FullStack

Missão Prática Nível 4 - Mundo 4

Darlyson Cavalcanti Nery de Souza

Informações

Curso

Curso: Desenvolvimento Full Stack

Universidade: Estácio de Sá - Campus São Lourenço da Mata

Tutor: Maria Manso

Período: 4º

Turma: 9001

Matéria: Tirando Proveito da Nuvem Para Projetos de Software

Aluno

Nome: Darlyson Cavalcanti Nery de Souza

Matrícula: 202301453471

GitHub: <https://github.com/DarlysonCavalcanti/DesenvolvimentoFullStack-mundo-4-missao-pratica-4>

1. Missão Prática: “Tirando proveito da nuvem para projetos de software”

2. Objetivos da Prática

- Demonstrar habilidade na criação e gerenciamento de recursos na Nuvem Azure, adquirindo conhecimento sobre a estrutura básica da plataforma Azure
- Utilizar efetivamente o portal Azure para criar e configurar uma Máquina Virtual (VM), demonstrando compreensão dos recursos e suas funções.
- Configurar regras de rede e grupos de segurança, adquirindo conhecimento sobre a estrutura das regras de rede na Nuvem Azure.
- Importar um arquivo .bacpac para um banco de dados no Banco de Dados SQL do Azure;
- Criar e configurar um aplicativo web no Azure, demonstrando compreensão do mecanismo de hospedagem e implantação de aplicativos web.

3. Configuração da máquina virtual e do banco de dados Azure

Imagem 1. Criando máquina virtual na nuvem Azure.

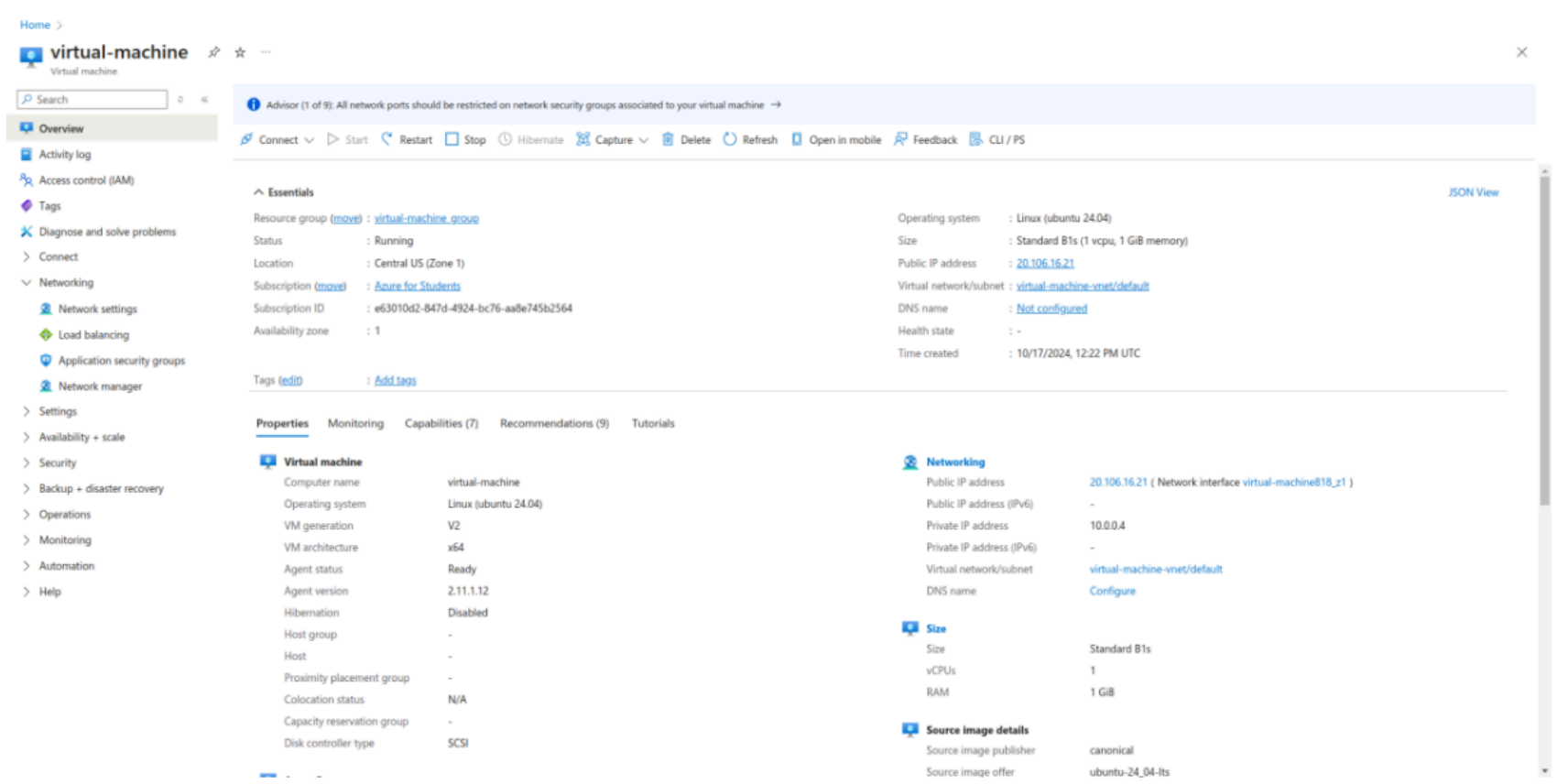
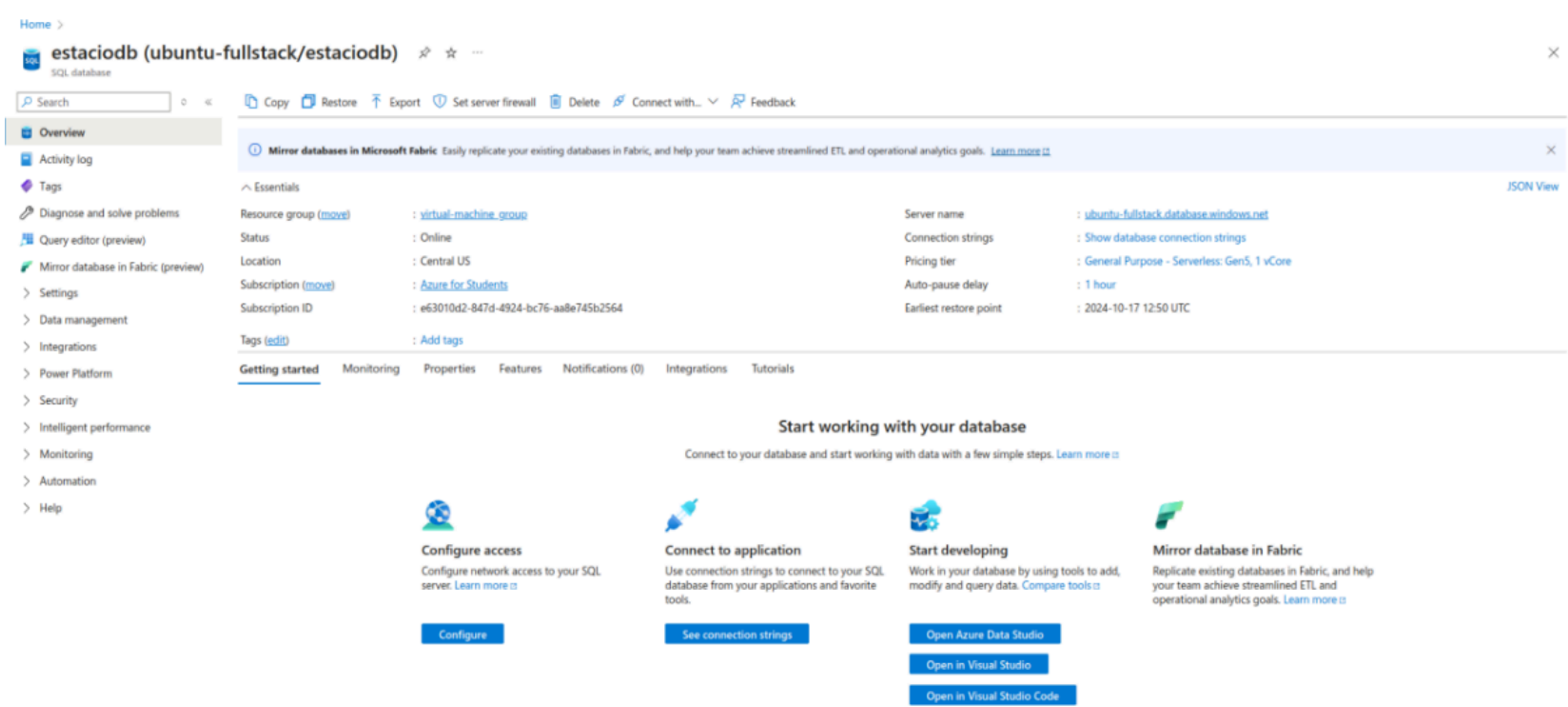


Imagem 2. Criando banco de dados SQL na nuvem Azure.



4. Script do banco de dados

Comandos SQL:

CREATE DATABASE [LogiMoveDB] (EDITION = 'GeneralPurpose', SERVICE_OBJECTIVE = 'GP_Gen5_2', MAXSIZE = 32 GB) WITH CATALOG_COLLATION = SQL_Latin1_General_CP1_CI_AS, LEDGER = OFF;

GO

CREATE TABLE [Drivers]([DriverID] [int] NOT NULL, [Nome] [varchar](100) NULL, [CNH] [varchar](20) NULL, [Endereço] [varchar](200) NULL, [Contato] [varchar](50) NULL, PRIMARY KEY CLUSTERED ([DriverID] ASC) WITH (STATISTICS_NORECOMPUTE = OFF, IGNORE_DUP_KEY = OFF, OPTIMIZE_FOR_SEQUENTIAL_KEY = OFF) ON [PRIMARY]) ON [PRIMARY]

GO

CREATE TABLE [DriverQualifications]([QualificationID] [int] NOT NULL, [DriverID] [int] NULL, [Qualificação] [varchar](100) NULL, [DataObtenção] [date] NULL, [Validade] [date] NULL, PRIMARY KEY CLUSTERED ([QualificationID] ASC) WITH (STATISTICS_NORECOMPUTE = OFF, IGNORE_DUP_KEY = OFF, OPTIMIZE_FOR_SEQUENTIAL_KEY = OFF) ON [PRIMARY]) ON [PRIMARY]

GO

ALTER TABLE [DriverQualifications] WITH CHECK ADD FOREIGN KEY([DriverID]) REFERENCES [Drivers] ([DriverID])

GO

CREATE TABLE [DriverTravelHistory]([TravelID] [int] NOT NULL, [DriverID] [int] NULL, [DataViagem] [date] NULL, [Origem] [varchar](200) NULL, [Destino] [varchar](200) NULL, [DistanciaPercorrida] [float] NULL, PRIMARY KEY CLUSTERED ([TravelID] ASC) WITH (STATISTICS_NORECOMPUTE = OFF, IGNORE_DUP_KEY = OFF, OPTIMIZE_FOR_SEQUENTIAL_KEY = OFF) ON [PRIMARY]) ON [PRIMARY]

GO

ALTER TABLE [DriverTravelHistory] WITH CHECK ADD FOREIGN KEY([DriverID]) REFERENCES [Drivers] ([DriverID])

GO

CREATE TABLE [Clients]([ClientID] [int] NOT NULL, [Nome] [varchar](100) NULL, [Empresa] [varchar](100) NULL, [Endereço] [varchar](200) NULL, [Contato] [varchar](50) NULL, PRIMARY KEY CLUSTERED ([ClientID] ASC) WITH (STATISTICS_NORECOMPUTE = OFF, IGNORE_DUP_KEY = OFF, OPTIMIZE_FOR_SEQUENTIAL_KEY = OFF) ON [PRIMARY]) ON [PRIMARY]

GO

CREATE TABLE [ClientPreferences]([PreferenceID] [int] NOT NULL, [ClientID] [int] NULL, [Preferencia] [text] NULL, PRIMARY KEY CLUSTERED ([PreferenceID] ASC) WITH (STATISTICS_NORECOMPUTE = OFF, IGNORE_DUP_KEY = OFF, OPTIMIZE_FOR_SEQUENTIAL_KEY = OFF) ON [PRIMARY]) ON [PRIMARY] TEXTIMAGE_ON [PRIMARY]

GO

ALTER TABLE [ClientPreferences] WITH CHECK ADD FOREIGN KEY([ClientID]) REFERENCES [Clients] ([ClientID])

GO

CREATE TABLE [Orders]([OrderID] [int] NOT NULL, [ClientID] [int] NULL,[DriverID] [int] NULL, [DetalhesPedido] [text] NULL, [DataEntrega] [date] NULL, [Status] [varchar](50) NULL, PRIMARY KEY CLUSTERED ([OrderID] ASC) WITH (STATISTICS_NORECOMPUTE = OFF, IGNORE_DUP_KEY = OFF, OPTIMIZE_FOR_SEQUENTIAL_KEY = OFF) ON [PRIMARY]) ON [PRIMARY] TEXTIMAGE_ON [PRIMARY]
GO

ALTER TABLE [Orders] WITH CHECK ADD FOREIGN KEY([ClientID]) REFERENCES [Clients] ([ClientID])
GO

ALTER TABLE [Orders] WITH CHECK ADD FOREIGN KEY([DriverID]) REFERENCES [Drivers] ([DriverID])
GO

CREATE TABLE [OrderStatusHistory]([StatusHistoryID] [int] NOT NULL, [OrderID] [int] NULL, [StatusAnterior] [varchar](50) NULL, [StatusAtual] [varchar](50) NULL, [DataMudanca] [date] NULL, PRIMARY KEY CLUSTERED ([StatusHistoryID] ASC) WITH (STATISTICS_NORECOMPUTE = OFF, IGNORE_DUP_KEY = OFF, OPTIMIZE_FOR_SEQUENTIAL_KEY = OFF) ON [PRIMARY]) ON [PRIMARY]
GO

ALTER TABLE [OrderStatusHistory] WITH CHECK ADD FOREIGN KEY([OrderID]) REFERENCES [Orders] ([OrderID])
GO

CREATE TABLE [ClientOrderHistory] ([HistoryID] [int] NOT NULL, [OrderID] [int] NULL, [ClientID] [int] NULL, [DataPedido] [date] NULL, [ResumoPedido] [text] NULL, PRIMARY KEY CLUSTERED ([HistoryID] ASC) WITH (STATISTICS_NORECOMPUTE = OFF, IGNORE_DUP_KEY = OFF, OPTIMIZE_FOR_SEQUENTIAL_KEY = OFF) ON [PRIMARY]) ON [PRIMARY] TEXTIMAGE_ON [PRIMARY]
GO

ALTER TABLE [ClientOrderHistory] WITH CHECK ADD FOREIGN KEY([ClientID]) REFERENCES [Clients] ([ClientID])
GO

ALTER TABLE [ClientOrderHistory] WITH CHECK ADD FOREIGN KEY([OrderID]) REFERENCES [Orders] ([OrderID])
GO

-- Inserção de motoristas

INSERT INTO Drivers (DriverID, Nome, CNH, Endereço, Contato) VALUES (1, 'João Silva', '123456789', 'Rua das Flores, 100', '11987654321'), (2, 'Maria Oliveira', '987654321', 'Avenida do Sol, 200', '11976543210');
GO

-- Inserção das qualificações dos motoristas

INSERT INTO DriverQualifications (QualificationID, DriverID, Qualificação, DataObtenção, Validade) VALUES (1, 1, 'Transporte de Cargas Perigosas', '2023-01-10', '2025-01-10'), (2, 2, 'Transporte Internacional', '2023-02-15', '2025-02-15');
GO

-- Inserção do histórico de viagens INSERT INTO DriverTravelHistory (TravelID, DriverID, DataViagem, Origem, Destino, DistanciaPercorrida) VALUES (1, 1, '2024-04-01', 'São Paulo', 'Rio de Janeiro', 430.5), (2, 2, '2024-04-02', 'Curitiba', 'Porto Alegre', 711.0);

GO

-- Inserção de clientes

INSERT INTO Clients (ClientID, Nome, Empresa, Endereço, Contato) VALUES (1, 'Empresa A', 'A Ltda', 'Rua A, 50', '11333445566'), (2, 'Empresa B', 'B S.A.', 'Avenida B, 100', '11222444666');

GO

-- Inserção de preferências dos clientes

INSERT INTO ClientPreferences (PreferenceID, ClientID, Preferencia) VALUES (1, 1, 'Preferência por transportes rápidos e seguros'), (2, 2, 'Preferência por custo baixo');

GO

-- Inserção de pedidos

INSERT INTO Orders (OrderID, ClientID, DriverID, DetalhesPedido, DataEntrega, Status) VALUES (1, 1, 1, '50 caixas de material inflamável', '2024-04-05', 'Entregue'), (2, 2, 2, '100 unidades de eletrônicos', '2024-04-06', 'Em trânsito');

GO

-- Inserção de histórico de status dos pedidos

INSERT INTO OrderStatusHistory (StatusHistoryID, OrderID, StatusAnterior, StatusAtual, DataMudança) VALUES (1, 1, 'Em preparação', 'Entregue', '2024-04-05'), (2, 2, 'Aguardando coleta', 'Em trânsito', '2024-04-03');

GO

-- Inserção de histórico de pedidos dos clientes

INSERT INTO ClientOrderHistory (HistoryID, OrderID, ClientID, DataPedido, ResumoPedido) VALUES (1, 1, 1, '2024-04-01', 'Pedido de 50 caixas de material inflamável'), (2, 2, 2, '2024-04-02', 'Pedido de 100 unidades de eletrônicos');

GO

-- Consulta para encontrar todos os motoristas com suas qualificações

SELECT d.Nome, dq.Qualificação, dq.DataObtenção, dq.Validade FROM Drivers d JOIN DriverQualifications dq ON d.DriverID = dq.DriverID;

GO

-- Consulta para listar todos os pedidos e seus status atuais

SELECT o.OrderID, c.Nome as Cliente, d.Nome as Motorista, o.DetalhesPedido, o.Status FROM Orders o JOIN Clients c ON o.ClientID = c.ClientID JOIN Drivers d ON o.DriverID = d.DriverID;

GO

-- Consulta para encontrar viagens realizadas em determinado período SELECT dt.DataViagem, dt.Origem, dt.Destino, dt.DistanciaPercorrida FROM DriverTravelHistory dt WHERE dt.DataViagem BETWEEN '2024-04-01' AND '2024-04-30';

GO

