Name Qin Du, Ziyu Lin, Lu Huang Date: 2024/12/05

(last name, first name)

Course Section Number: CSCI-GA.2433-001

**Project 3 Part III**

**Total in points** (100 points total): \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**Professor’s Comments:**

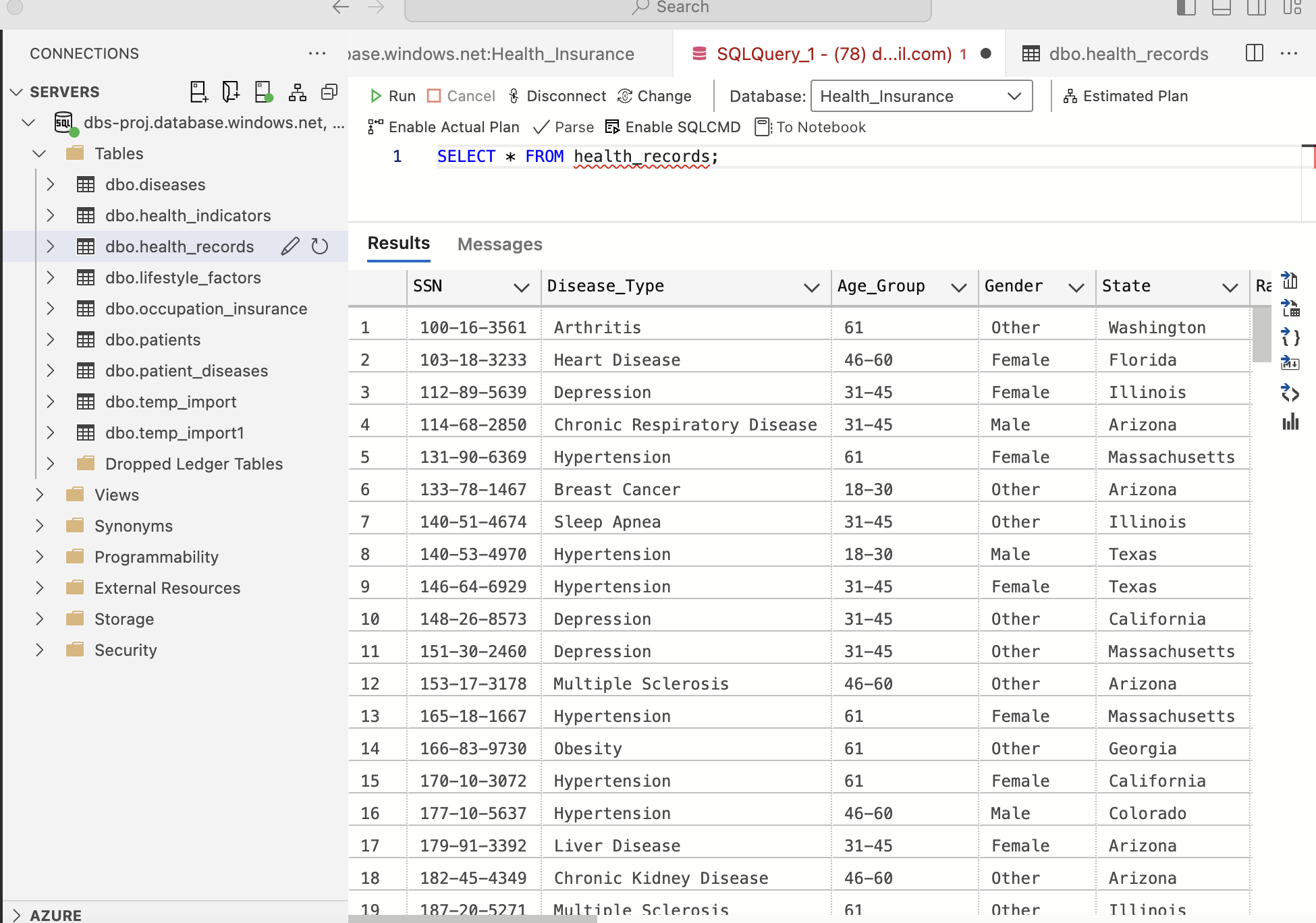
**Affirmation of my Independent Effort**  Qin Du, Ziyu Lin, Lu Huang

**(Sign here)**

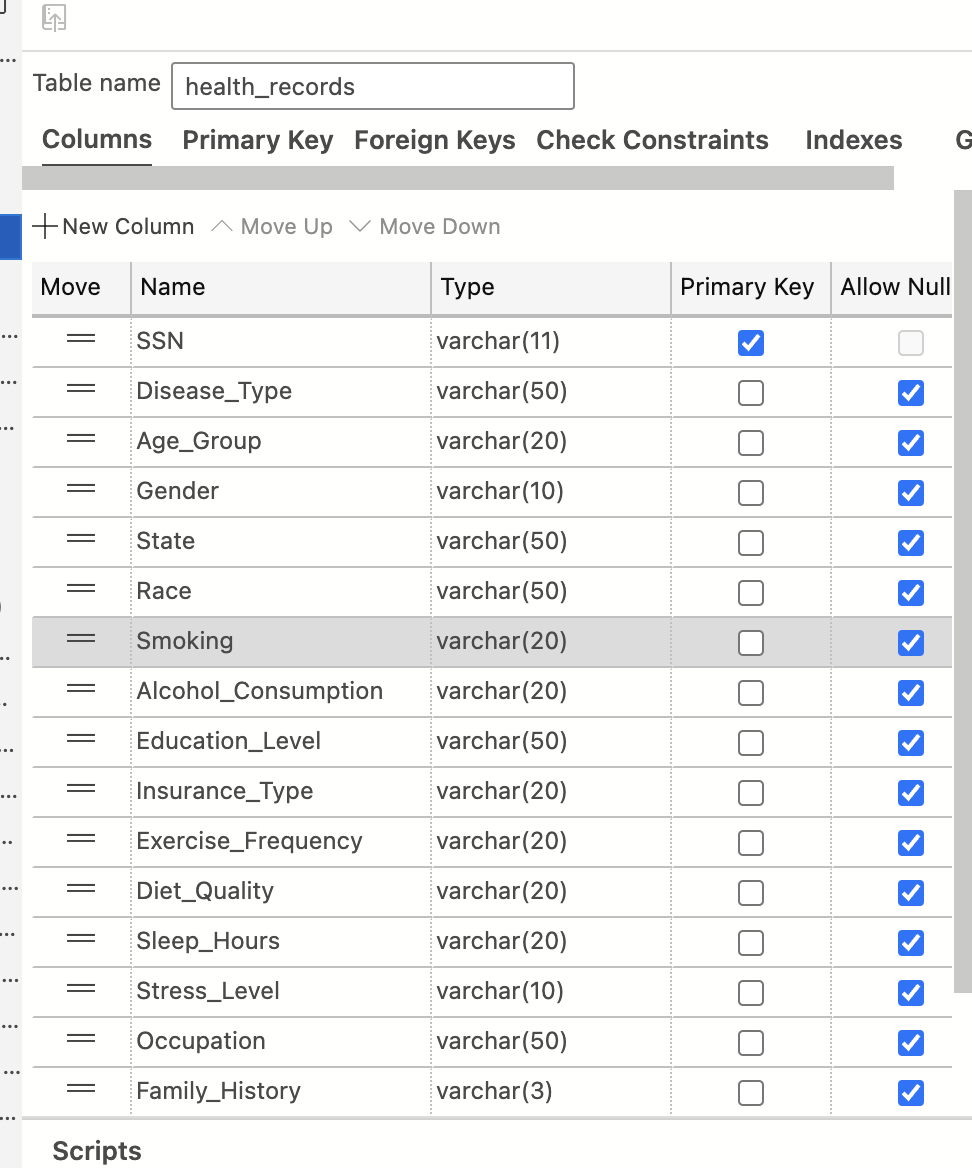
**1.** **Create a physical database design.**

**1.1 Create a database table.**

In this part, we use Azure Data Studio to create our database table according to our logical scheme. We can get our table in here



We can get our detailed data structure in here:



**1.2. Design Reasons.**

**1.2.1.** **Indexing**

Indexes can speed up the retrieval of rows from a table and are especially useful for frequently searched, sorted, or joined columns. We use the following SQL commands to create indexes:

*We create indexes to optimize the most frequently accessed columns in our medical database:*

*-- Patient SSN is frequently used for joins and lookups*

*CREATE INDEX idx\_patient\_ssn ON Patient (SSN);*

*-- Disease lookups and joins are common*

*CREATE INDEX idx\_disease\_name ON Disease (Disease\_Name);*

*-- Common demographic queries for analysis*

*CREATE INDEX idx\_patient\_age\_group ON Patient (Age\_Group);*

*CREATE INDEX idx\_patient\_state ON Patient (State);*

*CREATE INDEX idx\_patient\_income ON Patient\_Demographics (Income\_Level);*

*-- Health status queries for risk assessment*

*CREATE INDEX idx\_health\_bmi ON Health\_Status (BMI\_Category);*

*CREATE INDEX idx\_lifestyle\_smoking ON Lifestyle (Smoking\_Status);*

**1.2.2.** **Partitioning**

Partitioning can improve the management and query performance of large tables, particularly when a table contains a significant amount of data. To create partitions, use the following SQL command:

*ALTER TABLE Patient PARTITION BY LIST (State) (*

*PARTITION p\_northeast VALUES IN ('New York', 'Massachusetts'),*

*PARTITION p\_southeast VALUES IN ('Florida', 'Georgia', 'North Carolina'),*

*PARTITION p\_midwest VALUES IN ('Illinois', 'Michigan', 'Ohio'),*

*PARTITION p\_west VALUES IN ('California', 'Washington', 'Arizona', 'Colorado'),*

*PARTITION p\_other VALUES IN (DEFAULT)*

*);*

*-- Partition patient disease records by year*

*ALTER TABLE Patient\_Disease PARTITION BY RANGE (YEAR(Diagnosis\_Date)) (*

*PARTITION p2022 VALUES LESS THAN (2023),*

*PARTITION p2023 VALUES LESS THAN (2024),*

*PARTITION p2024 VALUES LESS THAN (2025),*

*PARTITION p\_future VALUES LESS THAN MAXVALUE*

*);*

**1.2.3.** **Selective Materialization**

We create materialized views for commonly accessed aggregate data:

-- Disease prevalence by demographic groups

CREATE TABLE Disease\_Demographics\_Summary AS

SELECT

d.Disease\_Name,

p.Age\_Group,

p.Gender,

COUNT(\*) as Patient\_Count,

COUNT(\*) \* 100.0 / (SELECT COUNT(\*) FROM Patient) as Prevalence\_Percentage

FROM Patient p

JOIN Patient\_Disease pd ON p.SSN = pd.SSN

JOIN Disease d ON pd.Disease\_ID = d.Disease\_ID

GROUP BY d.Disease\_Name, p.Age\_Group, p.Gender;

-- Lifestyle impact summary

CREATE TABLE Lifestyle\_Health\_Summary AS

SELECT

l.Exercise\_Frequency,

l.Diet\_Quality,

l.Smoking\_Status,

COUNT(DISTINCT pd.Disease\_ID) as Avg\_Disease\_Count,

AVG(CASE WHEN h.BMI\_Category = 'Obese' THEN 1 ELSE 0 END) as Obesity\_Rate

FROM Lifestyle l

JOIN Patient p ON l.SSN = p.SSN

JOIN Health\_Status h ON p.SSN = h.SSN

LEFT JOIN Patient\_Disease pd ON p.SSN = pd.SSN

GROUP BY l.Exercise\_Frequency, l.Diet\_Quality, l.Smoking\_Status;

-- Regional health patterns

CREATE TABLE Regional\_Health\_Summary AS

SELECT

p.State,

d.Disease\_Name,

COUNT(\*) as Case\_Count,

pd.Insurance\_Type,

AVG(CASE WHEN h.Stress\_Level = 'High' THEN 1 ELSE 0 END) as High\_Stress\_Rate

FROM Patient p

JOIN Patient\_Demographics pd ON p.SSN = pd.SSN

JOIN Patient\_Disease pdi ON p.SSN = pdi.SSN

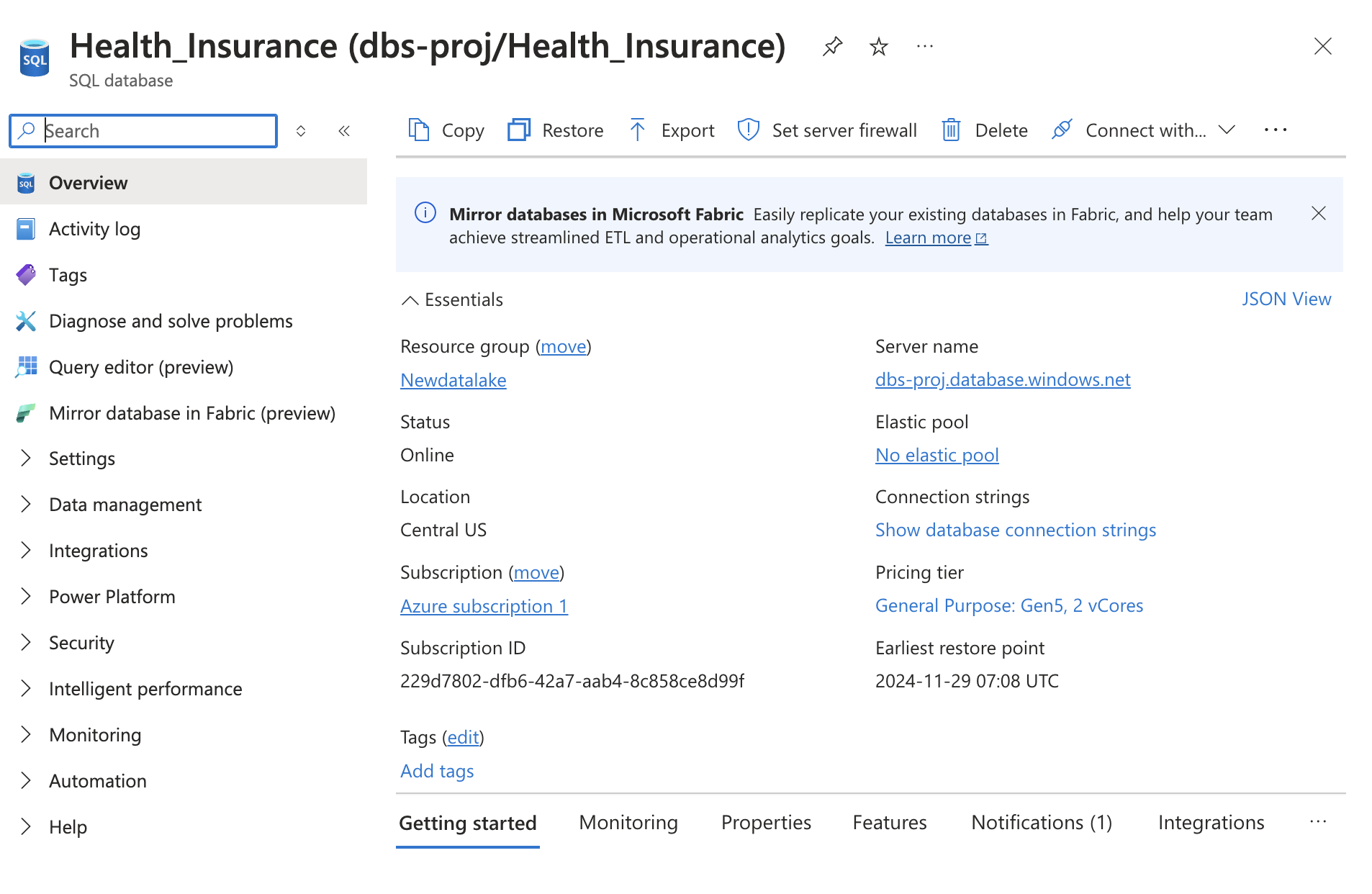
JOIN Disease d ON pdi.Disease\_ID = d.Disease\_ID

JOIN Health\_Status h ON p.SSN = h.SSN

GROUP BY p.State, d.Disease\_Name, pd.Insurance\_Type;

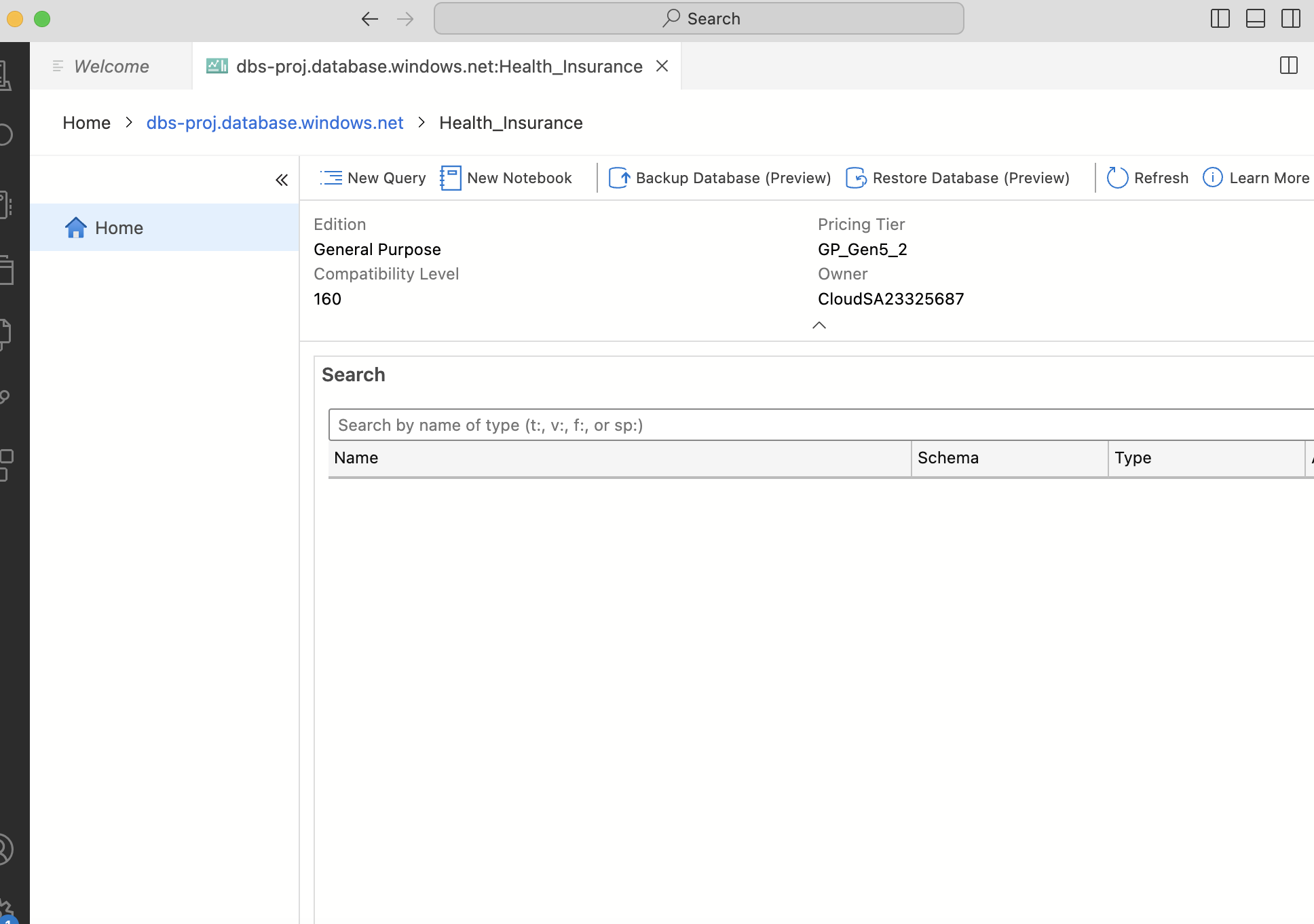
**2.** **Deploy design on the Microsoft Azure**

**2.1. Create a server for mysql in Microsoft Azure.**



**2.2. Connect to** Azure Data Studio**.**

First, we open the Azure Data Studio. Then we create a new connection like the following:



**2.3. Operate in Azure Data Studio.**

We create a database health.insuranceDB, add related data from the data we stored at blob service. Then we design our database like before to create indexing, partitioning, and Selective Materialization.

**3.** **Short, medium, and long-term benefits**

**Short-Term Benefits**

Rapid Risk Assessment: Through basic data analysis and simple machine learning models, insurance companies can quickly identify high-risk customer groups, allowing for more effective risk management and pricing strategies.

Cost Efficiency: In the short term, Big Data analytics can help identify and reduce unnecessary expenditures, such as avoiding fraud or unnecessary claims.

Enhanced Customer Satisfaction: By better understanding customer needs and risks, insurance companies can offer more personalized services, enhancing customer satisfaction and loyalty.

**Medium-Term Benefits**

Product and Service Innovation: Through in-depth machine learning, insurance companies can develop new insurance products and services that better meet market needs and customer expectations.

Refined Management: Using more complex data models, insurance companies can perform more precise risk assessments and pricing, optimizing resource allocation.

Enhanced Market Competitiveness: Through continuous data analysis and application, insurance companies can better understand market dynamics, gaining an advantageous position in the competition.

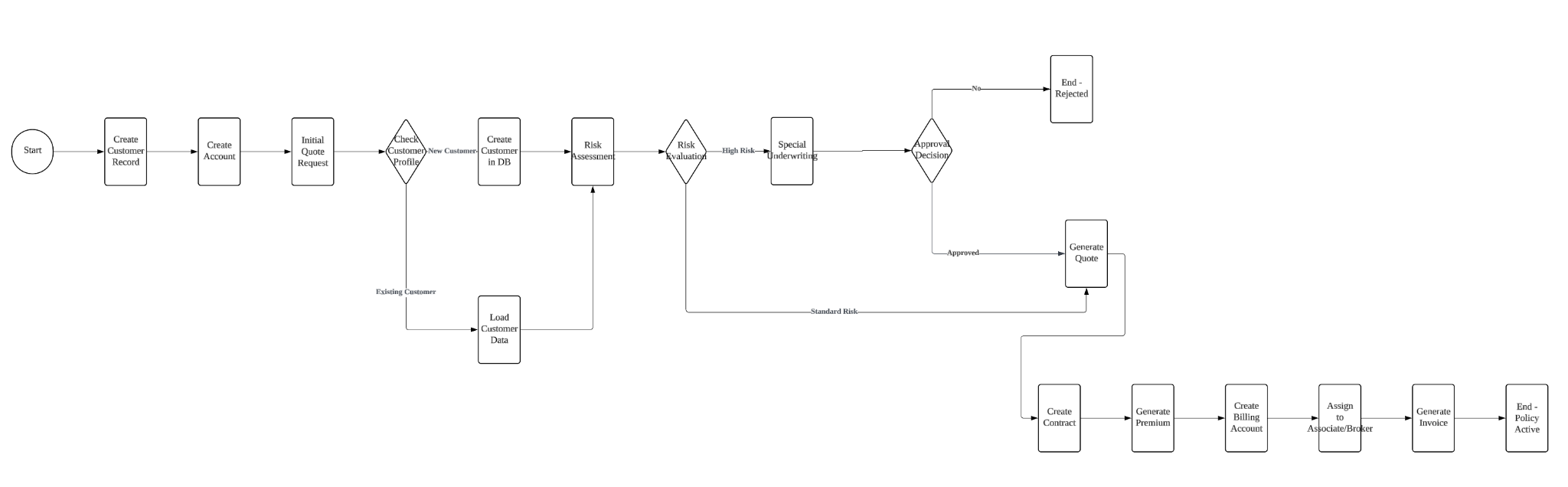
**Long-Term Benefits**

Continuous Learning and Adaptation: Long-term investment in advanced data analytics and artificial intelligence technologies will enable insurance companies to continuously learn and adapt to market changes, improving decision-making accuracy and efficiency.

Sustained Customer Relationships: Through ongoing customer data analysis, insurance companies can better predict customer needs, offer personalized services, and build long-term customer relationships.

Innovative Leadership: Long-term investment in Big Data and AI will help insurance companies maintain a leading position in technological innovation, setting industry trends.

**4.**  **Insurance Quote and Policy Application Workflow**

****

1. Initial Customer Interaction

When a customer begins the insurance application process, the first step is creating a customer record in the system. This involves collecting basic personal information which is stored in the Customer table. If the customer has family members to be insured, this information is also collected and stored in the Customer\_Family table.

2. Account Creation

Once the customer record is established, an Account is created in the system. This account serves as the central point for managing all customer interactions. Related information is stored across multiple tables:

* AccountAlias for alternative contact information
* AccountMember for additional account users
* BillingAccount for payment-related information

3. Risk Assessment Phase

The system then begins the risk evaluation process:

* For new customers, a fresh risk profile is created
* For existing customers, their historical data is retrieved and analyzed
* The assessment considers various factors stored in the database
* High-risk cases are flagged for special underwriting review

4. Quote Generation

Based on the risk assessment:

* For standard risk cases, the system automatically generates a quote using predefined rules
* For high-risk cases, the application is routed to underwriting for manual review
* The underwriting team can either approve with modified terms or reject the application
* Approved quotes are stored in the Contract and ContractPremium tables

5. Contract Creation

Once a quote is accepted:

* A formal contract is generated with a unique Contract\_number
* Contract benefits are documented in ContractBenefit
* Premium information is recorded in ContractPremium and PremiumMgmtContract
* The contract is assigned to a managing entity through ManagerContract

6. Agent/Broker Assignment

The application is then assigned to appropriate insurance professionals:

* An Associate or Broker is assigned to manage the policy
* The assignment is recorded in Broker\_Associate
* Writing numbers are generated and stored in WritingNumbers
* The Manage table tracks the relationship between contracts and writing numbers

7. Billing and Invoice Generation

The final phase involves setting up billing:

* A billing account is confirmed or created
* Initial invoices are generated and stored in the Invoice table
* Payment due dates and terms are established
* The policy becomes active upon successful payment processing

**5.** **Machine learning for Health Risk Prediction in Insurance Industry**

In this part, we developed a machine learning pipeline to predict health risks for an insurance company. The model classifies individuals into Low Risk, Medium Risk, or High Risk categories based on their demographic, behavioral, and health-related features. The ultimate goal is to optimize insurance premium pricing, enhance customer segmentation, and implement proactive health management strategies.

**5.1 Data and Preprocessing**

The dataset includes 5,000 samples with the following attributes:

1. Demographic Features: Age\_Group, Gender, Race.

2. Behavioral and Lifestyle Factors: Smoking, Alcohol Consumption, Exercise Frequency, Diet Quality, Sleep Hours.

3. Health History: Stress Level, Family History, BMI Category

4. Insurance Details: Insurance Type, Income Level.

5. Target Variable: Health Risk (derived from Disease\_Type).

手机屏幕截图

描述已自动生成

Preprocessing steps:

1. Categorical Encoding: Transformed categorical features into numerical values using LabelEncoder.

2. Target Transformation: Mapped Disease\_Type to risk levels:

- Low Risk: Arthritis, Depression.

- Medium Risk: Obesity, Hypertension, Sleep Apnea, Diabetes.

- High Risk: Heart Disease, Lung Cancer, Chronic Respiratory Disease, Colorectal Cancer.

3. Handling Class Imbalance: Used class\_weight='balanced' to address the unequal distribution of risk levels.

表格

描述已自动生成

**5.3** **Exploratory Data Analysis (EDA)**

Risk Distribution:

* Medium Risk accounted for the majority of samples, followed by High Risk and Low Risk.

Key Insights:

* Diet Quality and Stress Level were the most significant predictors of health risk, indicating the impact of lifestyle and psychological factors.
* High Risk customers were often associated with poor diet and high stress, whereas Low Risk customers exhibited healthier behaviors.

图表, 条形图, 直方图

描述已自动生成

图表, 条形图

描述已自动生成

图表, 条形图

描述已自动生成

图表, 条形图

描述已自动生成

**Model Selection**

1. Logistic Regression:
   * Accuracy: 45.9%.
   * Served as a baseline model to evaluate the linear separability of the data.
2. Random Forest:
   * Accuracy: 62.4%.
   * Performed better due to its ability to capture nonlinear relationships and handle class imbalance effectively.

**Model Selection Justification:**

* **Logistic Regression:** Chosen for its simplicity and ease of interpretation, Logistic Regression established a baseline for model performance. Its results highlight the limitations of linear approaches for the given dataset.
* **Random Forest:** Selected for its strength in handling complex feature interactions and nonlinear patterns, Random Forest's superior performance demonstrates its suitability for the dataset. Moreover, its ability to identify key features is instrumental in deriving actionable insights, making it particularly valuable for business applications.

**5.3 Insights & Conclusions**

**Feature Importance**:

The top contributing features were:

1． Diet Quality: Directly impacts the likelihood of obesity, diabetes, and heart-related issues.

2． Stress Level: A key factor for conditions like hypertension and heart disease.

3． Age Group: Higher age groups were associated with chronic diseases and higher risk levels.

图表, 漏斗图

描述已自动生成

**Business Implications:**

Diet Quality: Insurance companies can design wellness programs targeting diet improvement to reduce risk levels.

Stress Level: Offering stress management resources can help mitigate high-risk conditions like hypertension.

**Evaluation Performance:**

**Random Forest Results**The Random Forest model was evaluated based on its precision, recall, and F1-score across three risk categories: High Risk, Medium Risk, and Low Risk. Below is a detailed breakdown of its performance:

* **High Risk:**
  + **Precision:** 0.60
  + **Recall:** 0.46
  + **F1-Score:** 0.52  
    The model struggled to accurately classify High Risk instances, as indicated by the relatively low recall. This suggests that many High Risk cases were misclassified as Medium Risk, leading to a modest F1-score.
* **Medium Risk:**
  + **Precision:** 0.63
  + **Recall:** 0.75
  + **F1-Score:** 0.68  
    The model performed reasonably well in identifying Medium Risk cases, with a balanced precision and recall. However, there is room for improvement in reducing false negatives, where Medium Risk instances might be misclassified as Low Risk or High Risk.
* **Low Risk:**
  + **Precision:** 1.00
  + **Recall:** 0.87
  + **F1-Score:** 0.93  
    The model excelled in classifying Low Risk instances, achieving perfect precision. This indicates that all cases predicted as Low Risk were indeed correct. However, the slightly lower recall suggests a small portion of Low Risk cases were incorrectly classified as Medium Risk.

**Overall Insights:** The model was highly effective in distinguishing between Medium and Low Risk categories, achieving strong performance metrics in these groups. However, it faced challenges differentiating High Risk cases from Medium Risk ones. The confusion between High and Medium Risk categories indicates potential overlap in their feature distributions or limitations in the model's ability to capture subtle distinctions between these groups.

**Applications in Insurance:**

1. Dynamic Premium Pricing:

- High Risk customers pay higher premiums to offset potential claims.

- Low Risk customers receive discounted premiums to encourage healthier behaviors.

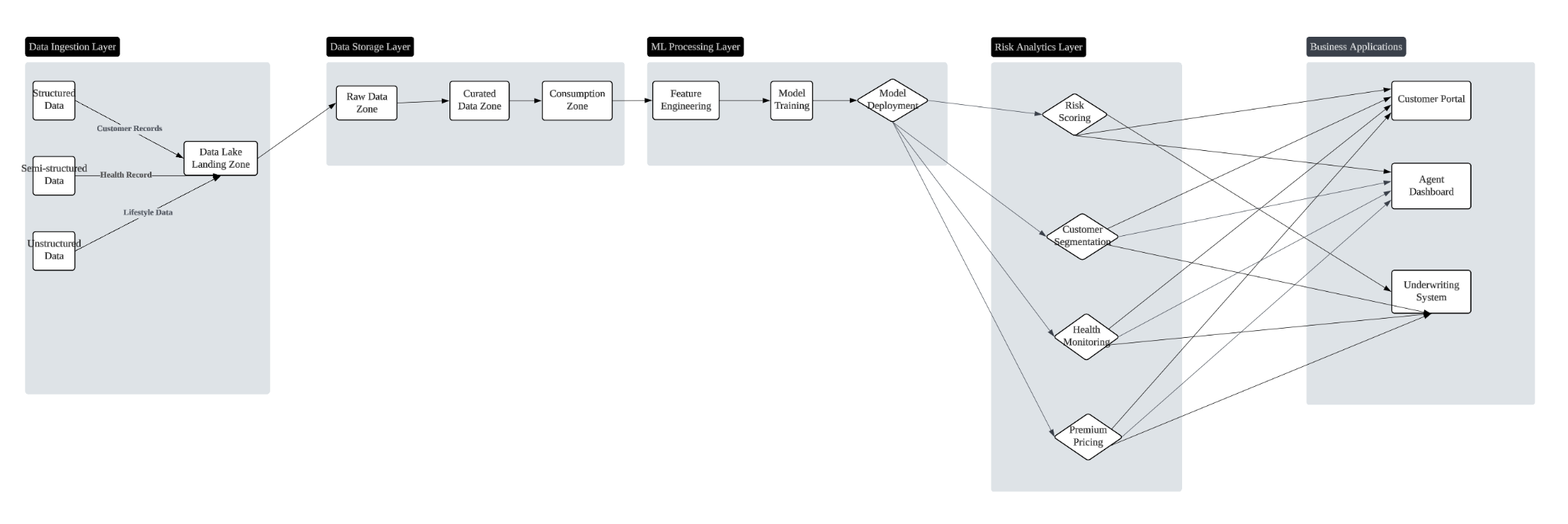
2. Proactive Health Management:

- High Risk customers are enrolled in health monitoring and intervention programs.

3. Customer Segmentation:

- Segregation of customers into risk tiers for targeted marketing and resource allocation.

**6. Fit-In Architecture based on ML**

****

Based on our machine learning model results and the company's data management needs, we have designed a comprehensive reference architecture that enables the insurance company to effectively leverage hybrid data. This architecture supports both traditional insurance operations and advanced analytics, with a particular focus on health risk prediction and dynamic premium pricing. Our proposed reference architecture consists of five primary layers, each designed to handle specific aspects of data processing and business operations.

1. Data Ingestion Layer

The first layer handles multiple types of data sources. Structured data includes customer demographics, policy information, and claims history stored in traditional databases. Semi-structured data encompasses health records and medical reports in various formats. Unstructured data sources include lifestyle information, customer interactions, and health consultation records.

1. Data Storage Layer

The storage layer implements a modern data lake architecture with three distinct zones. The Raw Data Zone stores original data in its native format. The Curated Data Zone contains cleaned and standardized data ready for analysis. The Consumption Zone hosts prepared datasets specifically structured for business applications and machine learning models.

1. ML Processing Layer

This layer is dedicated to machine learning operations, handling the entire pipeline from feature engineering to model deployment. Our current health risk prediction model, with its 62.4% accuracy using Random Forest classification, is hosted here. The layer processes key features such as diet quality, stress levels, and age groups, which have proven to be the most significant predictors of health risks.

1. Risk Analytics Layer

Building on the ML Processing Layer, the Risk Analytics Layer implements four key functionalities:

* 1. Risk Scoring: Evaluates customer health risks using the trained models
* 2. Premium Pricing: Calculates customized premiums based on risk assessments
* 3. Health Monitoring: Tracks customer health metrics over time
* 4. Customer Segmentation: Groups customers based on risk profiles and behaviors

1. Business Applications Layer

The topmost layer consists of user-facing applications that leverage the underlying analytics:

* Underwriting System: Automates risk assessment and policy pricing
* Customer Portal: Provides personalized health recommendations and policy information
* Agent Dashboard: Enables insurance agents to make data-driven decisions

This reference architecture provides a robust foundation for the insurance company's data operations. It effectively combines traditional insurance processes with modern machine learning capabilities, enabling better risk assessment and more personalized customer service. The architecture's modular design ensures it can evolve with the company's needs while maintaining operational efficiency and data security.

**7. Leveraging Azure Cloud Services for Analytics in Insurance**

Our implementation of big data analytics leverages Microsoft Azure's comprehensive cloud services to process and analyze insurance and health data. The solution begins with Azure Data Factory handling data ingestion from multiple sources, including customer demographics from Azure SQL Database, health records via Blob Storage, real-time health monitoring through Event Hubs, and policy data in Data Lake Storage.

For large-scale data processing, Azure Databricks serves as our primary computation engine, running Spark clusters that handle health risk assessments and machine learning pipelines. The implementation utilizes Delta Lake to ensure ACID compliance in data transactions, while Azure Synapse Analytics powers complex queries across insurance datasets and creates specialized data marts for business needs.

The machine learning component operates through Azure Machine Learning service, which hosts our Random Forest model for risk classification. The service manages the entire ML lifecycle, from model training to deployment and monitoring. For real-time analytics, we deployed Azure Stream Analytics to process IoT health device data, enabling immediate risk metric monitoring and automated alerts for high-risk customers.

Visualization and reporting capabilities are implemented through Power BI, creating interactive dashboards that display risk analysis, customer segmentation, and premium pricing trends. These dashboards serve as the primary interface for business users to interact with the processed data, offering both high-level overviews and detailed drill-down capabilities for specific metrics. To maintain optimal performance and security, the implementation includes query optimization techniques and uses Azure Cache for Redis for frequently accessed data.

**Appendix:**

CREATE DATABASE InsuranceDB;

USE InsuranceDB;

CREATE TABLE Customer (

Ssn CHAR(11) NOT NULL,

FirstName VARCHAR(100),

LastName VARCHAR(100),

DOB DATE,

Address VARCHAR(255),

PRIMARY KEY (Ssn)

);

CREATE TABLE Customer\_Family (

Member1\_Ssn CHAR(11) NOT NULL,

Member2\_Ssn CHAR(11) NOT NULL,

FOREIGN KEY (Member1\_Ssn) REFERENCES Customer(Ssn),

FOREIGN KEY (Member2\_Ssn) REFERENCES Customer(Ssn)

);

CREATE TABLE Company (

Code INT NOT NULL AUTO\_INCREMENT,

Name VARCHAR(100),

Address VARCHAR(255),

Zip VARCHAR(20),

PRIMARY KEY (Code)

);

CREATE TABLE Account (

Number CHAR(20) NOT NULL,

Name VARCHAR(100),

Company\_Code INT,

PRIMARY KEY (Number),

FOREIGN KEY (Company\_Code) REFERENCES Company(Code)

);

CREATE TABLE AccountAlias (

ID INT NOT NULL AUTO\_INCREMENT,

Alias\_Name VARCHAR(100),

Alias\_Address VARCHAR(255),

Alias\_Desc TEXT,

Account\_Number CHAR(20),

PRIMARY KEY (ID),

FOREIGN KEY (Account\_Number) REFERENCES Account(Number)

);

CREATE TABLE AccountMember (

ID INT NOT NULL AUTO\_INCREMENT,

Name VARCHAR(100),

Phone VARCHAR(20),

Account\_Number CHAR(20),

PRIMARY KEY (ID),

FOREIGN KEY (Account\_Number) REFERENCES Account(Number)

);

CREATE TABLE BillingAccount (

B\_Name VARCHAR(100),

B\_Address VARCHAR(255),

Account\_Number CHAR(20),

FOREIGN KEY (Account\_Number) REFERENCES Account(Number)

);

CREATE TABLE Contract (

Contract\_number CHAR(20) NOT NULL,

Plan\_name VARCHAR(100),

Account\_Number CHAR(20),

PRIMARY KEY (Contract\_number),

FOREIGN KEY (Account\_Number) REFERENCES Account(Number)

);

CREATE TABLE ContractBenefit (

Benefit\_ID INT NOT NULL AUTO\_INCREMENT,

Contract\_number CHAR(20),

PRIMARY KEY (Benefit\_ID),

FOREIGN KEY (Contract\_number) REFERENCES Contract(Contract\_number)

);

CREATE TABLE ContractPremium (

Premium\_code CHAR(20) NOT NULL,

Contract\_number CHAR(20),

PRIMARY KEY (Premium\_code),

FOREIGN KEY (Contract\_number) REFERENCES Contract(Contract\_number)

);

CREATE TABLE PremiumMgmtContract (

Amount DECIMAL(10, 2),

Premium\_code CHAR(20),

PRIMARY KEY (Premium\_code),

FOREIGN KEY (Premium\_code) REFERENCES ContractPremium(Premium\_code)

);

CREATE TABLE Associate (

ID INT NOT NULL AUTO\_INCREMENT,

FirstName VARCHAR(100),

LastName VARCHAR(100),

DOB DATE,

Suffix VARCHAR(10),

Recruiter\_ID INT,

PRIMARY KEY (ID)

);

CREATE TABLE Recruiter (

ID INT NOT NULL AUTO\_INCREMENT,

LastName VARCHAR(100),

FirstName VARCHAR(100),

DOB DATE,

PRIMARY KEY (ID)

);

CREATE TABLE Invoice (

Number CHAR(20) NOT NULL,

Paid\_Date DATE,

Due\_Date DATE,

Associate\_ID INT,

PRIMARY KEY (Number),

FOREIGN KEY (Associate\_ID) REFERENCES Associate(ID)

);

CREATE TABLE WritingNumbers (

W\_number CHAR(20) NOT NULL,

Date DATE,

Status VARCHAR(50),

Associate\_ID INT,

PRIMARY KEY (W\_number),

FOREIGN KEY (Associate\_ID) REFERENCES Associate(ID)

);

CREATE TABLE ManagerContract (

Contract\_number CHAR(20) NOT NULL,

Type VARCHAR(50),

Issuedate DATE,

Sitecode CHAR(20),

PRIMARY KEY (Contract\_number)

);

CREATE TABLE Manage (

ManagerContract\_number CHAR(20),

WritingNumber\_W\_number CHAR(20),

FOREIGN KEY (ManagerContract\_number) REFERENCES ManagerContract(Contract\_number),

FOREIGN KEY (WritingNumber\_W\_number) REFERENCES WritingNumbers(W\_number)

);

CREATE TABLE Broker (

ID INT NOT NULL AUTO\_INCREMENT,

LastName VARCHAR(100),

FirstName VARCHAR(100),

DOB DATE,

Suffix VARCHAR(10),

PRIMARY KEY (ID)

);

CREATE TABLE Broker\_Associate (

Broker\_ID INT,

Associate\_ID INT,

FOREIGN KEY (Broker\_ID) REFERENCES Broker(ID),

FOREIGN KEY (Associate\_ID) REFERENCES Associate(ID)

);