Name Guo, Xinyu Wang, Yuqin Date: 12/20/2022

(last name, first name)

NYU ID: xg693, yw3875

Course Section Number: DS-GA 2433-001

Final Project

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

Professor’s Comments:

2433 Final Report

*Xinyu Guo, Yuqin Wang*

Link to Github: <https://github.com/xinyuuuuuG/Database-Final-Project>

**Introduction**

**Project 1:** This project is focused on the development of an Entity Relationship (ER) diagram for a database. This ER diagram provides the framework for the database, which consists of tables and relationships, in order to store and access data in an organized fashion. This project involves identifying the different entities, attributes, and relationships necessary for the database, and using this information to design the ER diagram.

**Project 2:** This project is focusing on building a database based on the Entity Relationship (ER) diagram developed in the previous project. This database is used to store and access data in an organized fashion. This project involves creating the tables, columns, entries, and relationships necessary for the database, using the ER diagram as a guide. It also involves setting up the database to ensure that data is properly stored and retrieved. The end result of this project is a fully functional database, allowing for efficient data storage and retrieval.

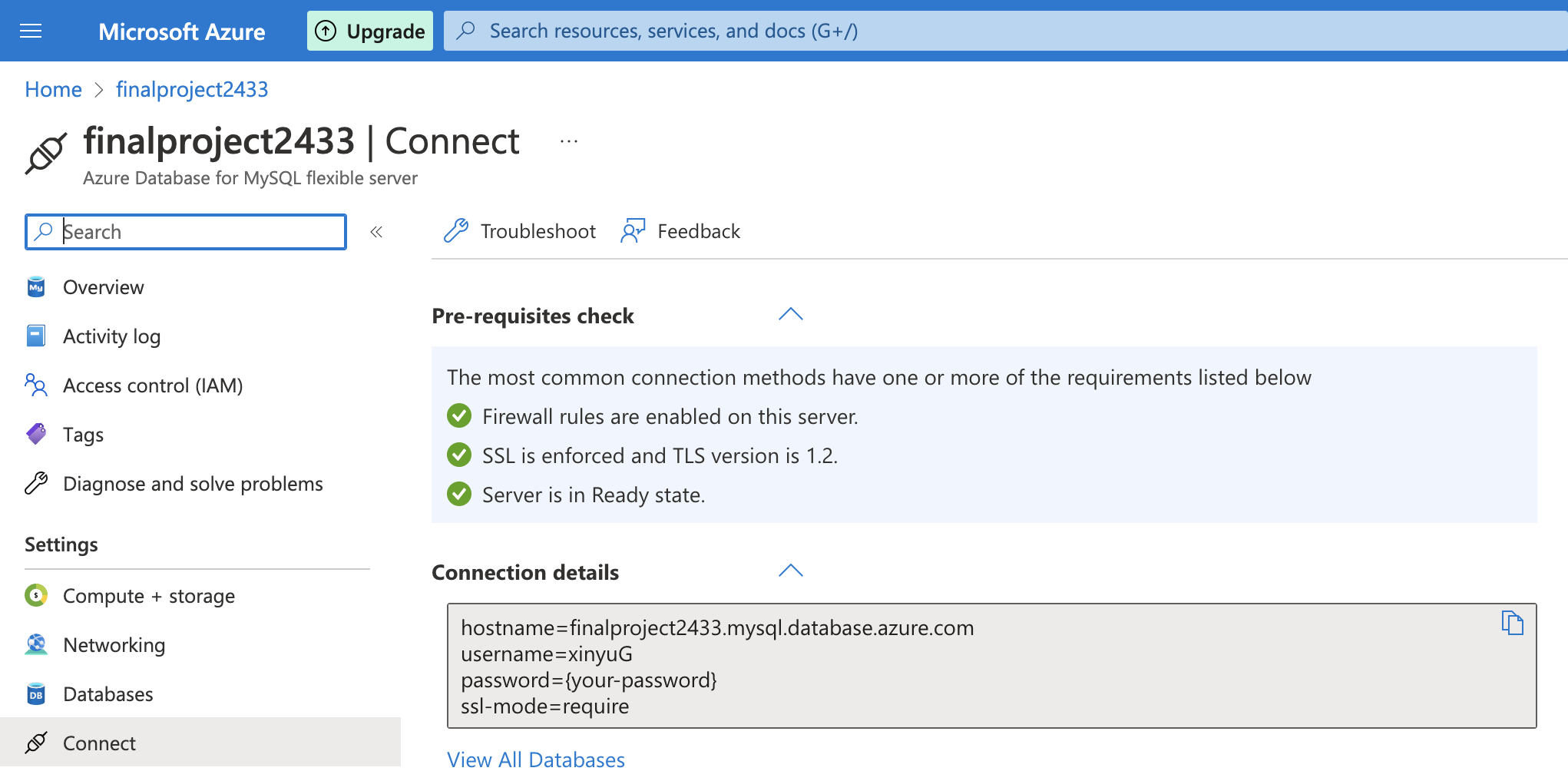
**Project 3:** This project is focused on implementing machine learning models to do predictions based on heart disease data. This project involves collecting health data and using it to train, validate, and test the accuracy of machine learning models. This project also involves optimizing the models and adjusting the parameters to make the classifications more accurate and reliable. The end result of this project will be a fully functional machine learning system that can best predict whether a person is likely to have a heart disease.

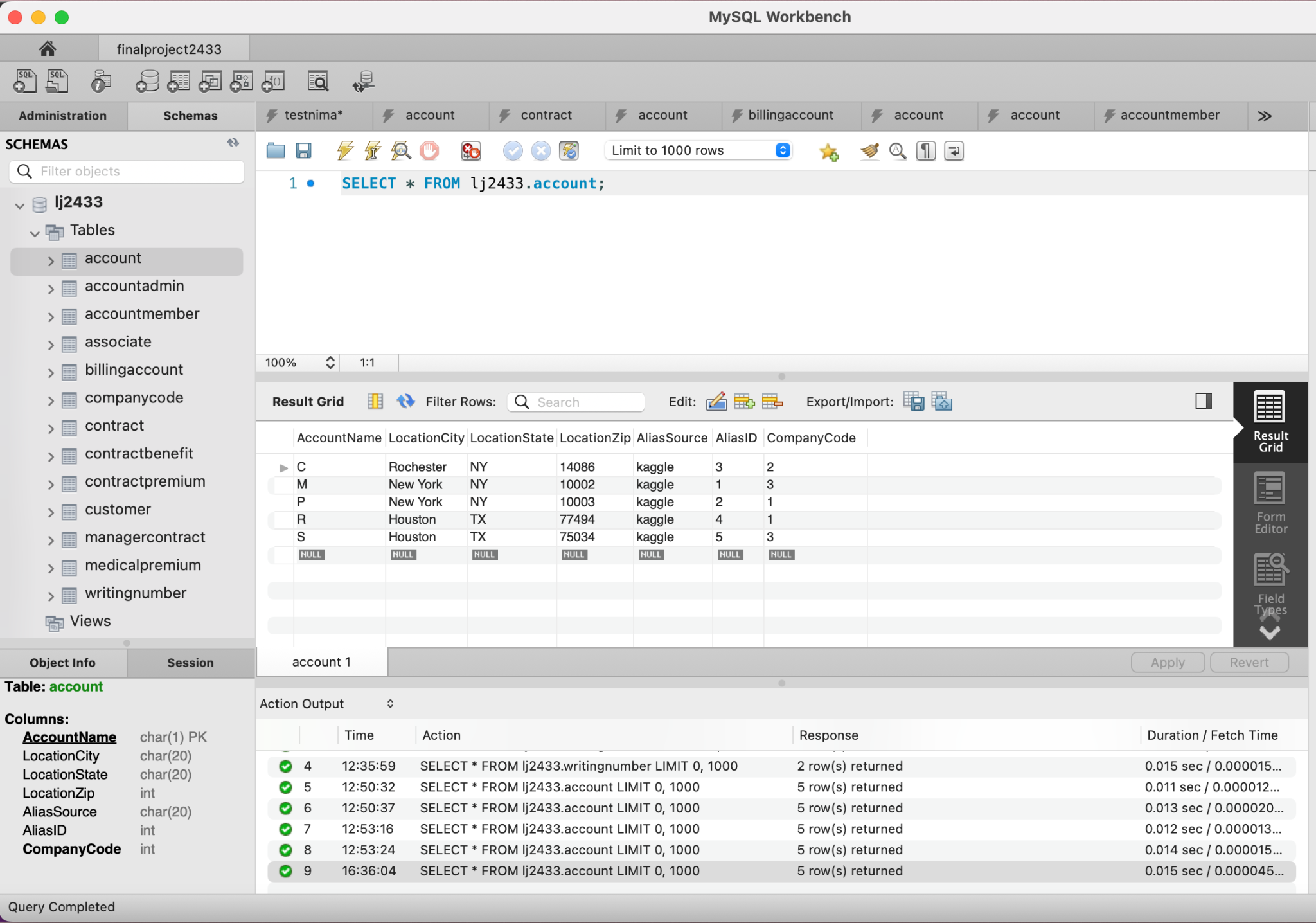
**Final Project:** This project is focused on building an interface to predict insurance quote prices based on the information entered by the user. The interface allows the user to enter their information, such as name, account , contract, and other relevant information. The end result of this project will be a fully functional interface that can predict insurance quote prices and allow the user to easily modify their information.

**Architecture for Application**

We build our architecture for application following the steps below.

Step 1. We use Microsoft Azure as the Cloud Database Platform, from which we get information including hostname, username, certificate etc.. It may require a certificate to connect the database to the python files and the html files. If the grader needs our certificate information, please email xg693@nyu.edu and yw3875@nyu.edu.

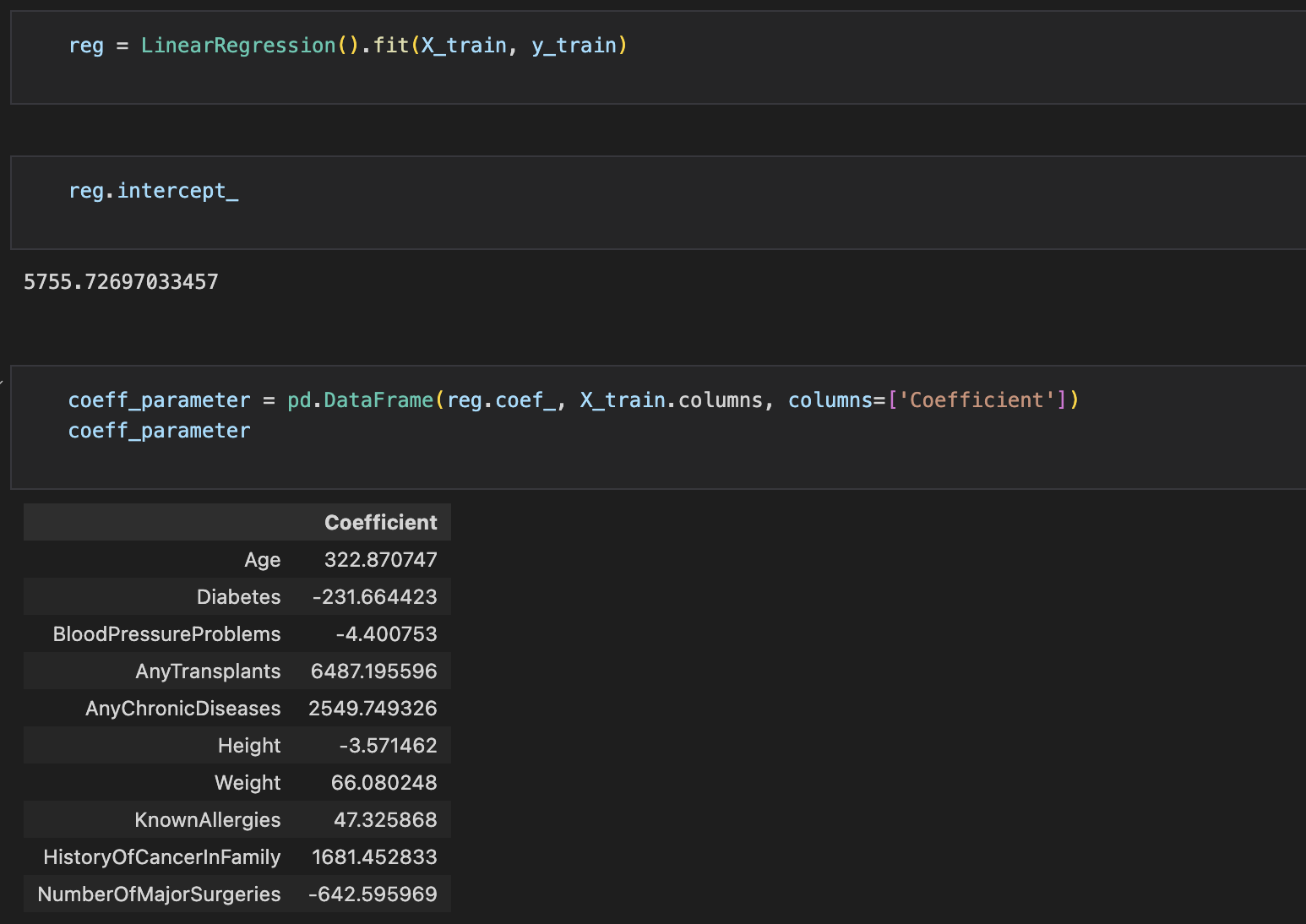


Step 2. Build the database in local using MySQL Workbench. The sql code and the data inserted are from Project 2.

Step 3. Build the ML model using the data from <https://www.kaggle.com/datasets/tejashvi14/medical-insurance-premium-prediction>.

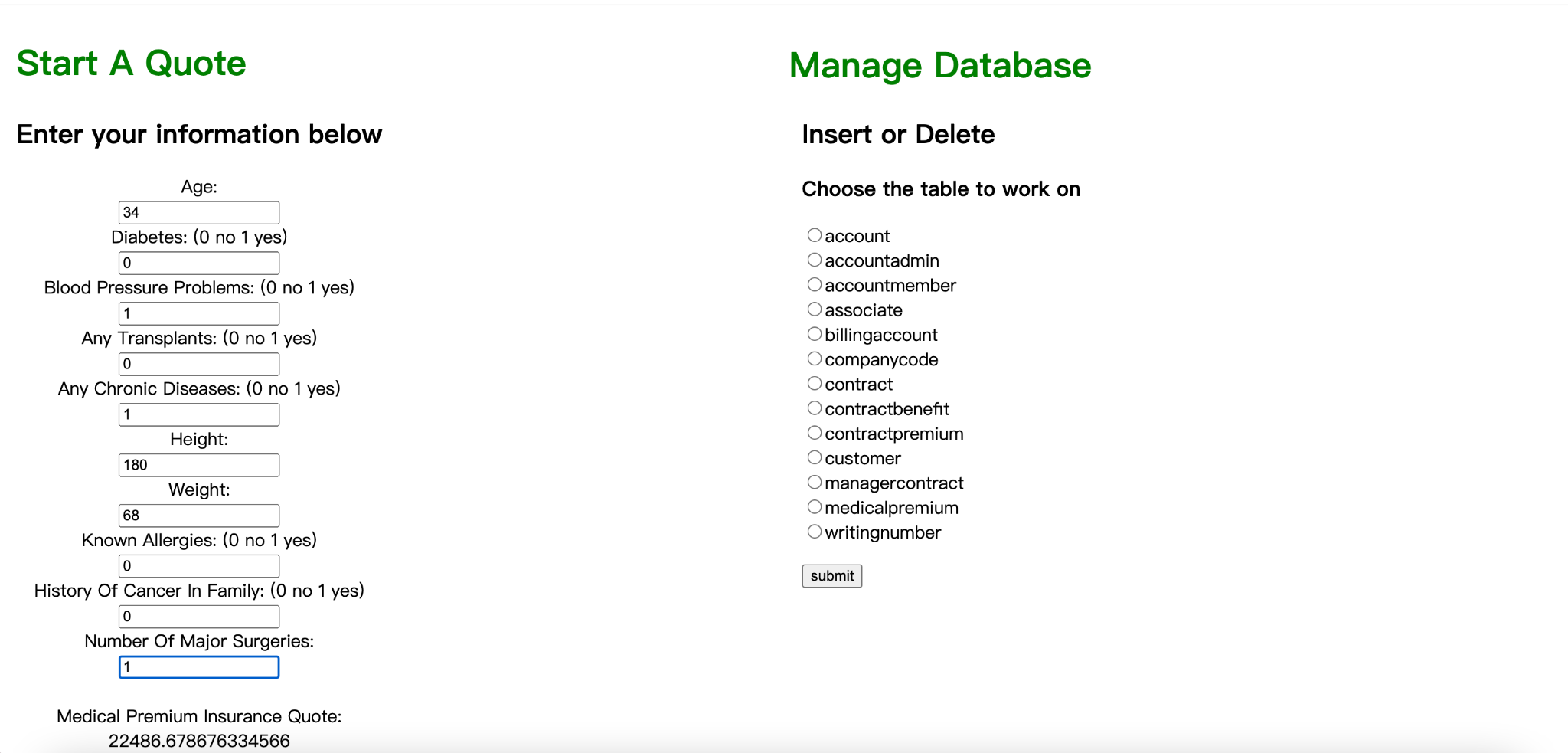
We also include this dataset as a table into our database that is mentioned in Step 2.

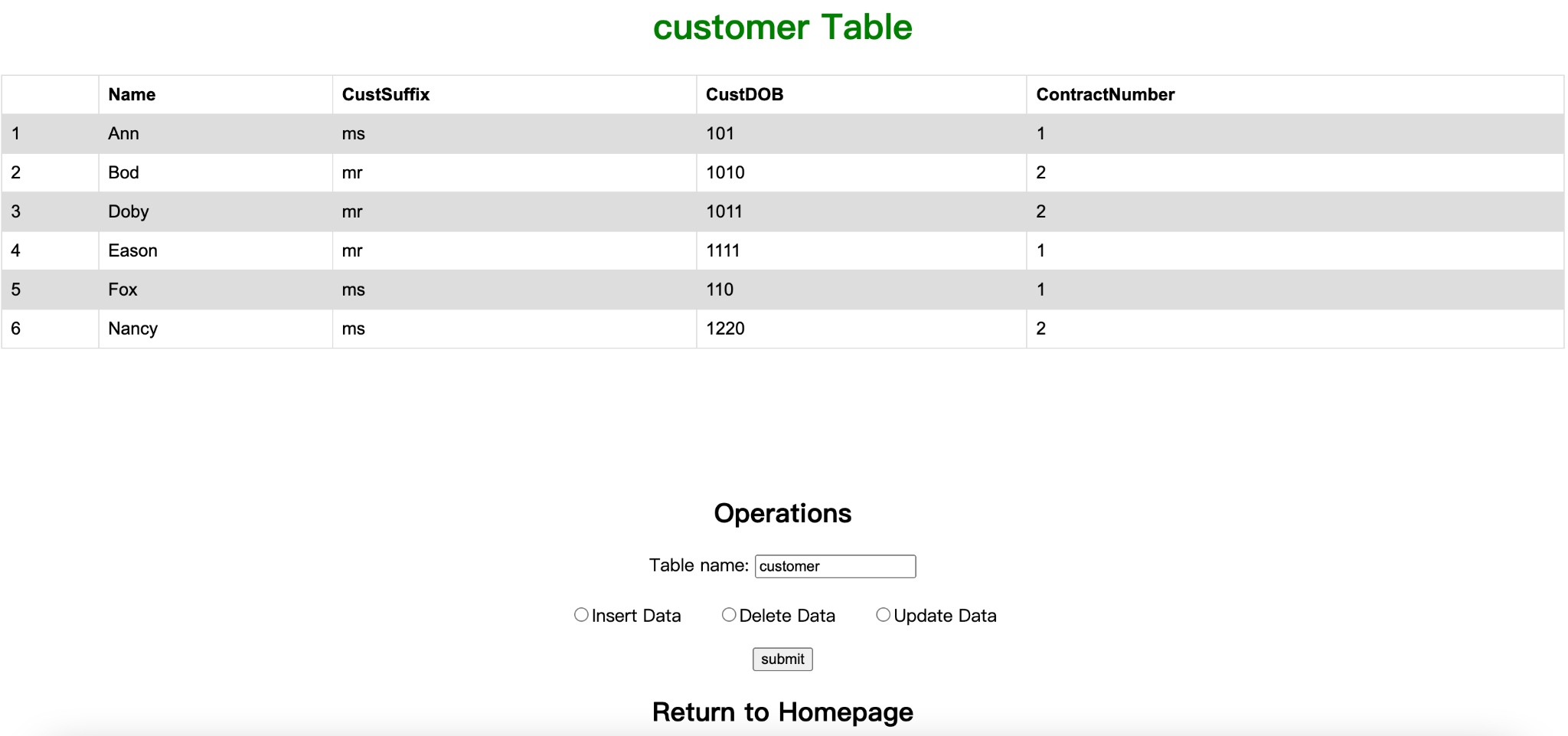
Though we have built a lot of ML models in Project 3 using the data from <https://www.kaggle.com/datasets/kamilpytlak/personal-key-indicators-of-heart-disease?resource=download>, the target value for the models is binary class (having value 0 or 1) and not suitable to be applied to a website user interface. Accordingly, we choose the new data set, which has the column *PremiumPrice*, a numerical value, as the target value. And we build a linear regression model that uses the other features, such as *Age*, *Height*, *Weight*, *KnownAllergies* and etc., to as the predictors.



Step 4. Construct the website interface using python and html. Our website is composed of two parts: *Start A Quote* and *Manage Database*.

*Start A Quote* is our application over the database, taking the user’s input as the values run into the ML model we built in Step 3, returning a *Medical Premium Insurance Quote* as the response. *Manage Database* embraces the functions to modify the values in our dataset, including insert, delete, and update. This also shows how our database is connected to the website user interface.

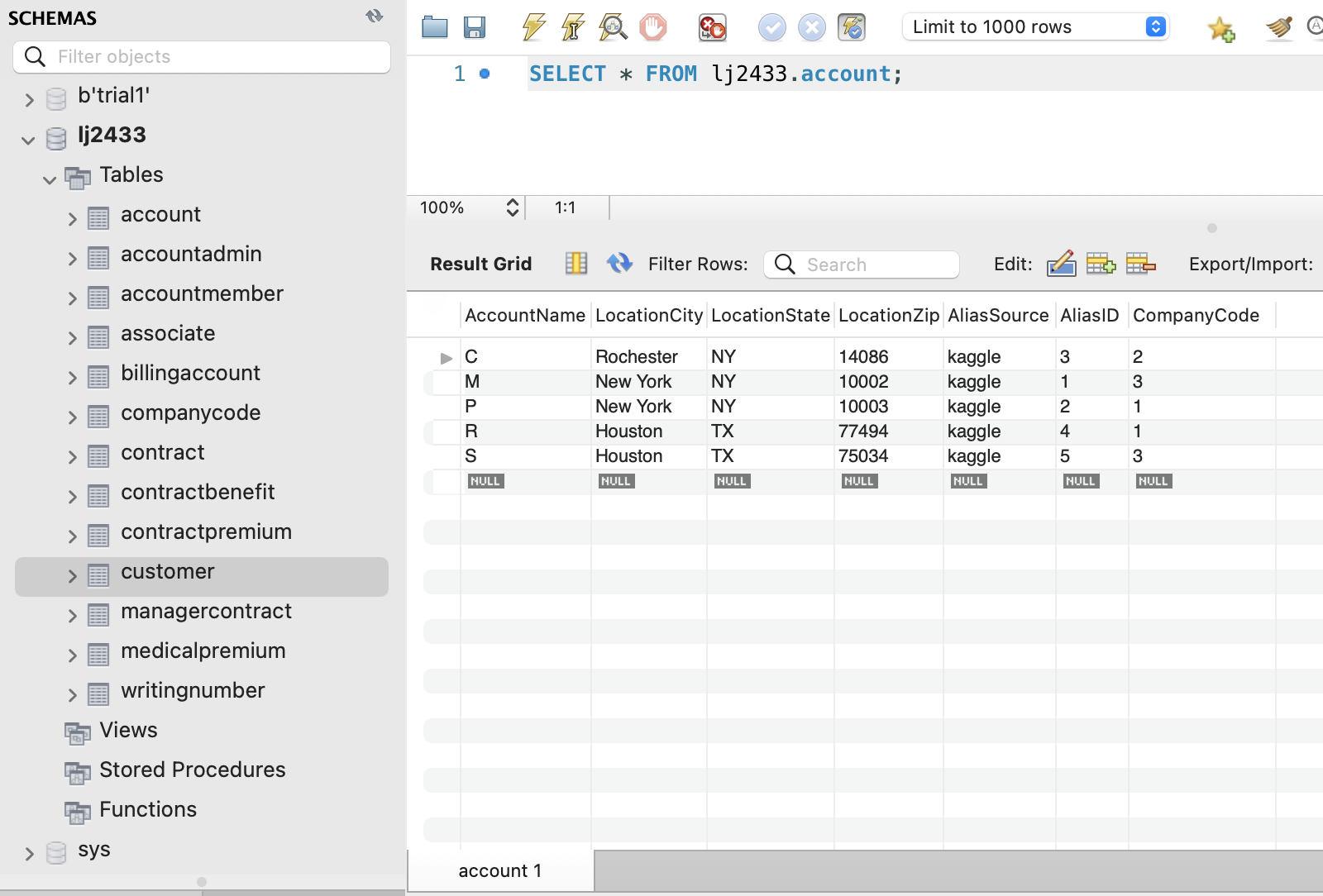




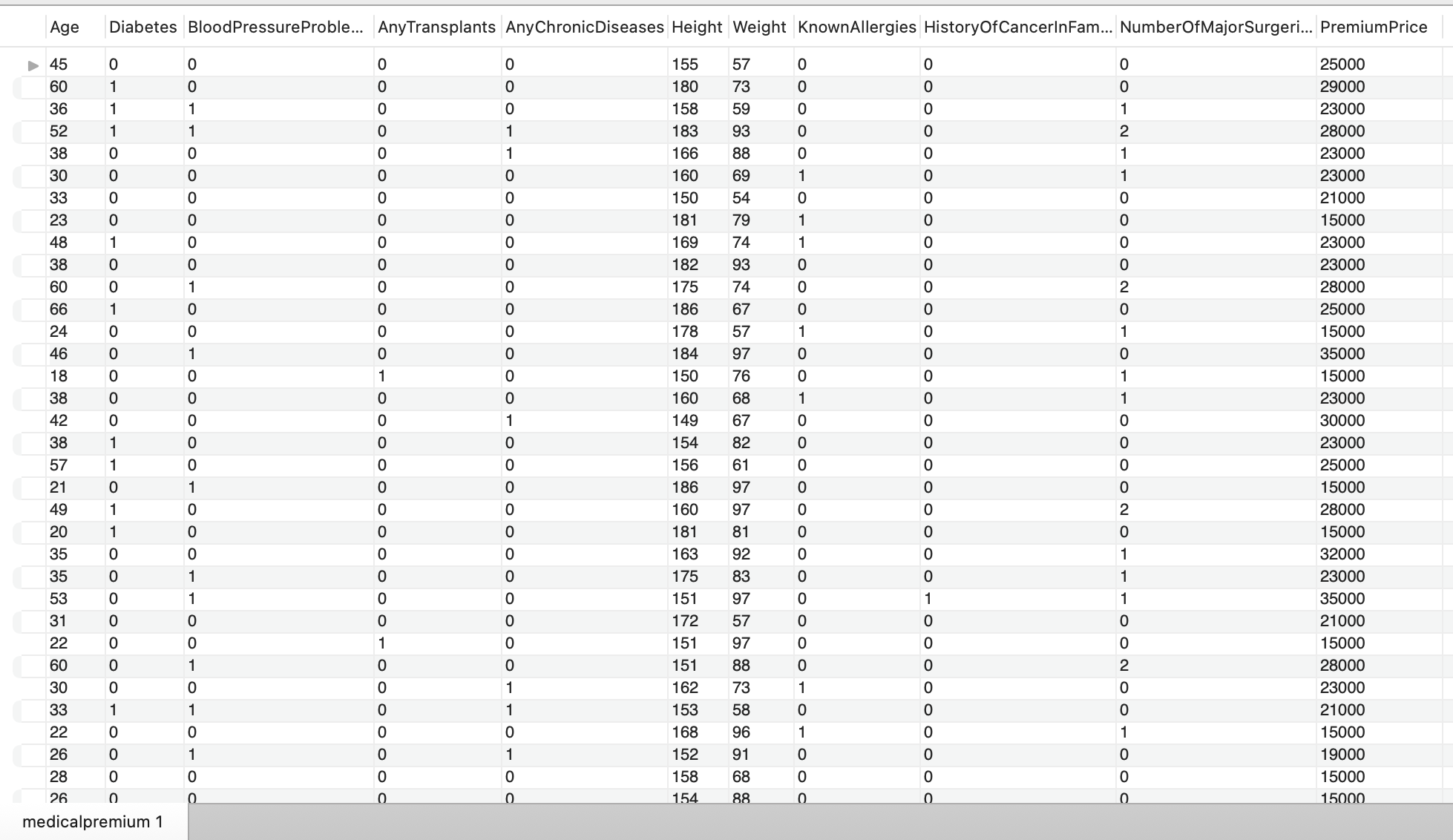
**Database Explanation**

We include data from two sources in our database. As mentioned in Step 3, we have data from Project 2 for most parts of our database, including the tables *account, associate, contract*, and etc.. On the other hand, to implement the ML model, we also include a table called *MedicalPremium* in our database.

1. Database from Project 2 that is constructed and has value generated using MySQL.

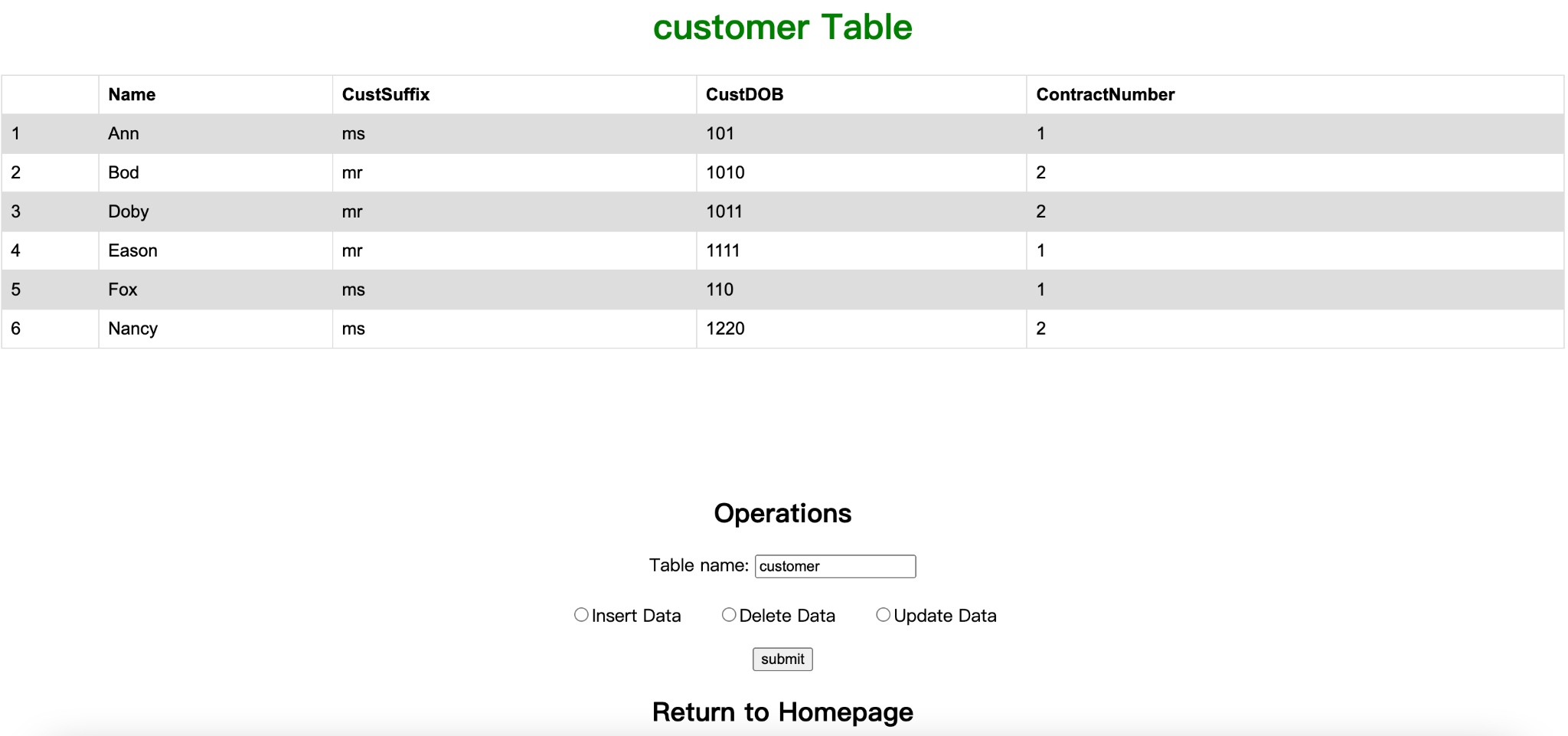


1. Medical-insurance-premium-prediction.csv



**Database Interface with ML Model**

As mentioned in Step 4 above, users can easily modify their information in our database by simply clicking on the corresponding tabs and buttons.



**Conclusion**

In summary, these four projects have all focused on developing databases and interfaces to improve efficiency and accuracy for an insurance company. The first project focused on creating an Entity Relationship (ER) diagram for the database, the second project focused on building the database based on the ER diagram, the third project focused on implementing machine learning models for predictions, and the final project focused on building an interface using machine learning prediction. All four projects have been successful in developing the necessary databases and interfaces to improve the efficiency and accuracy of data storage and retrieval.