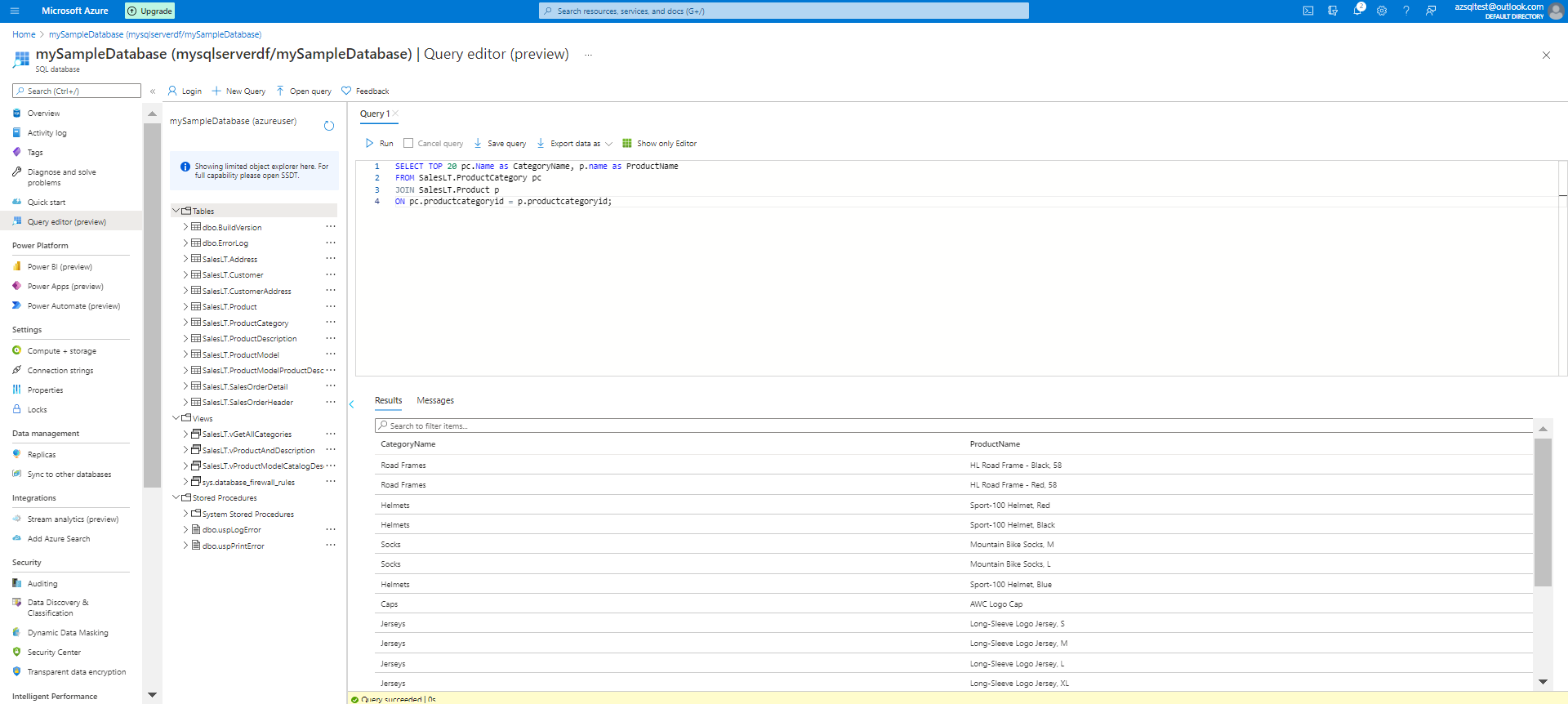
Creating Databases in SQL Objective

The database was created on Microsoft Azure, using their SQL Database service. This was due to it linking into a work project for myself (Azure was an environment I needed to become more familiar with).

Azure SQL Database is an intelligent, scalable, relational database service built for the cloud. Optimise performance and durability with automated, AI-powered features that are always up to date. With serverless compute and Hyperscale storage options that automatically scale resources on demand, you’re free to focus on building new applications without worrying about storage size or resource management.

The database was created using the serverless compute tier offering in Microsoft Azure, as it automatically scales compute based on workload demand and bills for the amount of compute used per second. The serverless compute tier also automatically pauses databases during inactive periods when only storage is billed and automatically resumes databases when activity returns.

After creating the initial database, I uploaded some sample data to it to ensure it ran correctly. The output of a query can be seen below:



**Notes**

The first step to designing any database in SQL is to identify what to include and what not to include. The next steps involve deciding how the included items relate to each other and then setting up tables accordingly.

To design a database in SQL, follow these basic steps:

1. Decide what objects you want to include in your database.
2. Determine which of these objects should be tables and which should be columns within those tables.
3. Define tables based on how you need to organize the objects.

*Step 1: Define objects*

The first step in designing a database is deciding which aspects of the system are important enough to include in the model. Create a full list of all the objects you can think of and then decide how the objects relate to each other. Ensuring to identify the major entities and their subsidiaries.

*Step 2: Identify tables and columns*

Major entities translate into database tables, and each of them has a set of attributes — the table columns. When trying to identify which objects should be tables and which attributes in the system belong to which tables, consider:

* The information you want to get from the database
* How you want to use that information

When deciding how to structure database tables, involve future users of the database as well as the people who will make decisions based on database information.

*Step 3: Define tables*

Define a table for each entity and a column for each attribute.