**CS 6065 - Intro to Cloud Computing Assignment -1**

**By**

**Bhavitha Uppala**

1. **Setting up Storage Account**

A computer screen shot of a computer screen

Description automatically generated

A screenshot of a computer

Description automatically generated

1. **Setting up SǪL database and creating Students table**

**A screenshot of a computer

Description automatically generated**

**A screenshot of a computer

Description automatically generated**

**A screenshot of a computer

Description automatically generated**

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

**A screenshot of a computer

Description automatically generated**

1. **Building Data pipeline in Azure Data Factory**

A computer screen shot of a computer

Description automatically generated**A screenshot of a computer

Description automatically generated**

A computer screen shot of a computer

Description automatically generated

A computer screen shot of a computer screen

Description automatically generated

A computer screen shot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

A computer screen shot of a computer screen

Description automatically generated

A screenshot of a computer

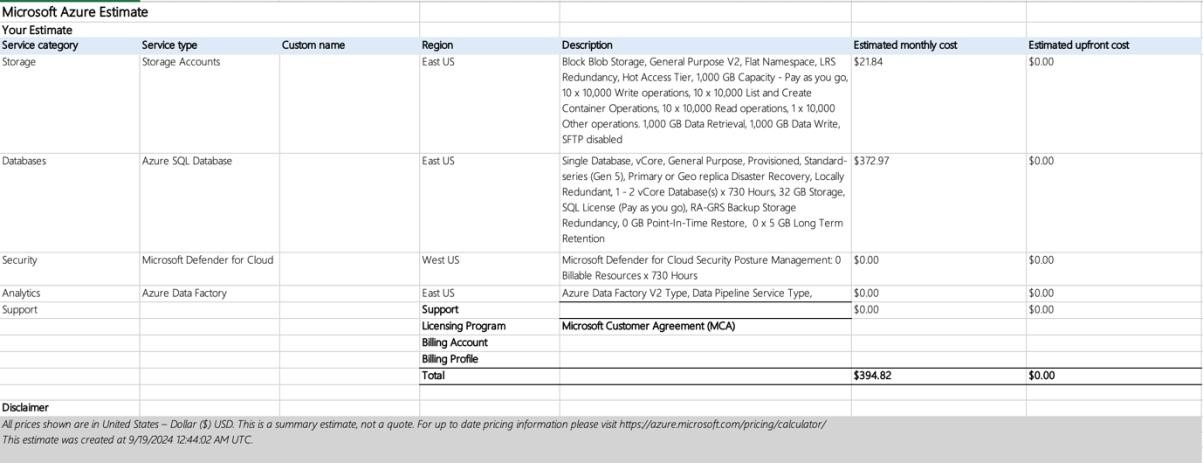
Description automatically generated

**4)**

* **Authentication**: SǪL Server Authentication
* **Server name**: bhavitua.database.windows.net
* **Database name: bhavitha (bhavitua/bhavitha)**
* **Login**: sysadmin
* **Password**: bhaviU@2306

**5)**

**Pricing Estimate**



Storage accounts, SǪL database and Data factory are used in the project to save the

.csv file, store the entries in a database and copy all data from .csv to the cloud respectively. Microsoft defender is used as a security measure to protect the server from malicious attacks and unauthorised access. The total estimated monthly cost is

$394.82 and there is no upfront cost since an azure student account was used

**6)**

**Explanation:**

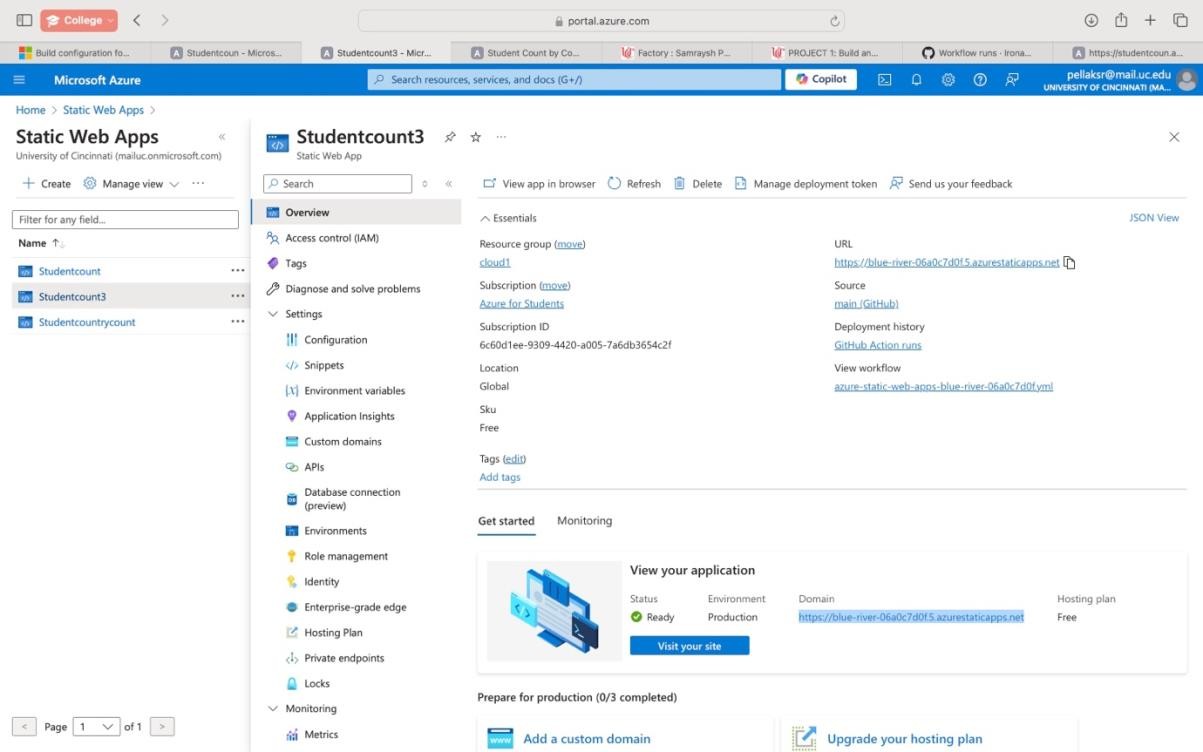
The objective of the project is to extract data from a .csv file to the cloud database.

First set up an Azure Storage account and upload the student-dataset.csv file. Then, create an Azure SǪL Database and the corresponding Students table. Next, set up an Azure Data Factory instance, configure linked services for Azure Blob Storage and the Azure SǪL Database, and create datasets for both the CSV file and the SǪL table. Build

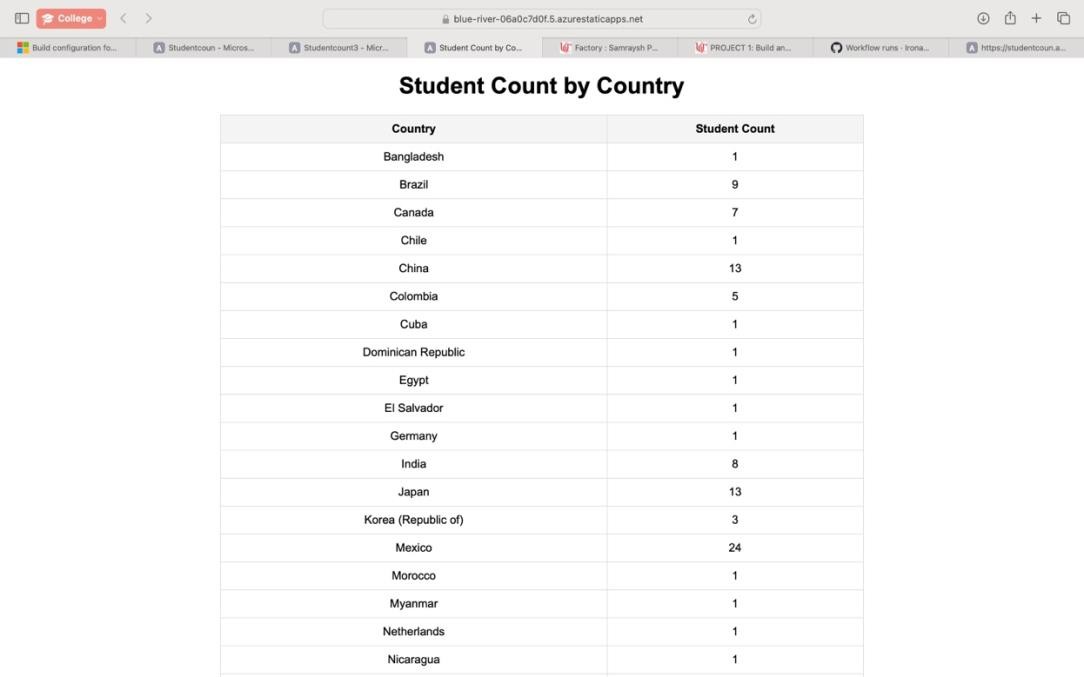
a pipeline using the Copy Data activity to transfer the data, test and publish the pipeline, and verify the transfer. Finally, query the SǪL database to retrieve the student count by country. The resulting query output is displayed as the output.

**Extra Credit**

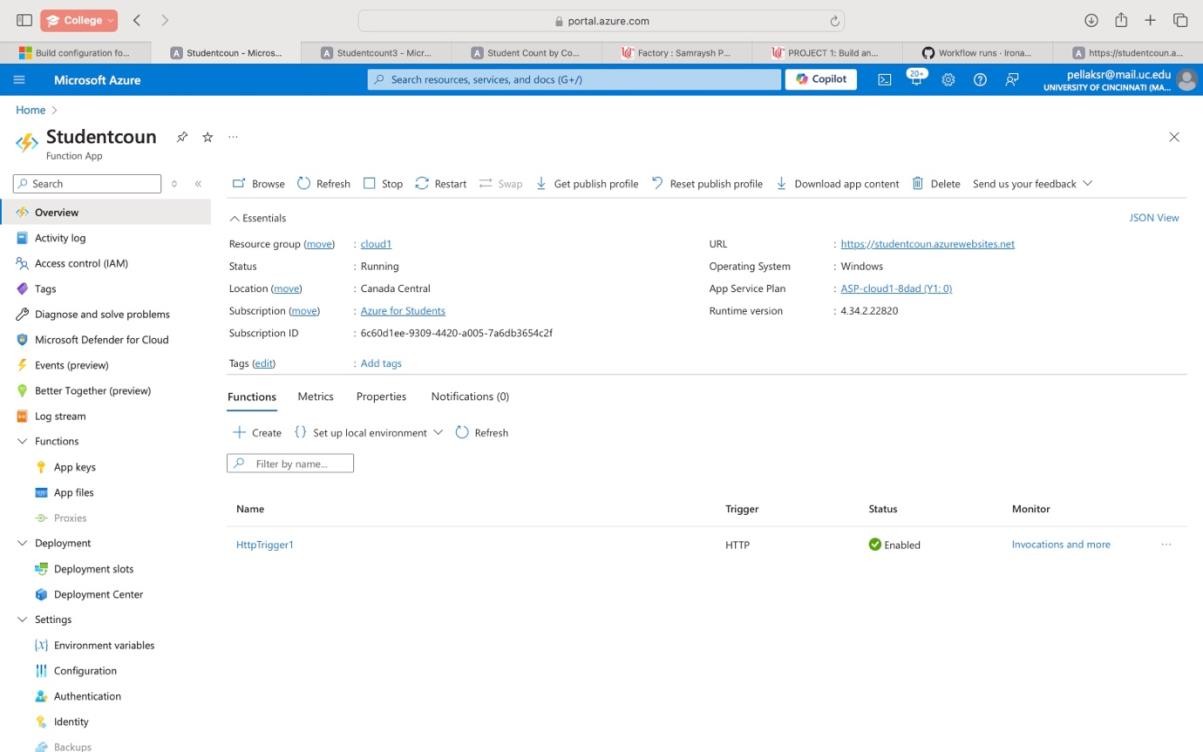
**Azure Static Web Apps Page:**



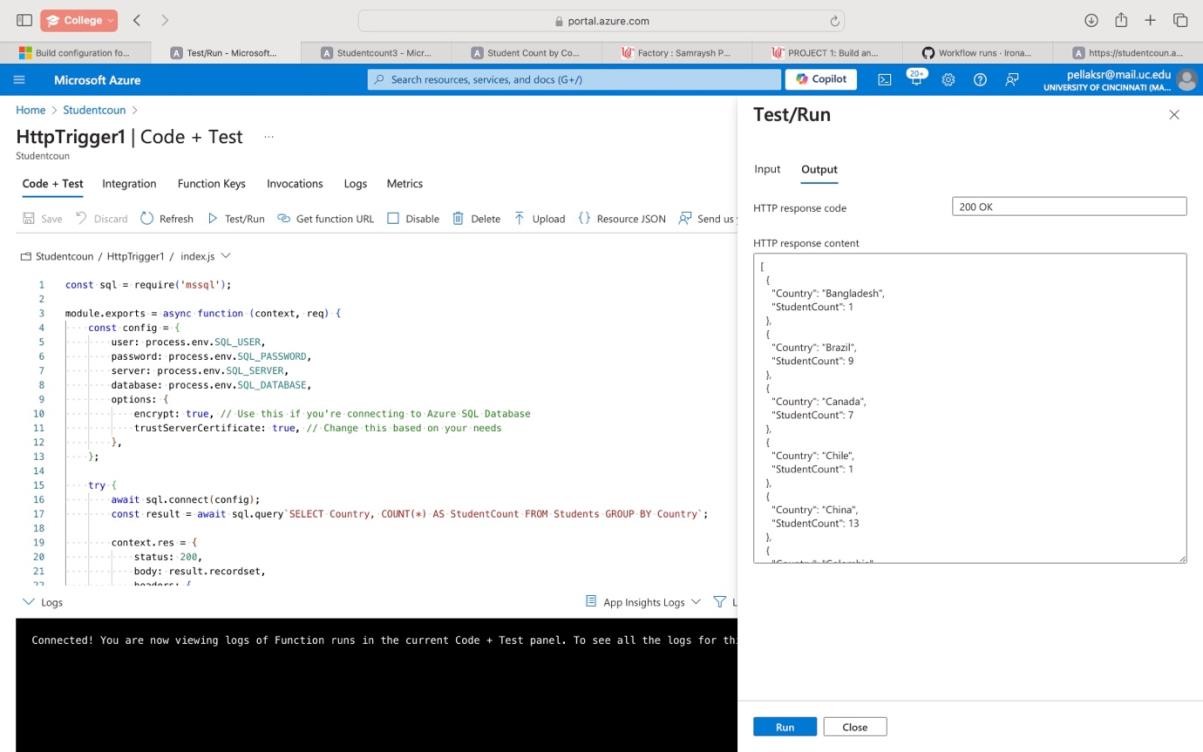
**HTML Page (Working):**



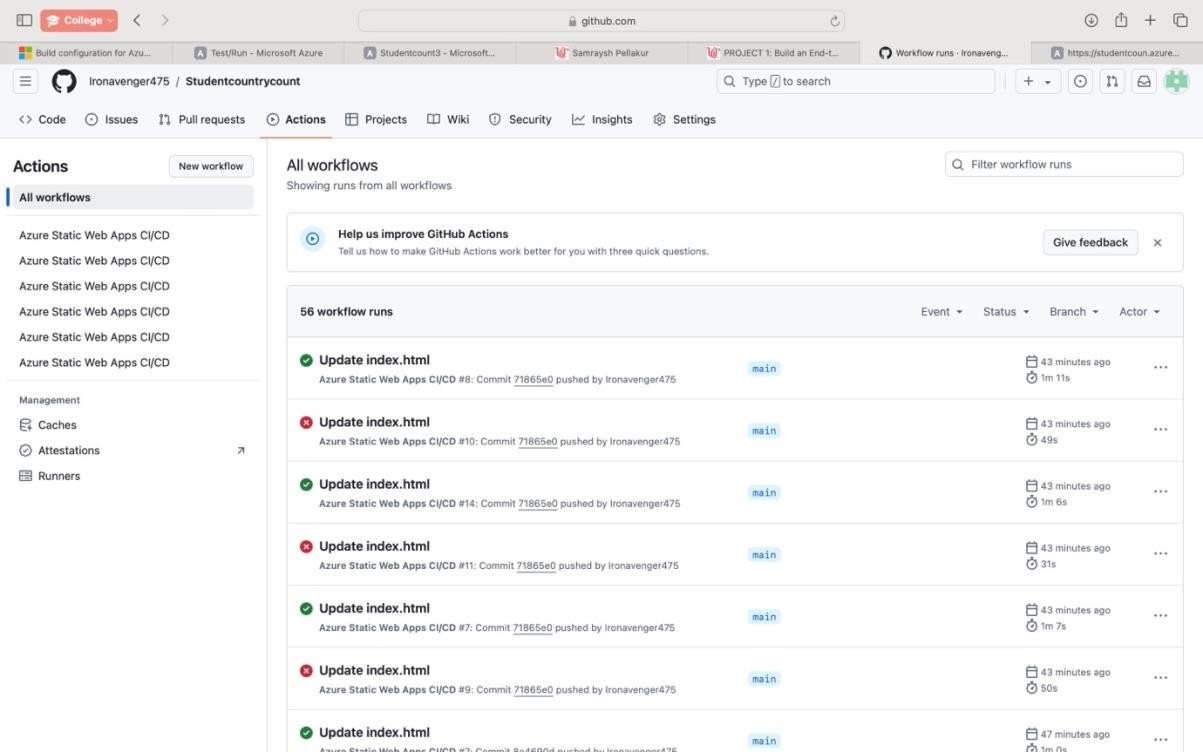
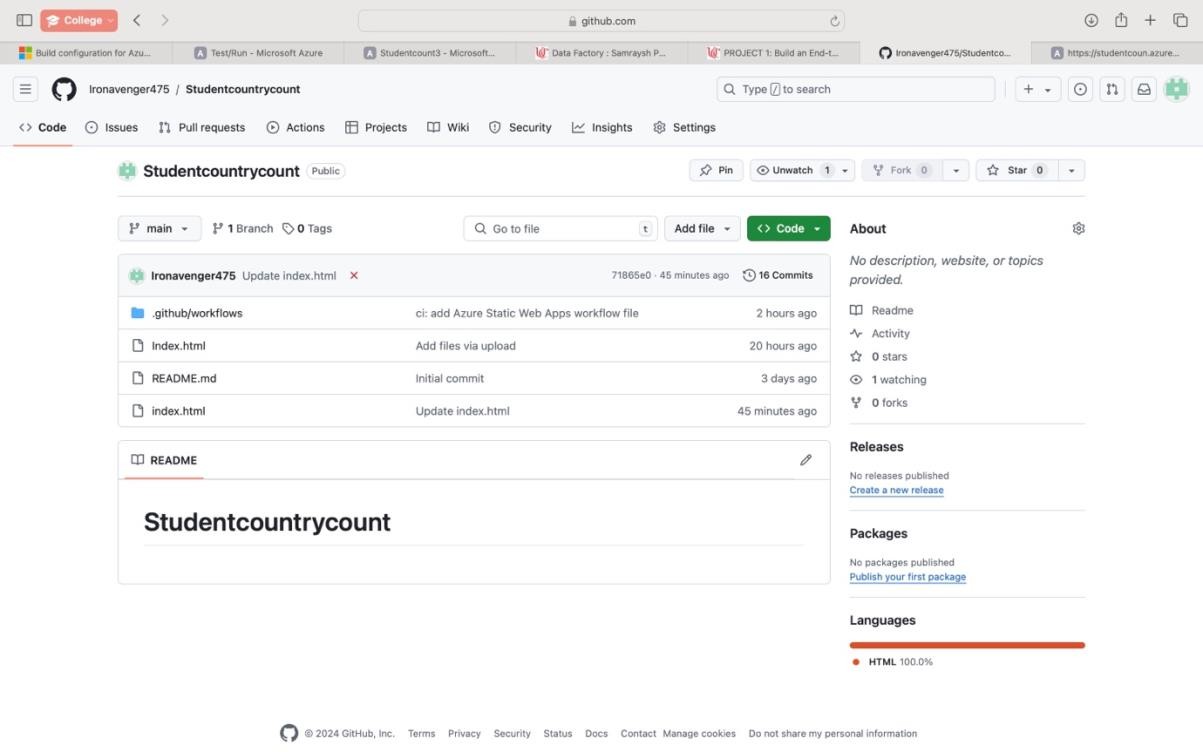
**Azure Function App Page:**



**HTTP Trigger Function Page (Working):**



**Github Page:**



**Link to Github Repository:**

[**https://github.com/Ironavenger475/Studentcountrycount**](https://github.com/Ironavenger475/Studentcountrycount) **Function URL:**

[**https://studentcoun.azurewebsites.net/api/HttpTrigger1?**](https://studentcoun.azurewebsites.net/api/HttpTrigger1) **Static Web App URL (Working):**

[**https://blue-river-06a0c7d0f.5.azurestaticapps.net**](https://blue-river-06a0c7d0f.5.azurestaticapps.net/) **Explanation:**

The objective here is to create a static web application hosted on Azure that retrieves and displays the student count by country from data stored in an Azure SǪL Database.

The tools used are:

* Azure Static Web Apps
* Azure Function App
* Github
* Azure SǪL Database
* Azure SǪL Server

**Implementation:**

First, prepare your data by creating a Students table with the necessary columns. Next, develop a web application using HTML, CSS, and JavaScript to fetch data from an API endpoint. Deploy this application to Azure Static Web Apps by creating a new app and linking it to your code repository. After that, create an Azure Function that serves as an API to query the SǪL Database for student counts, ensuring you configure the required environment settings. Update your web application to connect with the Azure Function API. Finally, test the application to verify it displays the correct data, and document your work with screenshots and explanations of your implementation and results. The end result will be a web page that retrieves and displays the student count by country from data stored in the SǪL database.