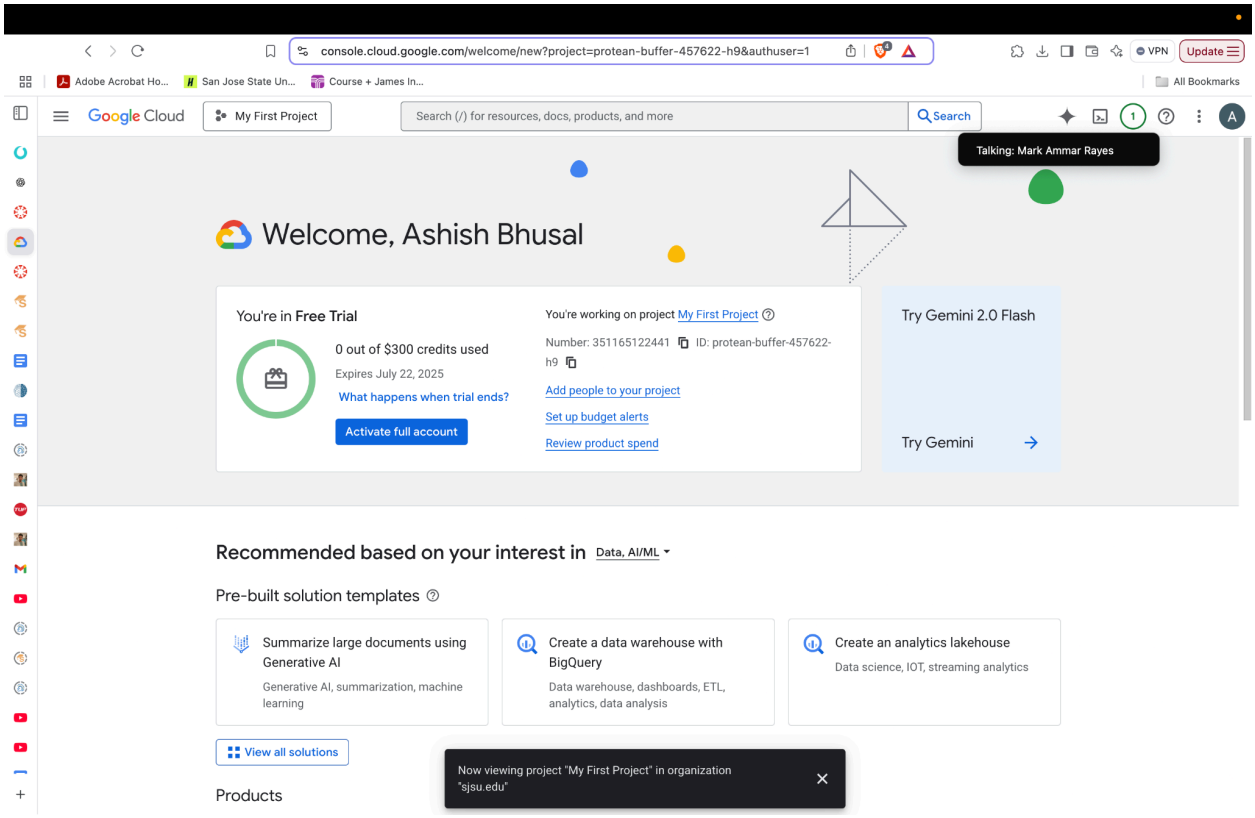
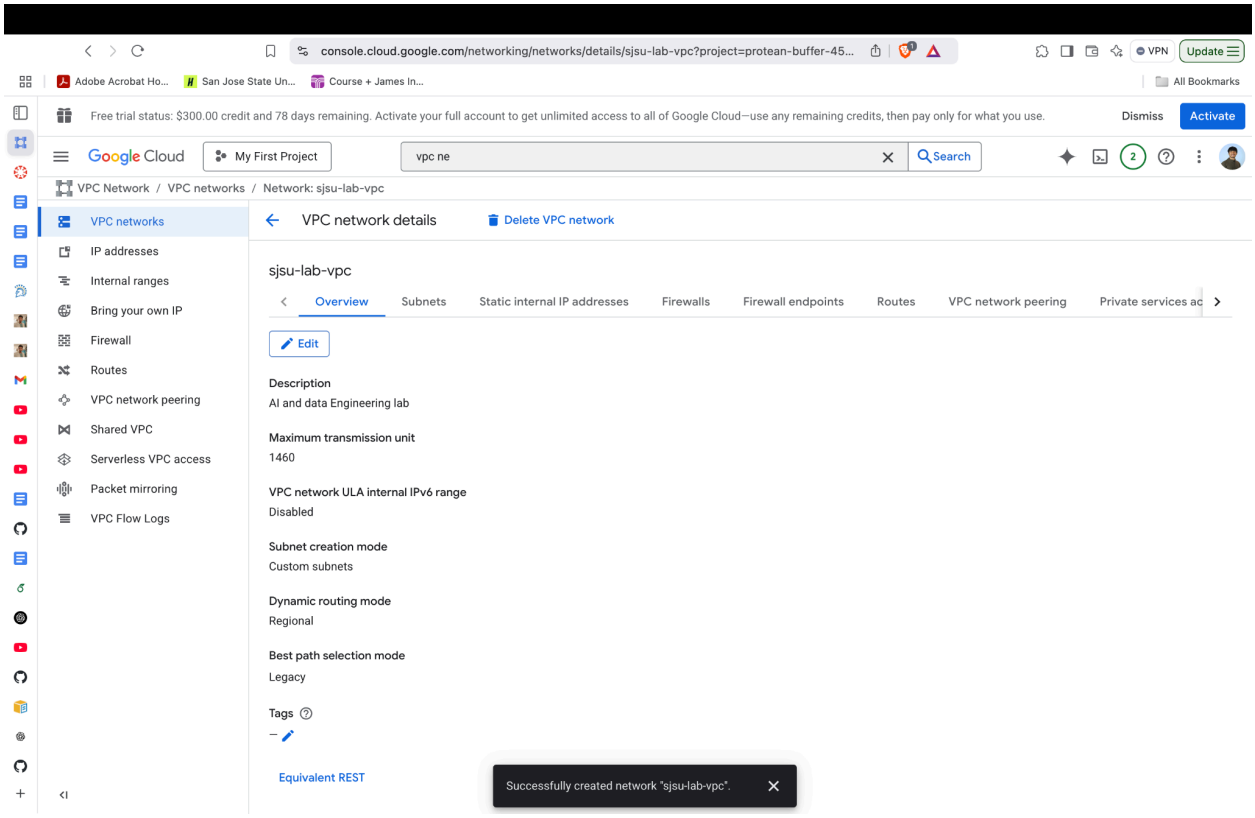
CMPE 252 - AI: Lab 2

Github: https://github.com/bhusalashish/AI_LAB2/

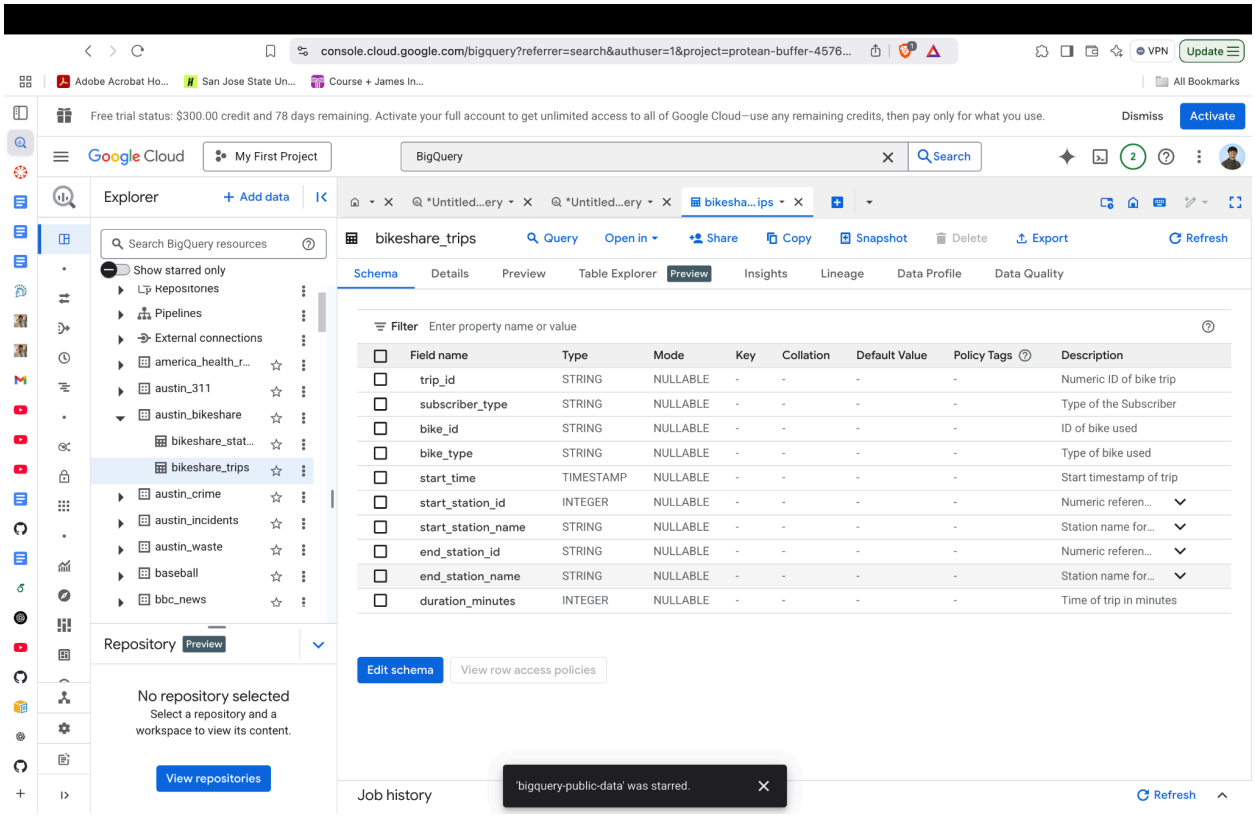
Step 1: Set Up a Free GCP Account

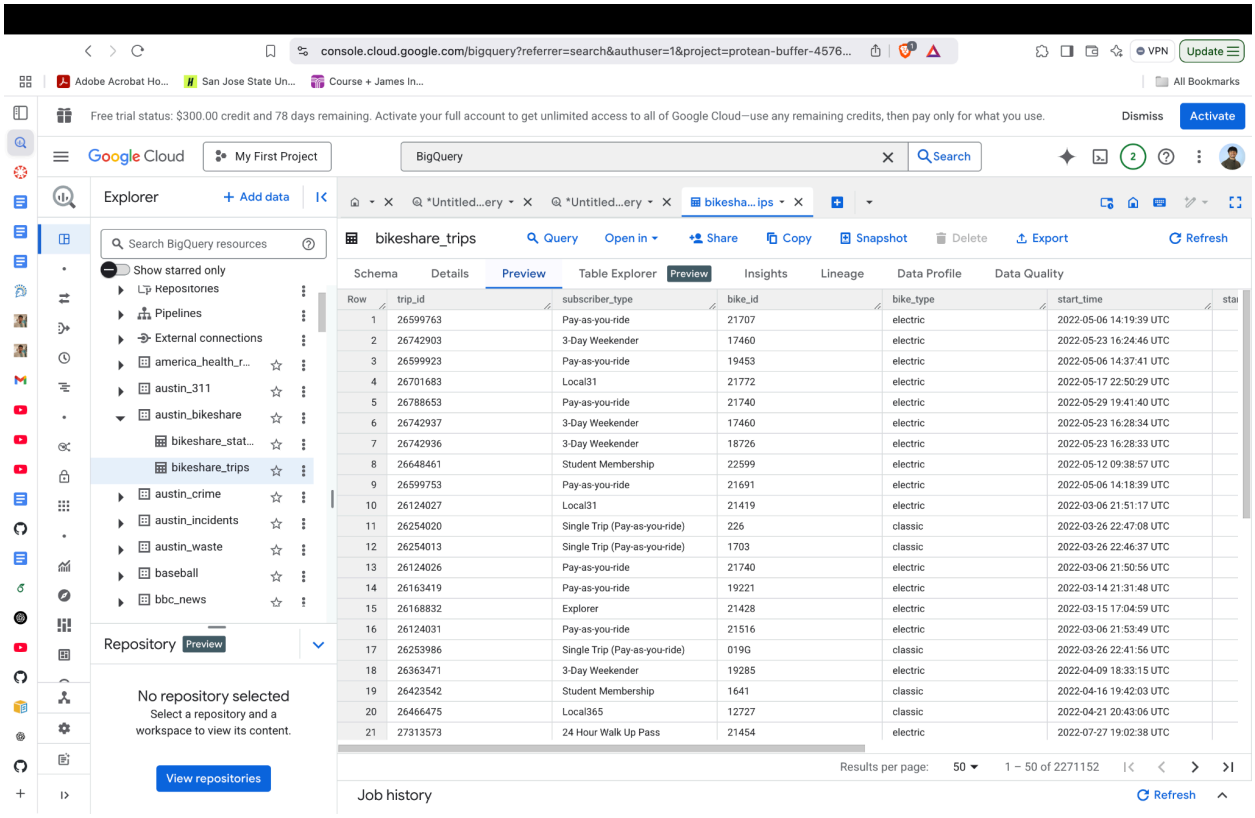


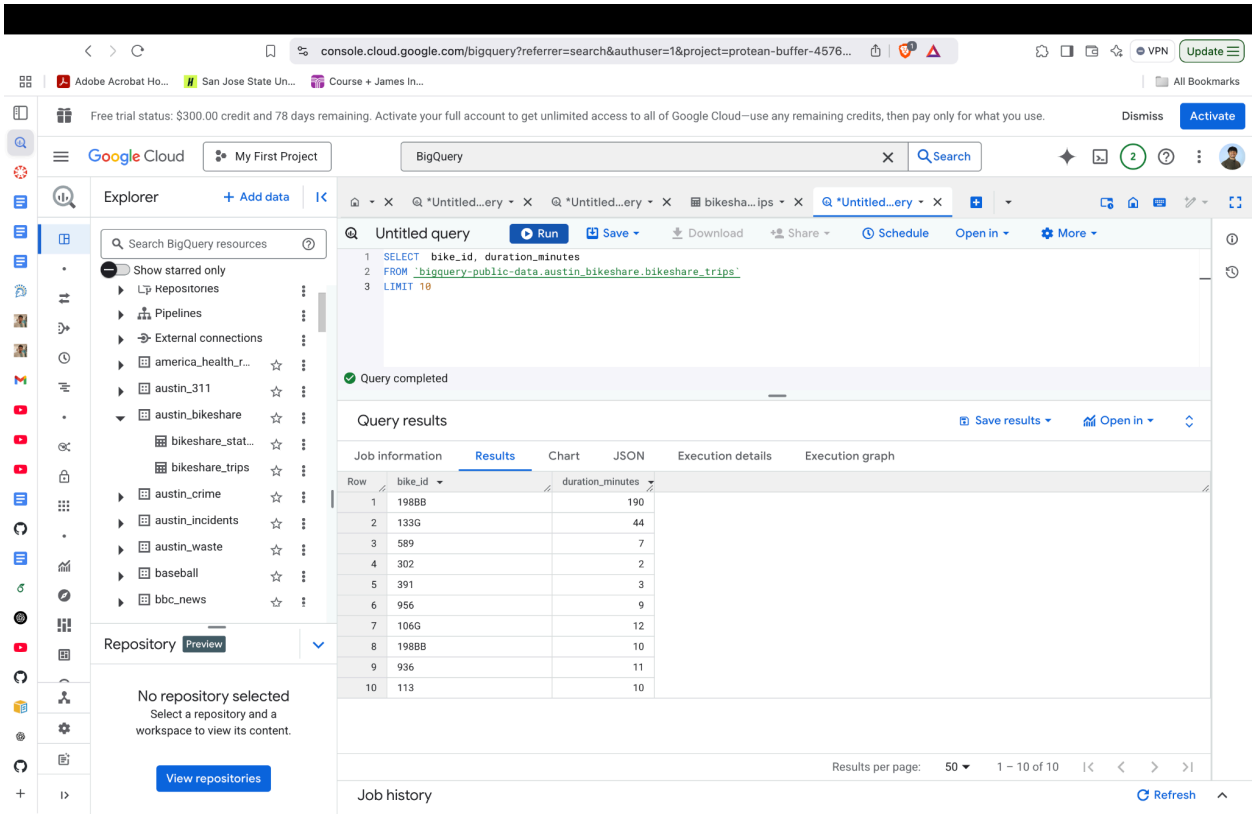
Step 2: Create a Virtual Private Cloud (VPC)



Step 3: Explore BigQuery

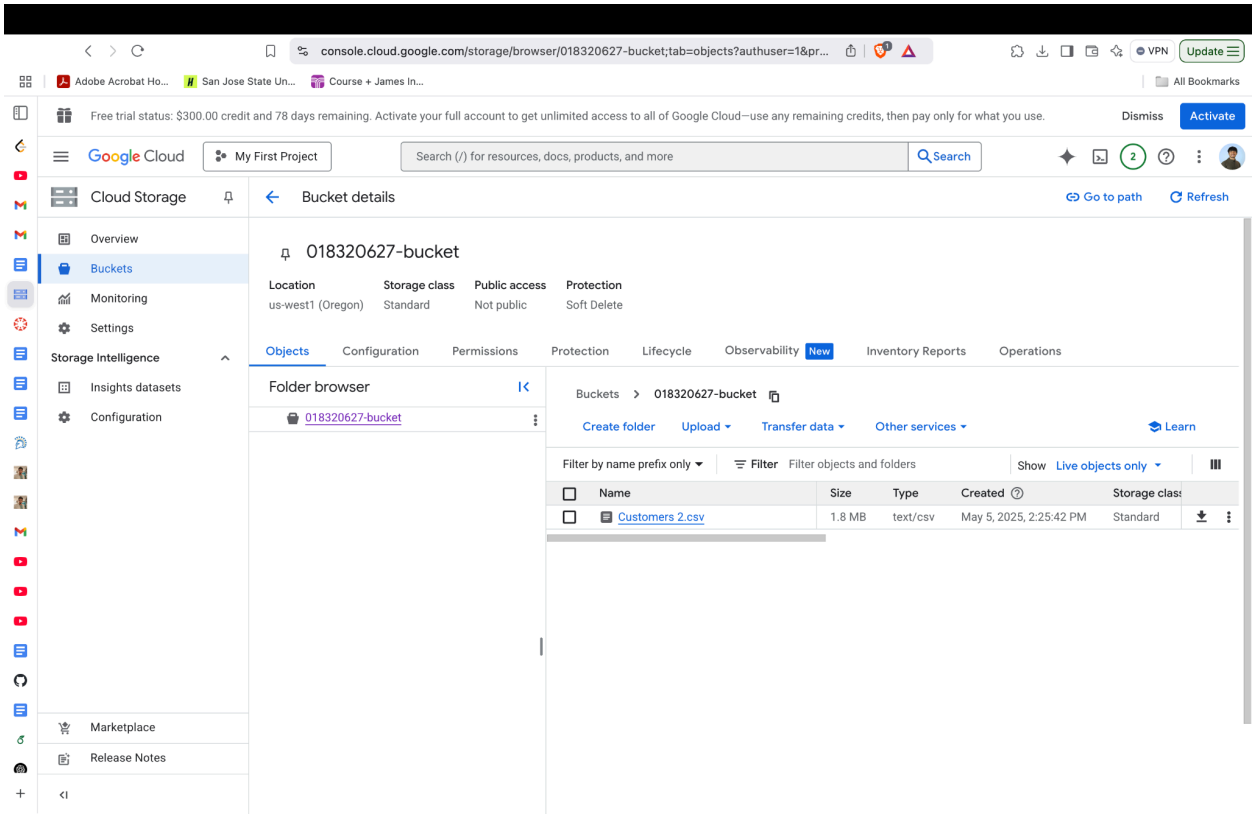

console.cloud.google.com/bigquery?referrer=search&authuser=1&project=protean-buffer-4576...
Free trial status: $300.00 credit and 78 days remaining. Activate your full account to get unlimited access to all of Google Cloud—use any remaining credits, then pay only for what you use.
Dismiss
Activate
Google Cloud
My First Project
BigQuery
Search
Explorer
Add data
Search BigQuery resources
Show starred only
Repositories
Pipelines
External connections
america_health_r...
austin_311
austin_bikeshare
bikeshare_stat...
bikeshare_trips
austin_crime
austin_incidents
austin_waste
baseball
bbc_news
Repository
Preview
No repository selected
Select a repository and a workspace to view its content.
View repositories
bikeshare_trips
Query
Open in
Share
Copy
Snapshot
Delete
Export
Refresh
Schema
Details
Preview
Table Explorer
Preview
Insights
Lineage
Data Profile
Data Quality
Filter
Enter property name or value
Field name
Type
Mode
Key
Collation
Default Value
Policy Tags
Description
trip_id STRING NULLABLE Numeric ID of bike trip
subscriber_type STRING NULLABLE Type of the Subscriber
bike_id STRING NULLABLE ID of bike used
bike_type STRING NULLABLE Type of bike used
start_time TIMESTAMP NULLABLE Start timestamp of trip
start_station_id INTEGER NULLABLE Numeric referen...
start_station_name STRING NULLABLE Station name for...
end_station_id STRING NULLABLE Numeric referen...
end_station_name STRING NULLABLE Station name for...
duration_minutes INTEGER NULLABLE Time of trip in minutes
Edit schema
View row access policies
'bigquery-public-data' was starred.
Job history
Refresh


console.cloud.google.com/bigquery?referrer=search&authuser=1&project=protean-buffer-4576...
Free trial status: $300.00 credit and 78 days remaining. Activate your full account to get unlimited access to all of Google Cloud—use any remaining credits, then pay only for what you use.
Dismiss
Activate
Google Cloud
My First Project
BigQuery
Search
Explorer
+ Add data
Search BigQuery resources
Show starred only
Repositories
Pipelines
External connections
america_health_r...
austin_311
austin_bikeshare
bikeshare_stat...
bikeshare_trips
austin_crime
austin_incidents
austin_waste
baseball
bbc_news
Repository
Preview
No repository selected
Select a repository and a workspace to view its content.
View repositories
bikeshare_trips
Query
Open in
Share
Copy
Snapshot
Delete
Export
Refresh
Schema
Details
Preview
Table Explorer
Preview
Insights
Lineage
Data Profile
Data Quality
Row trip_id subscriber_type bike_id bike_type start_time
1 26599763 Pay-as-you-ride 21707 electric 2022-05-06 14:19:39 UTC
2 26742903 3-Day Weekender 17460 electric 2022-05-23 16:24:46 UTC
3 26599923 Pay-as-you-ride 19453 electric 2022-05-06 14:37:41 UTC
4 26701683 Local31 21772 electric 2022-05-17 22:50:29 UTC
5 26788653 Pay-as-you-ride 21740 electric 2022-05-29 19:41:40 UTC
6 26742937 3-Day Weekender 17460 electric 2022-05-23 16:28:34 UTC
7 26742936 3-Day Weekender 18726 electric 2022-05-23 16:28:33 UTC
8 26648461 Student Membership 22599 electric 2022-05-12 09:38:57 UTC
9 26599753 Pay-as-you-ride 21691 electric 2022-05-06 14:18:39 UTC
10 26124027 Local31 21419 electric 2022-03-06 21:51:17 UTC
11 26254020 Single Trip (Pay-as-you-ride) 226 classic 2022-03-26 22:47:08 UTC
12 26254013 Single Trip (Pay-as-you-ride) 1703 classic 2022-03-26 22:46:37 UTC
13 26124026 Pay-as-you-ride 21740 electric 2022-03-06 21:50:56 UTC
14 26163419 Pay-as-you-ride 19221 electric 2022-03-14 21:31:48 UTC
15 26168832 Explorer 21428 electric 2022-03-15 17:04:59 UTC
16 26124031 Pay-as-you-ride 21516 electric 2022-03-06 21:53:49 UTC
17 26253986 Single Trip (Pay-as-you-ride) 019G classic 2022-03-26 22:41:56 UTC
18 26363471 3-Day Weekender 19285 electric 2022-04-09 18:33:15 UTC
19 26423542 Student Membership 1641 classic 2022-04-16 19:42:03 UTC
20 26466475 Local365 12727 classic 2022-04-21 20:43:06 UTC
21 27313573 24 Hour Walk Up Pass 21454 electric 2022-07-27 19:02:38 UTC
Results per page: 50
1 – 50 of 2271152
Job history
Refresh

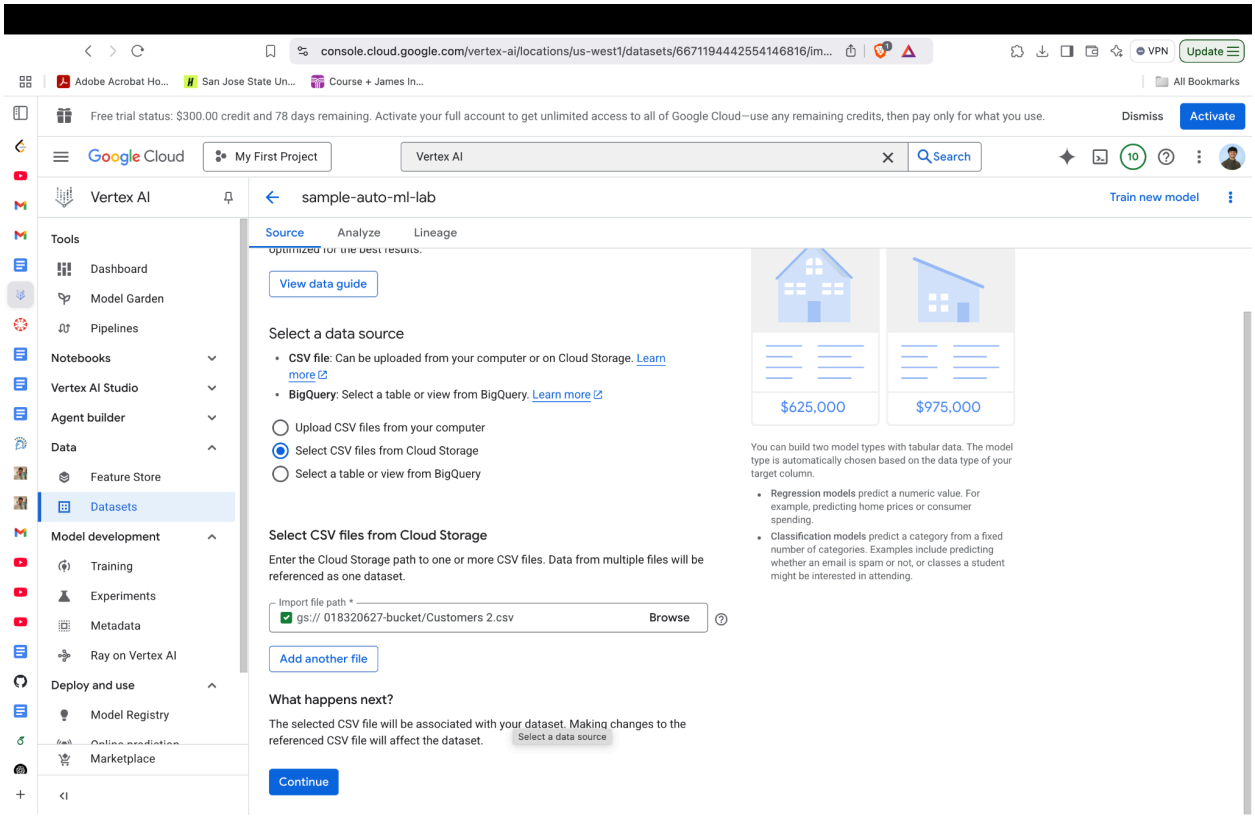


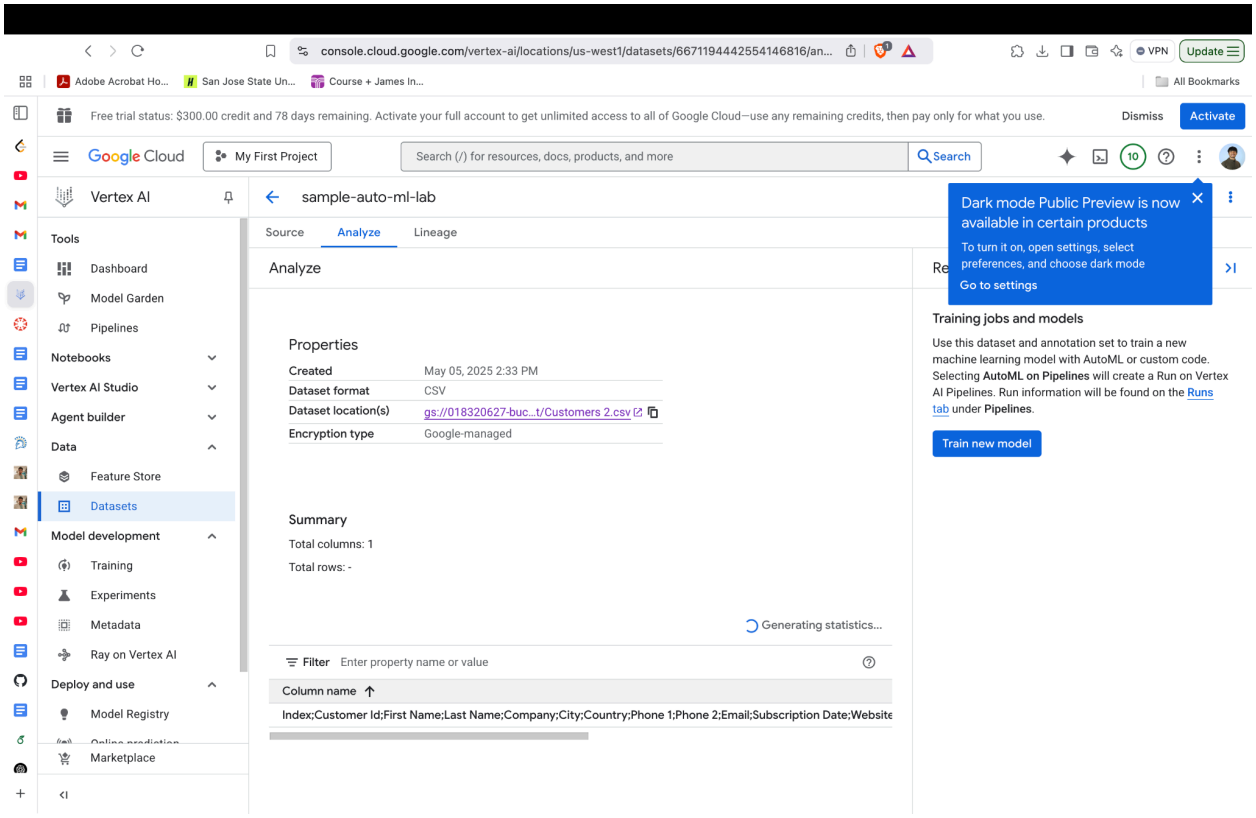
Step 4: Explore AutoML (Vertex AI)

Part A: Upload CSV to Cloud Storage

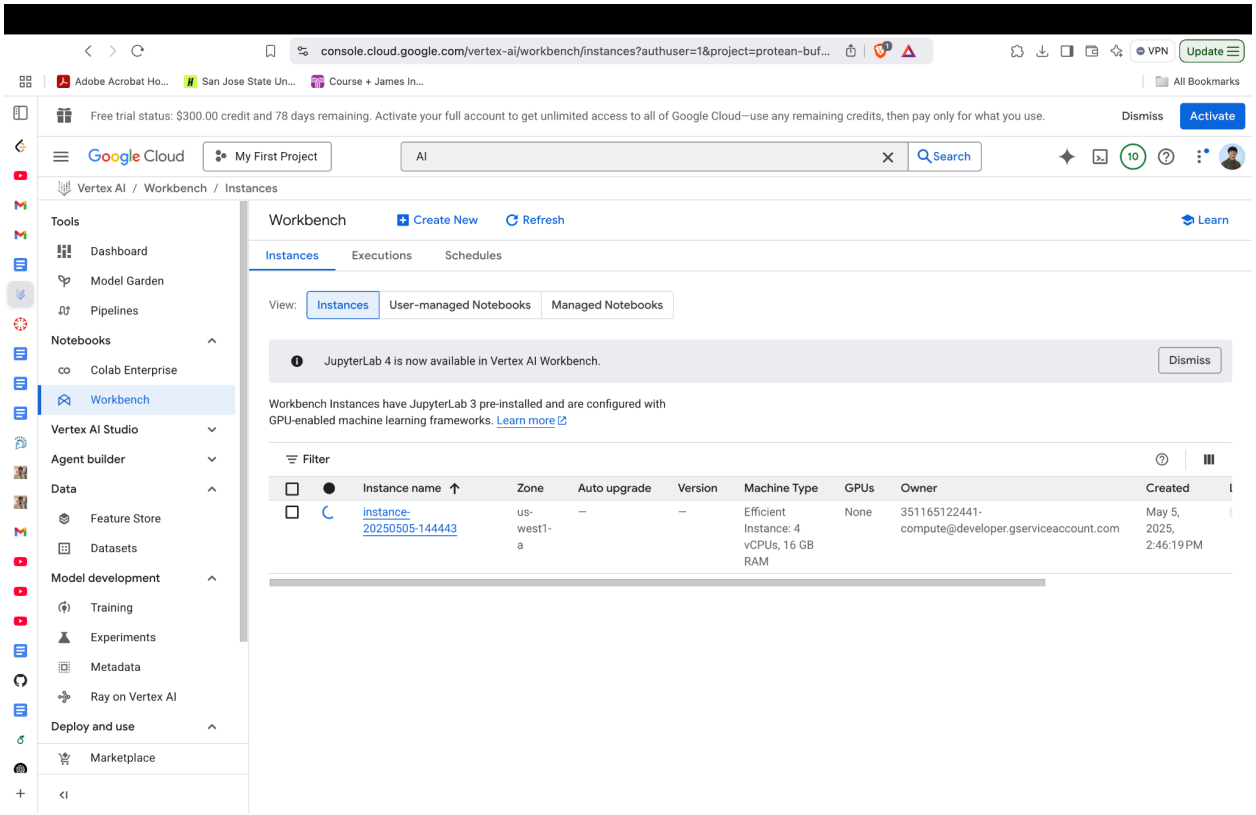

console.cloud.google.com/storage/browser/018320627-bucket;tab=objects?authuser=1&pr...
Update
Adobe Acrobat Ho...
San Jose State Un...
Course + James In...
All Bookmarks
Free trial status: $300.00 credit and 78 days remaining. Activate your full account to get unlimited access to all of Google Cloud—use any remaining credits, then pay only for what you use.
Dismiss
Activate
Google Cloud
My First Project
Search (/) for resources, docs, products, and more
Search
Cloud Storage
Bucket details
Go to path
Refresh
Overview
Buckets
Monitoring
Settings
Storage Intelligence
Insights datasets
Configuration
Marketplace
Release Notes
018320627-bucket
Location
us-west1 (Oregon)
Storage class
Standard
Public access
Not public
Protection
Soft Delete
Objects
Configuration
Permissions
Protection
Lifecycle
Observability
New
Inventory Reports
Operations
Folder browser
018320627-bucket
Buckets
018320627-bucket
Create folder
Upload
Transfer data
Other services
Learn
Filter by name prefix only
Filter
Filter objects and folders
Show
Live objects only
Name
Size
Type
Created
Storage class
Customers 2.csv
1.8 MB
text/csv
May 5, 2025, 2:25:42 PM
Standard

Part B: Open and Use Notebook

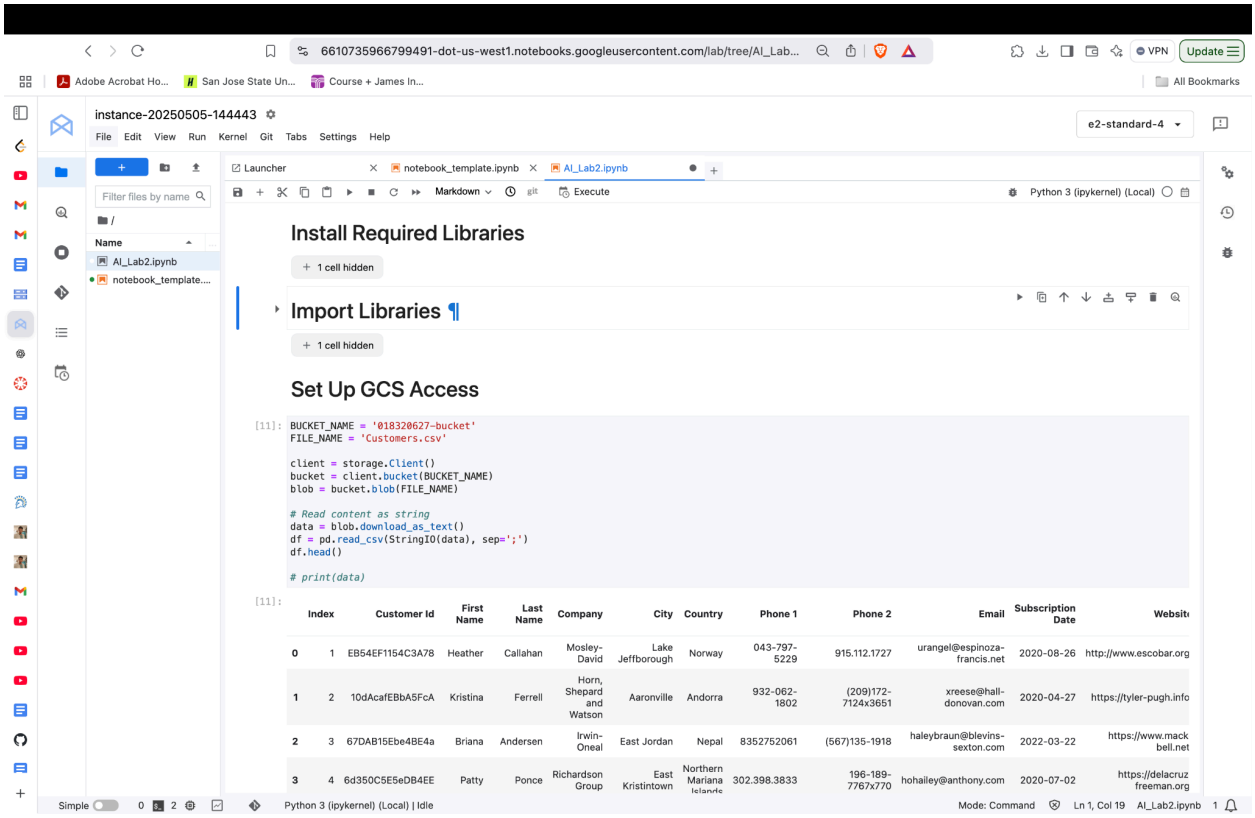


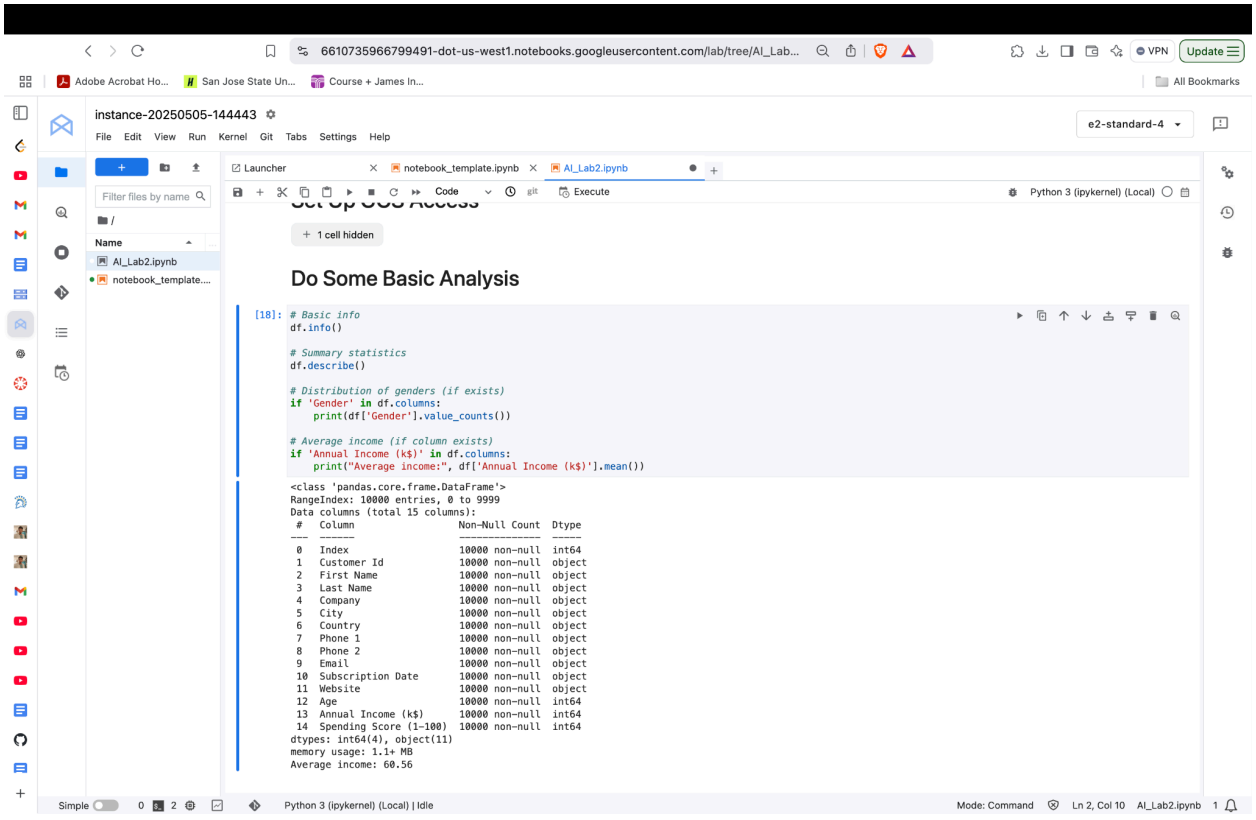


Step 5: Explore AI Platform



Read and Analyze customers.csv


6610735966799491-dot-us-west1.notebooks.googleusercontent.com/lab/tree/AI_Lab...
instance-20250505-144443
e2-standard-4
File Edit View Run Kernel Git Tabs Settings Help
Launcher
notebook_template.ipynb
AI_Lab2.ipynb
Markdown
Execute
Python 3 (ipykernel) (Local)
Filter files by name
Name
AI_Lab2.ipynb
notebook_template....
Install Required Libraries
1 cell hidden
Import Libraries ¶
1 cell hidden
Set Up GCS Access
[11]:
BUCKET_NAME = '018320627-bucket'
FILE_NAME = 'Customers.csv'
client = storage.Client()
bucket = client.bucket(BUCKET_NAME)
blob = bucket.blob(FILE_NAME)
# Read content as string
data = blob.download_as_text()
df = pd.read_csv(StringIO(data), sep=';')
df.head()
# print(data)
[11]:
Index Customer Id First Name Last Name Company City Country Phone 1 Phone 2 Email Subscription Date Website
0 1 EB54EF1154C3A78 Heather Callahan Mosley-David Lake Jeffborough Norway 043-797-5229 915.112.1727 urangel@espinoza-francis.net 2020-08-26 http://www.escobar.org
1 2 10dAcafEBbA5FcA Kristina Ferrell Horn, Shepard and Watson Aaronville Andorra 932-062-1802 (209)172-7124x3651 xreese@hall-donovan.com 2020-04-27 https://tyler-pugh.info
2 3 67DAB15Ebe4BE4a Briana Andersen Irwin-Oneal East Jordan Nepal 8352752061 (567)135-1918 haleybraun@blevins-sexton.com 2022-03-22 https://www.mack bell.net
3 4 6d350C5E5eDB4EE Patty Ponce Richardson Group East Kristintown Northern Mariana 302.398.3833 196-189-7767x770 hohailey@anthony.com 2020-07-02 https://delacruz freeman.org
Simple
Python 3 (ipykernel) (Local) | Idle
Mode: Command
Ln 1, Col 19
AI_Lab2.ipynb



Short answer Questions

What is the average value of a numeric column (e.g., income or age)?

=> The average income is 60.56 and average age is 38.85

```
[21]: print("Average income:", df['Annual Income (k$)'].mean())
      print("Average age:", df['Age'].mean())

      Average income: 60.56
      Average age: 38.85
```

How many unique values did you observe in the Gender or Category column?

=> There is no such column as ‘Gender’ or ‘Category’ in the given dataset

[24]:
```
print(df.columns)
if not "Gender" in df.columns:
    print("'Gender' column does not exists")
```

```
Index(['Index', 'Customer Id', 'First Name', 'Last Name', 'Company', 'City',
       'Country', 'Phone 1', 'Phone 2', 'Email', 'Subscription Date',
       'Website', 'Age', 'Annual Income (k$)', 'Spending Score (1-100)'],
      dtype='object')
'Gender' column does not exists
```

Were there any missing (NaN) values? Which column had the most?

=> There is no any column with missing values.

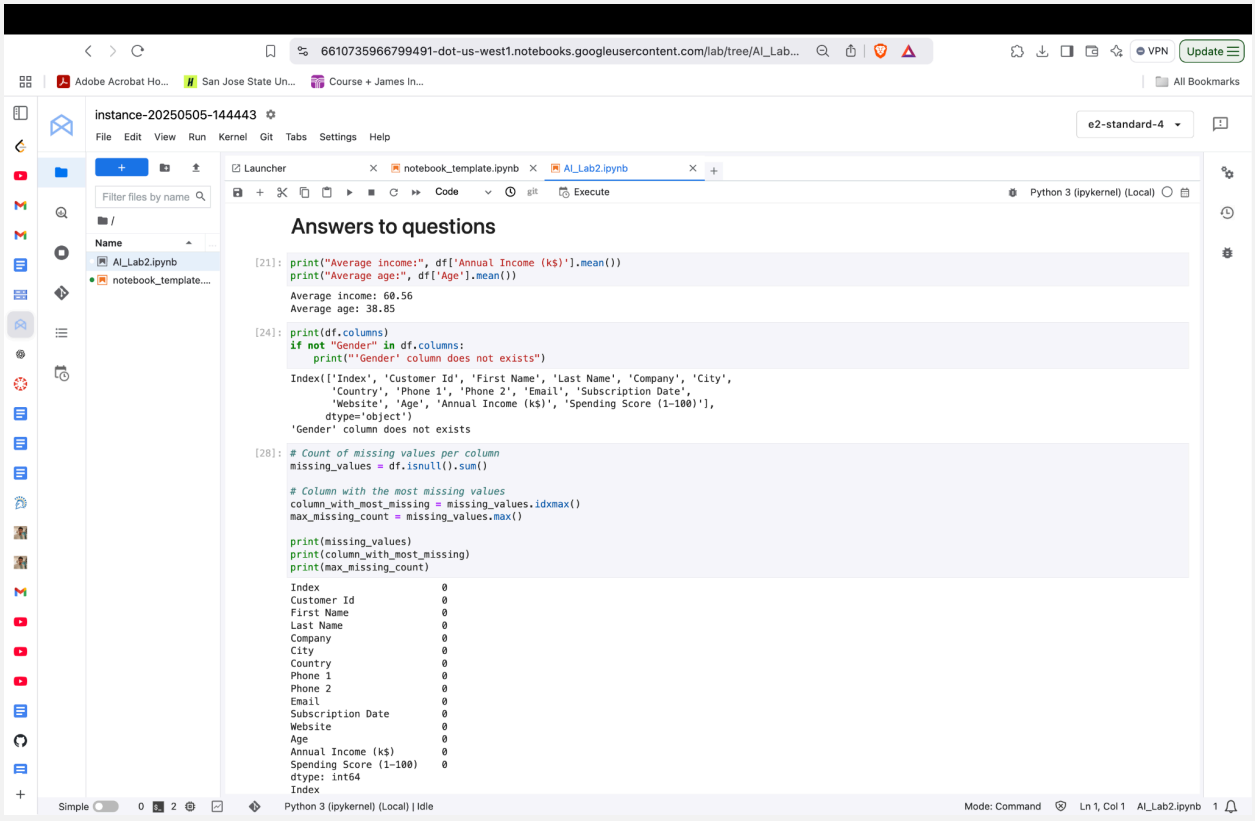
[28]:
```
# Count of missing values per column
missing_values = df.isnull().sum()

# Column with the most missing values
column_with_most_missing = missing_values.idxmax()
max_missing_count = missing_values.max()

print(missing_values)
print(column_with_most_missing)
print(max_missing_count)
```

```
Index                    0
Customer Id              0
First Name               0
Last Name                0
Company                  0
City                     0
Country                  0
Phone 1                  0
Phone 2                  0
Email                    0
Subscription Date        0
Website                  0
Age                      0
Annual Income (k$)       0
Spending Score (1-100)   0
dtype: int64
Index
0
```

Answers



Link for the notebook

https://github.com/bhusalashish/AI_LAB2/blob/main/AI_Lab2.ipynb