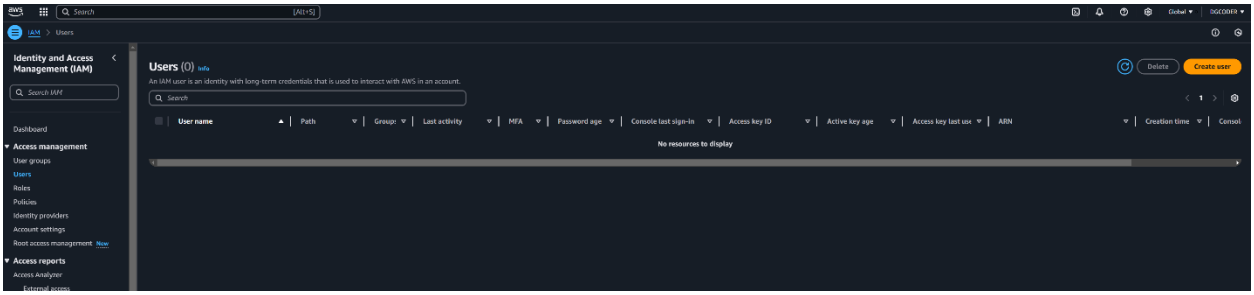
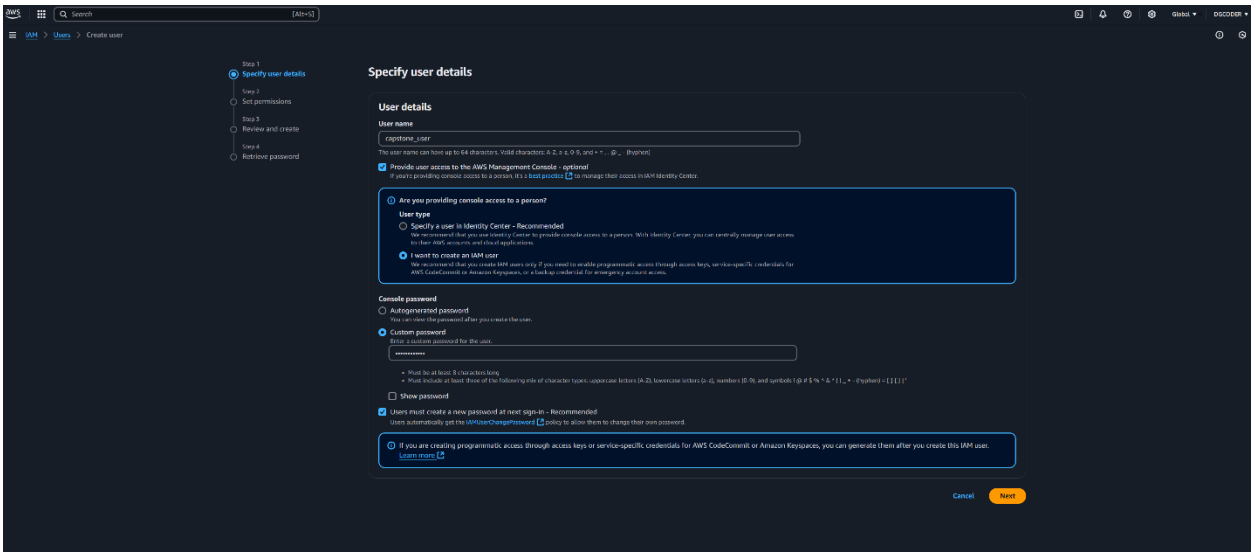
Capstone Project

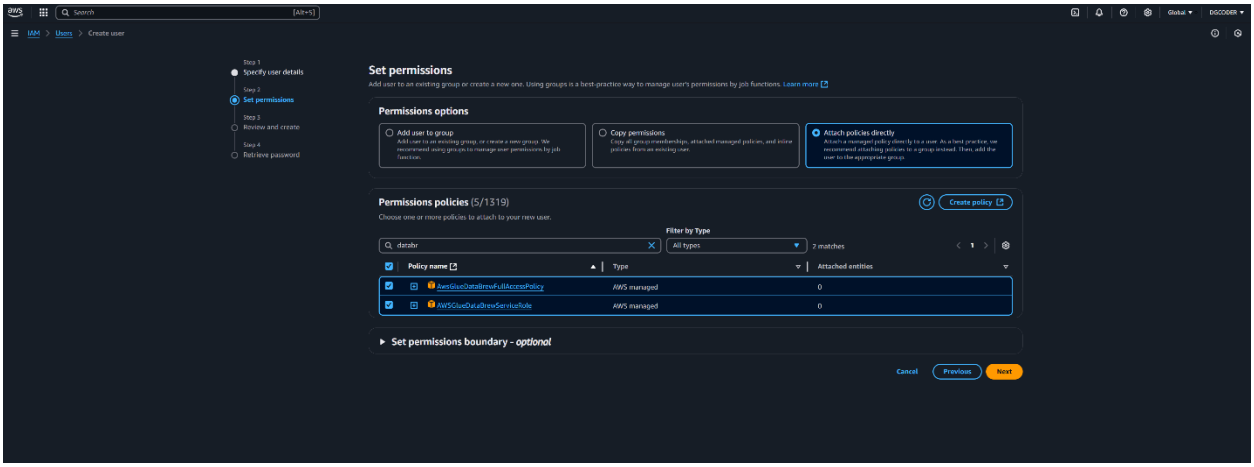
1. Creating S3 Bucket and Uploading Data

1.1 Creating a user from root user

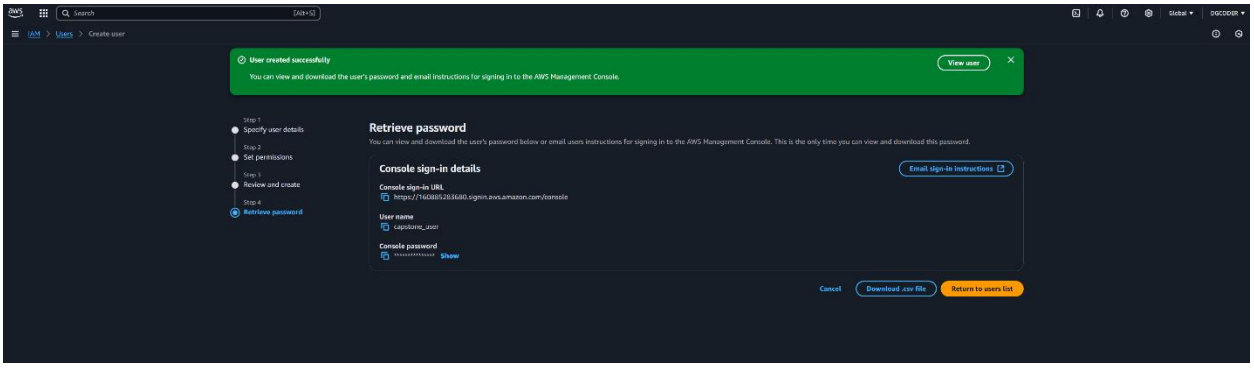




Setting up the username (**capstone_user**) and password.

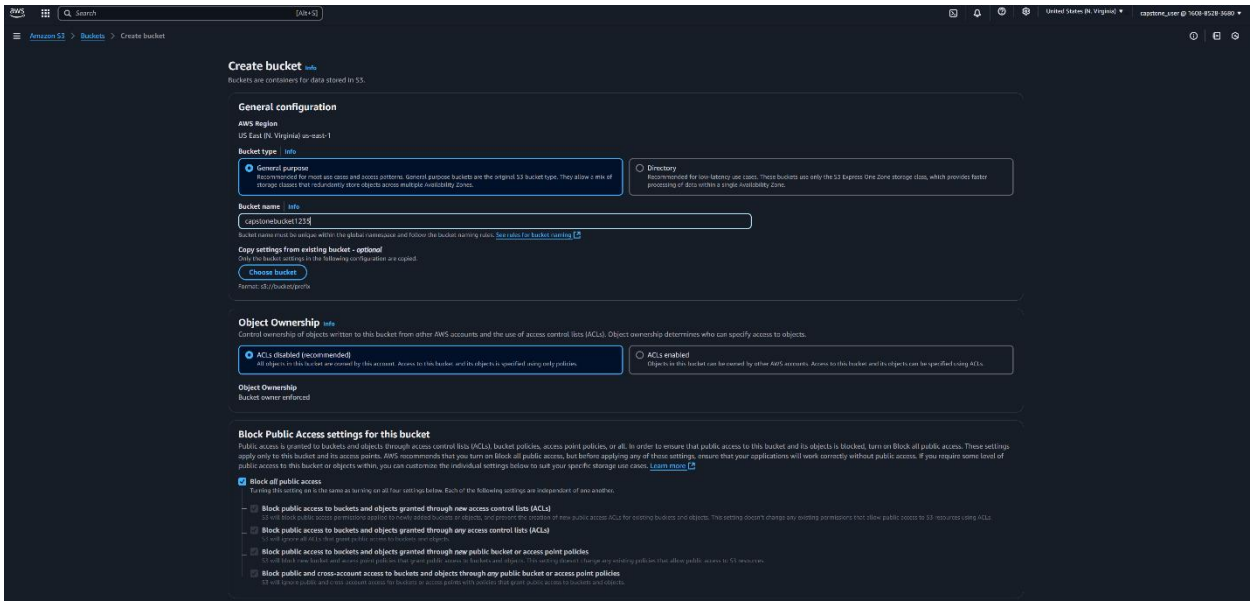


Attaching all the required policies.

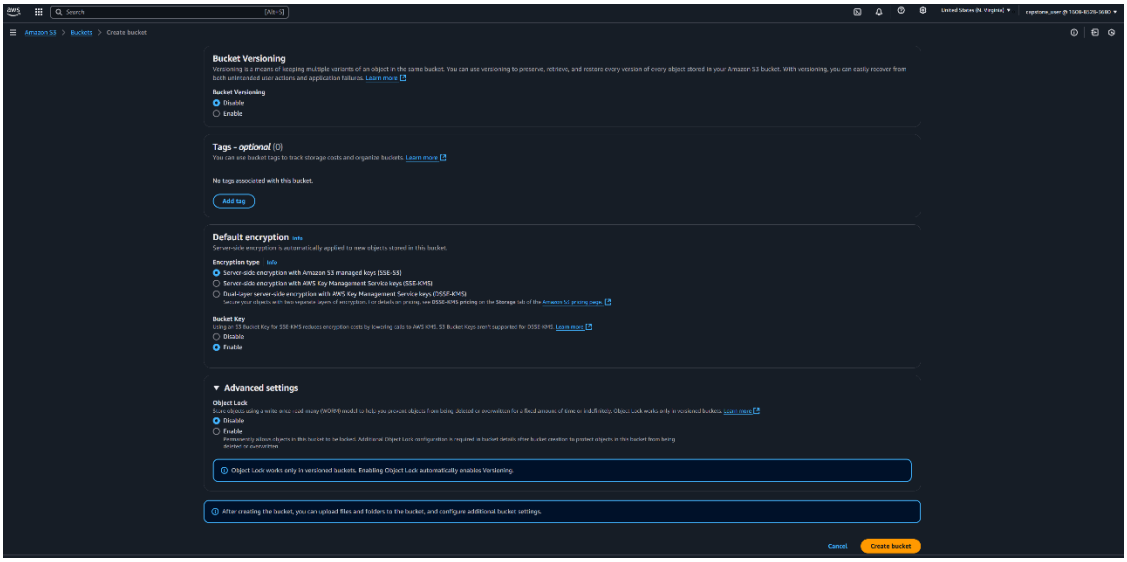


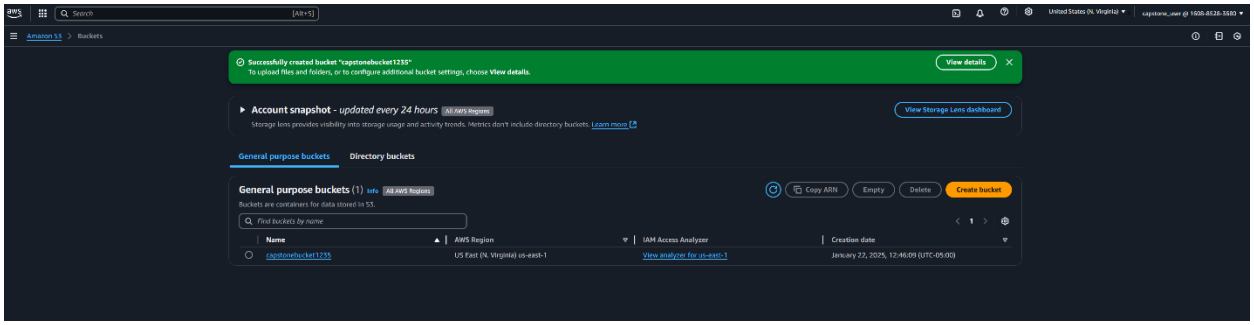
The user has been successfully created.

1.2 Creating an S3 Bucket



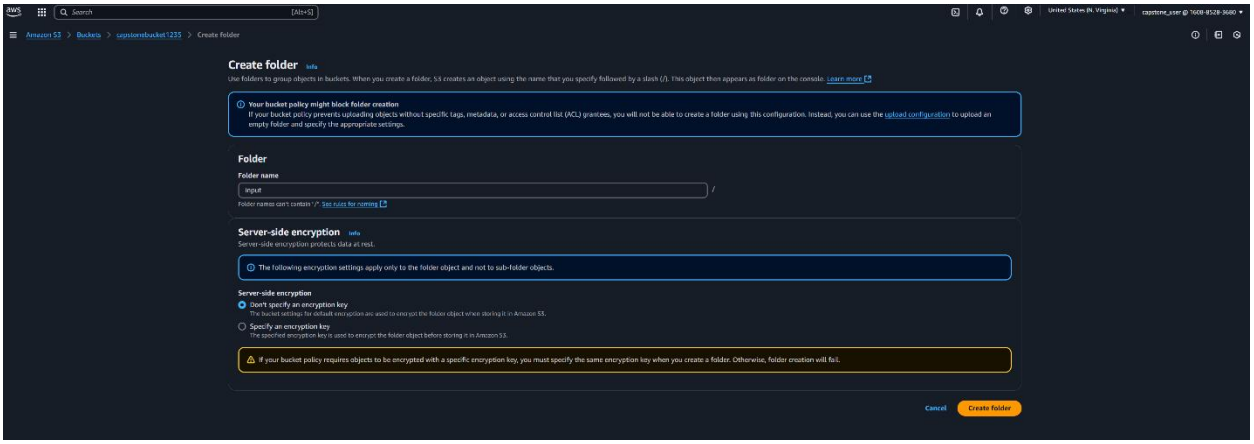
Creating an S3 bucket named (**capstonebucket1235**).

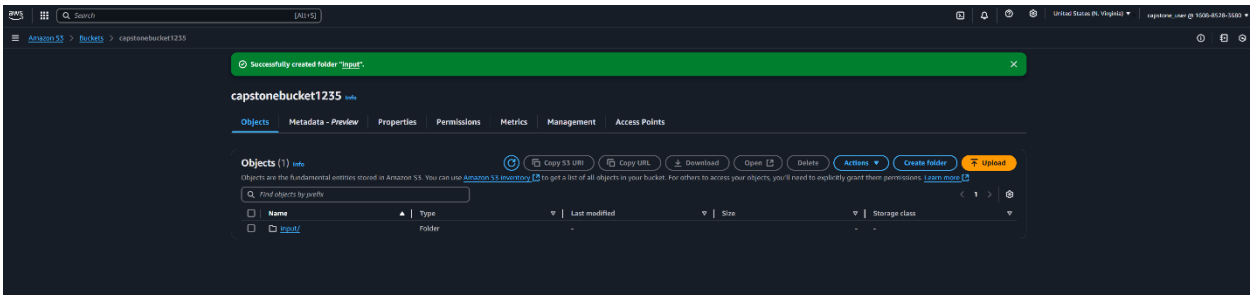




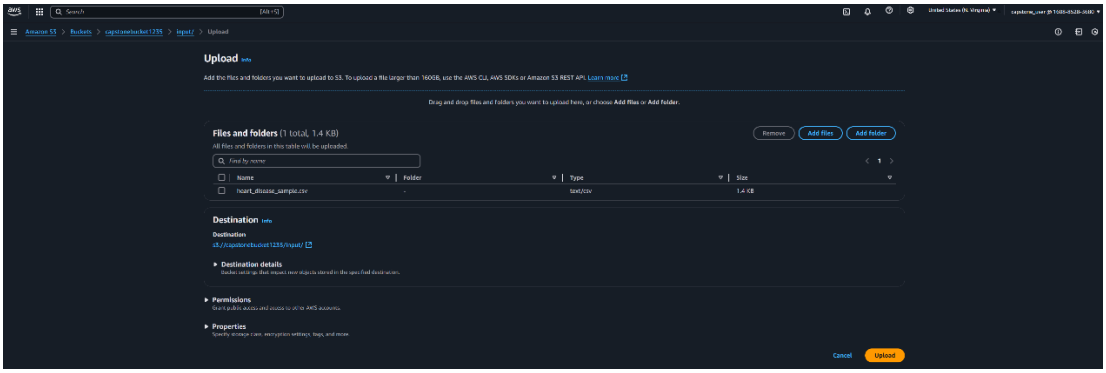
The S3 bucket(**capstonebucket1235**) has been successfully created.

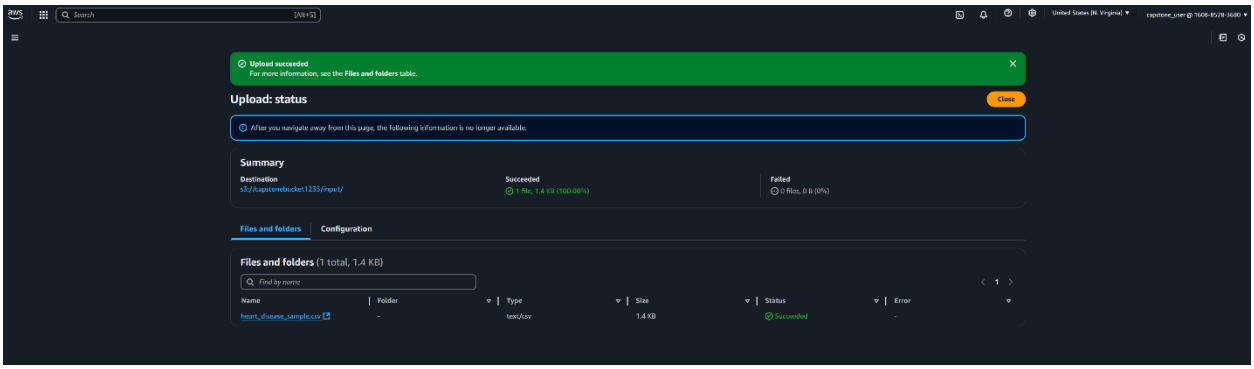
1.3 Creating an Input Container



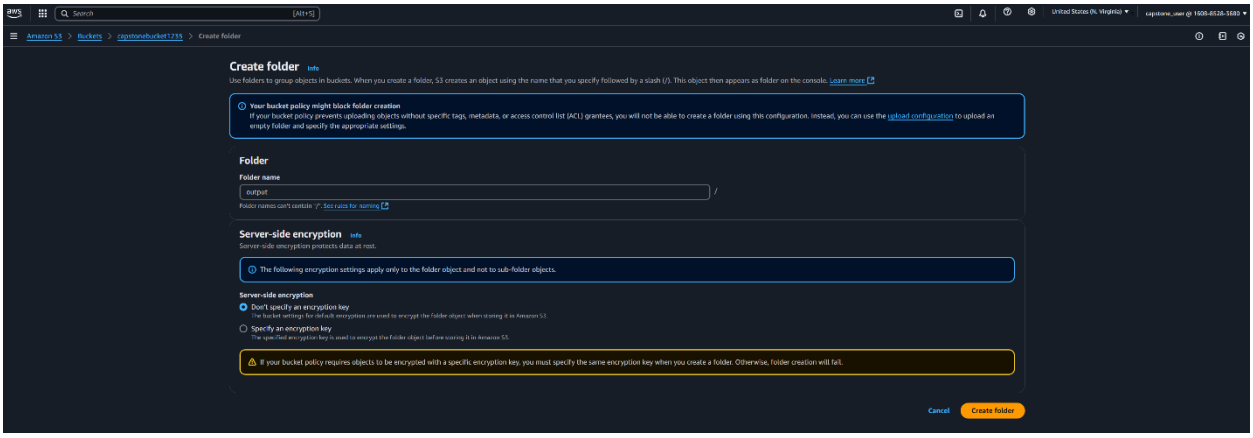


1.4 Uploading the heart disease sample csv file in input container

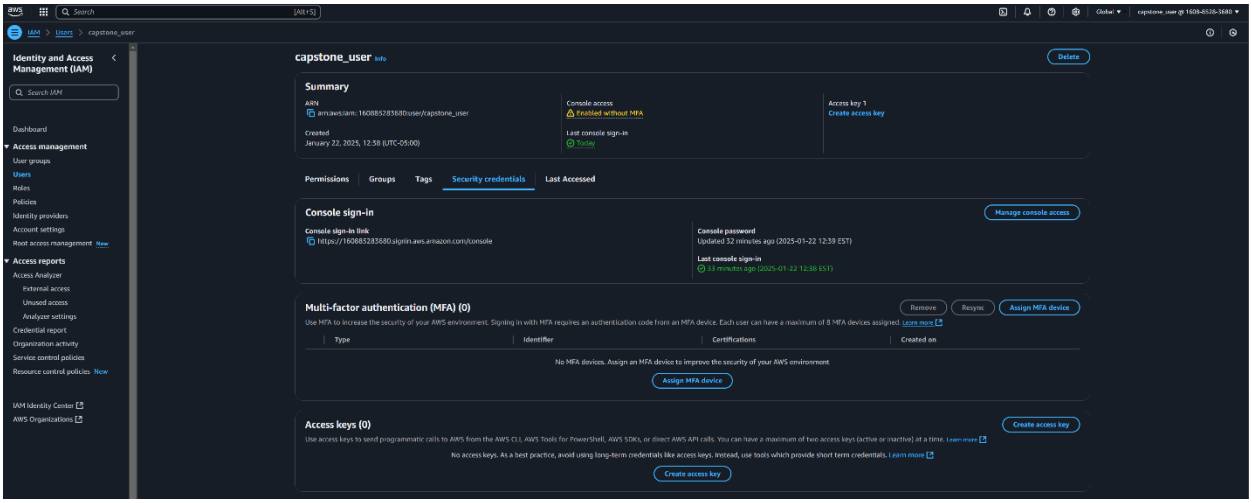


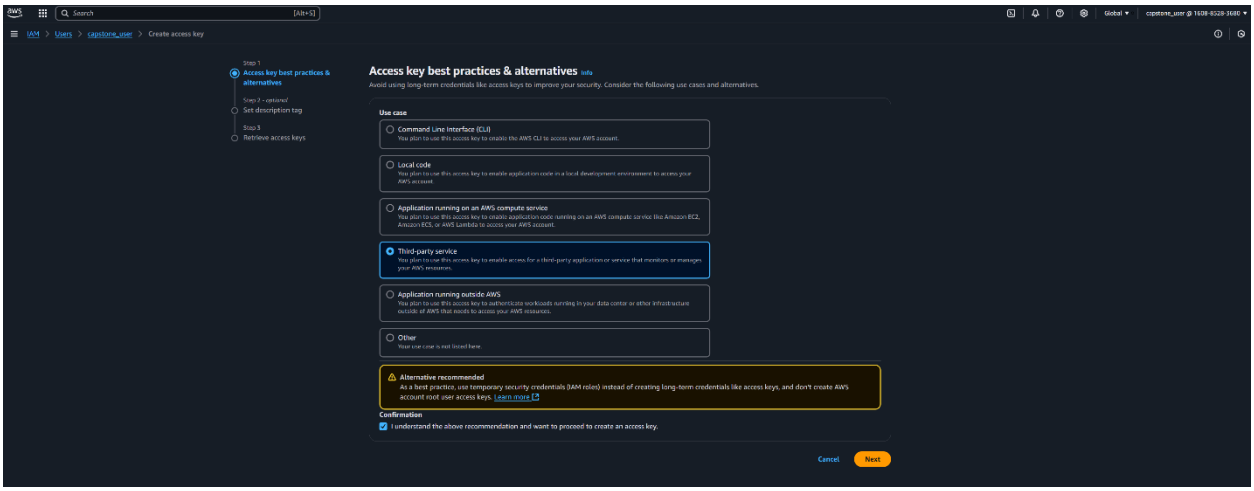


1.5 Creating an Output Container

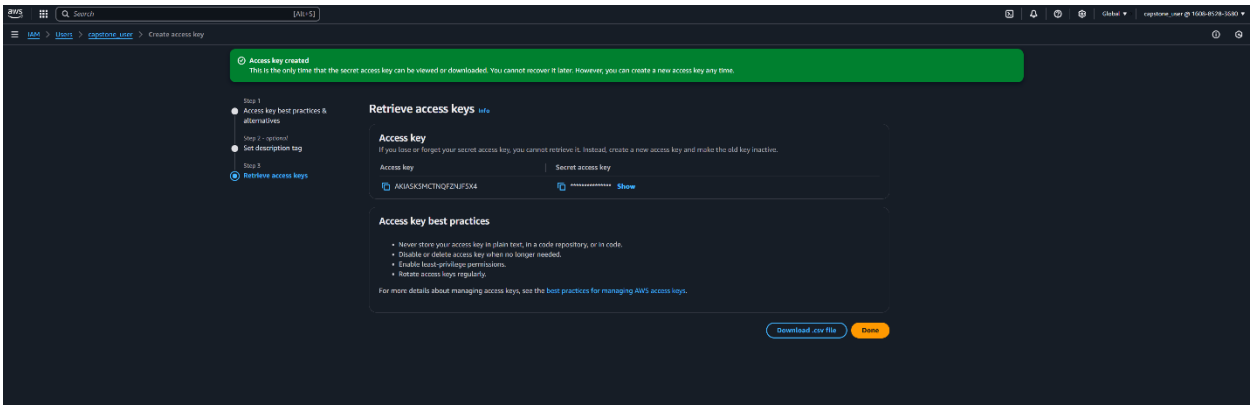


1.6 Creating the access key for mounting AWS with data bricks

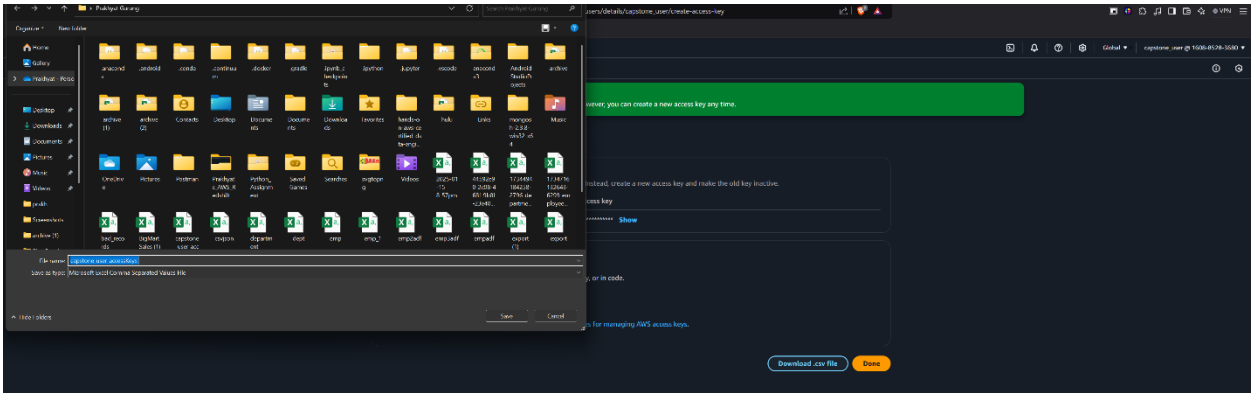




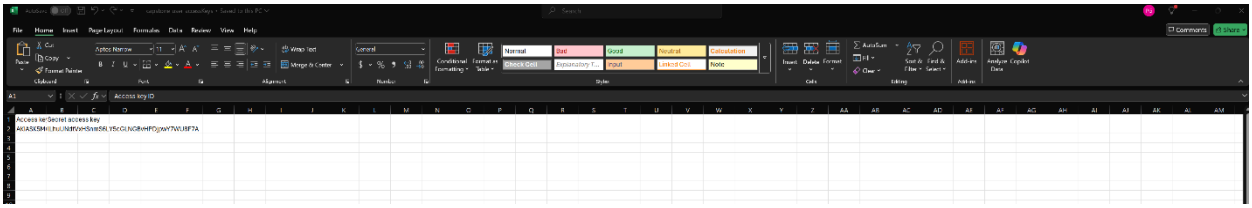
For creating access key for mounting AWS with data bricks we need to select Third-party service, so I selected the Third-party service.



The access key has been created.

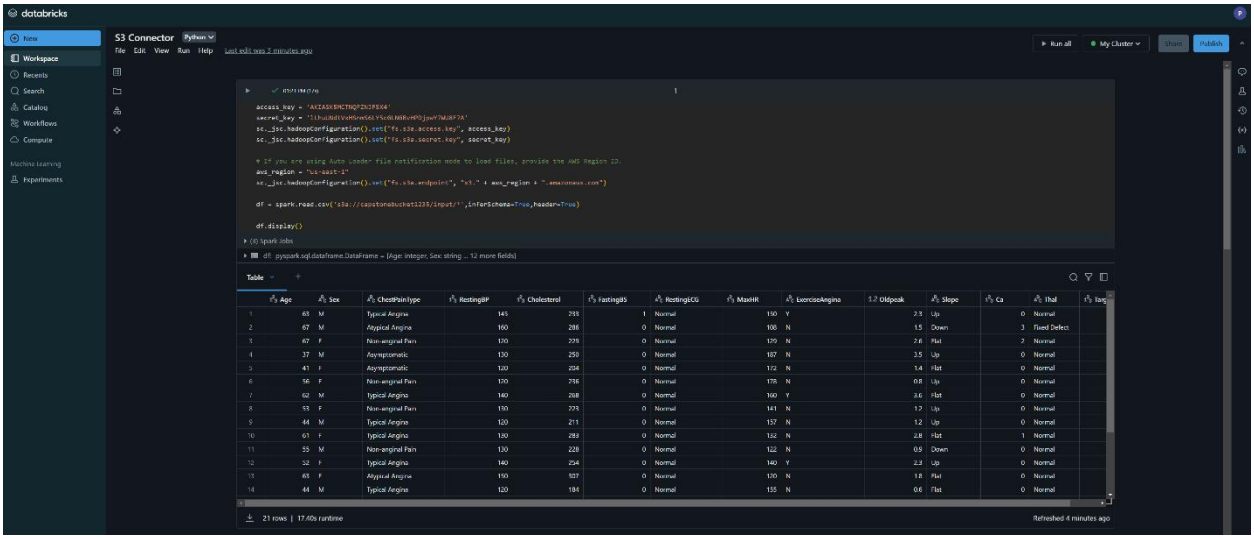


I downloaded the access key on my computer.



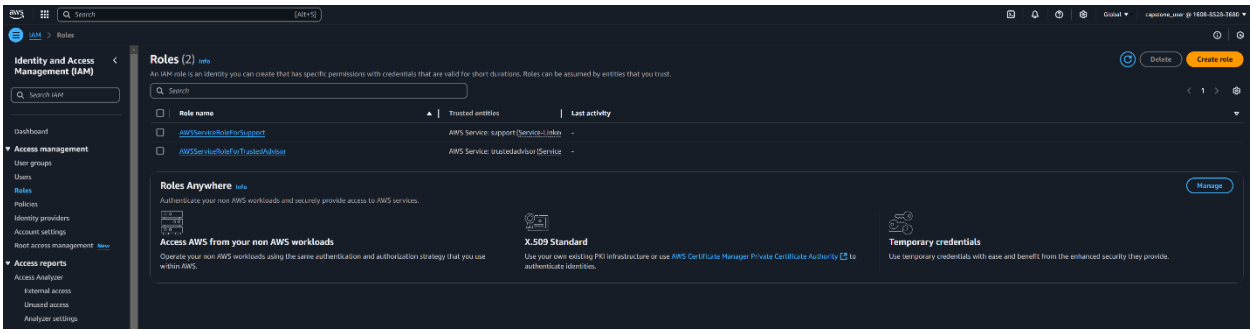
My access key and secret key is shown above on the picture.

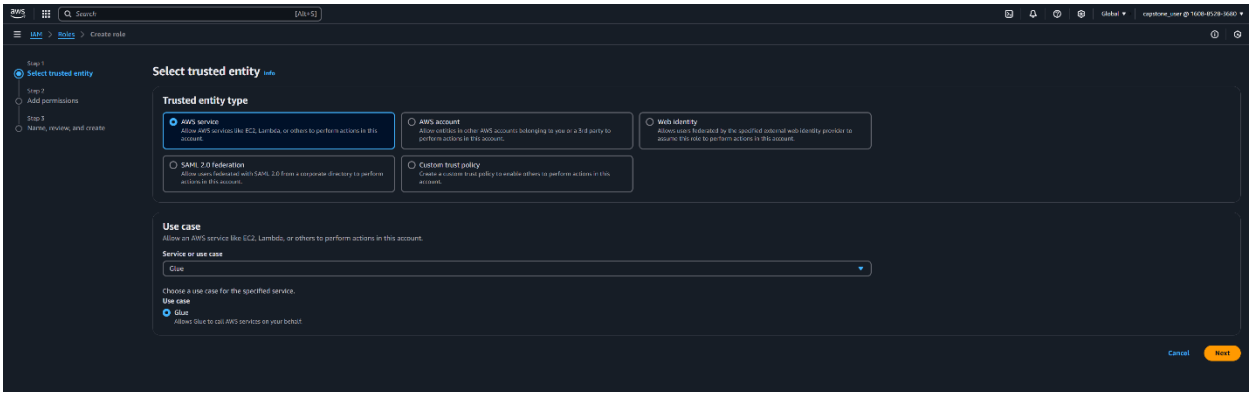
1.7 Connecting the data bricks with the access key and the secret key

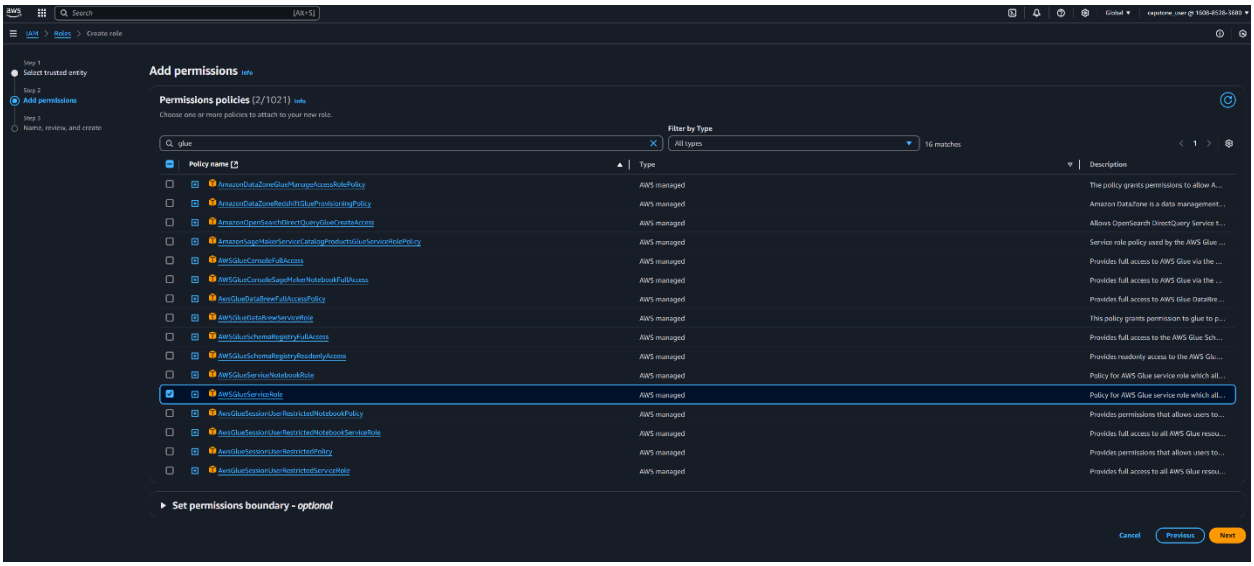


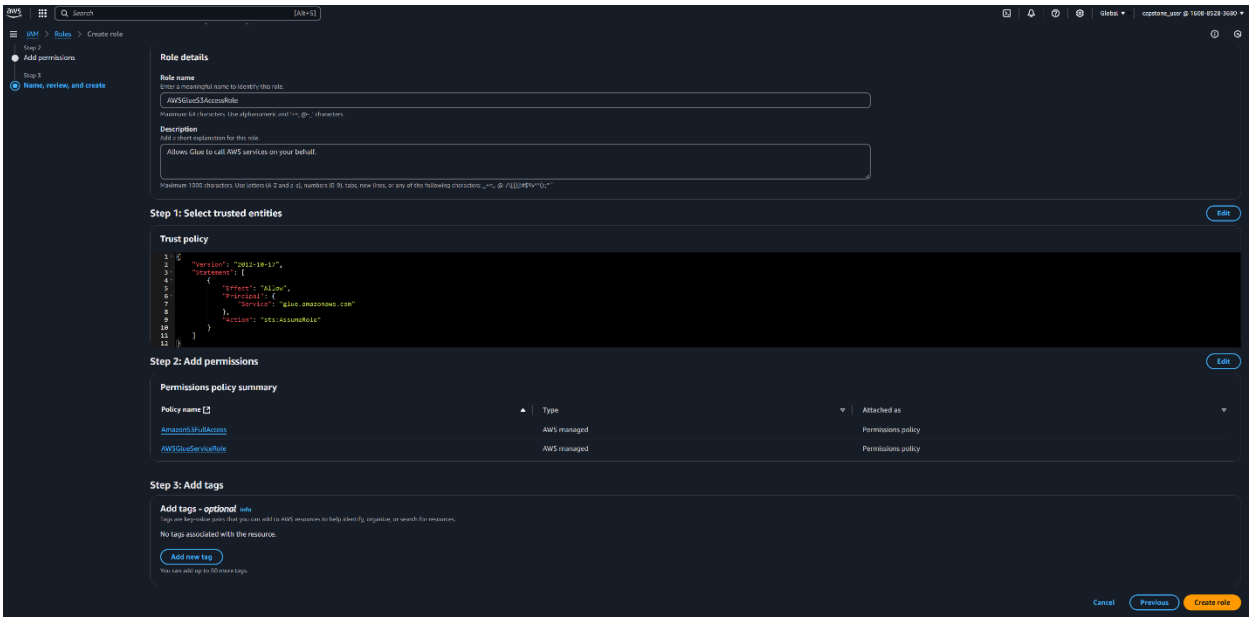
2. Setting Up IAM Roles and Policies

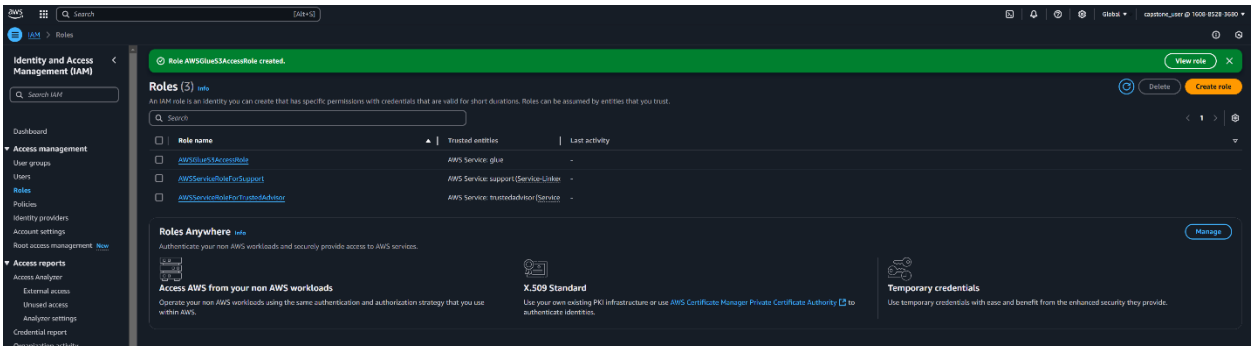
2.1 Creating IAM Roles for AWS Glue



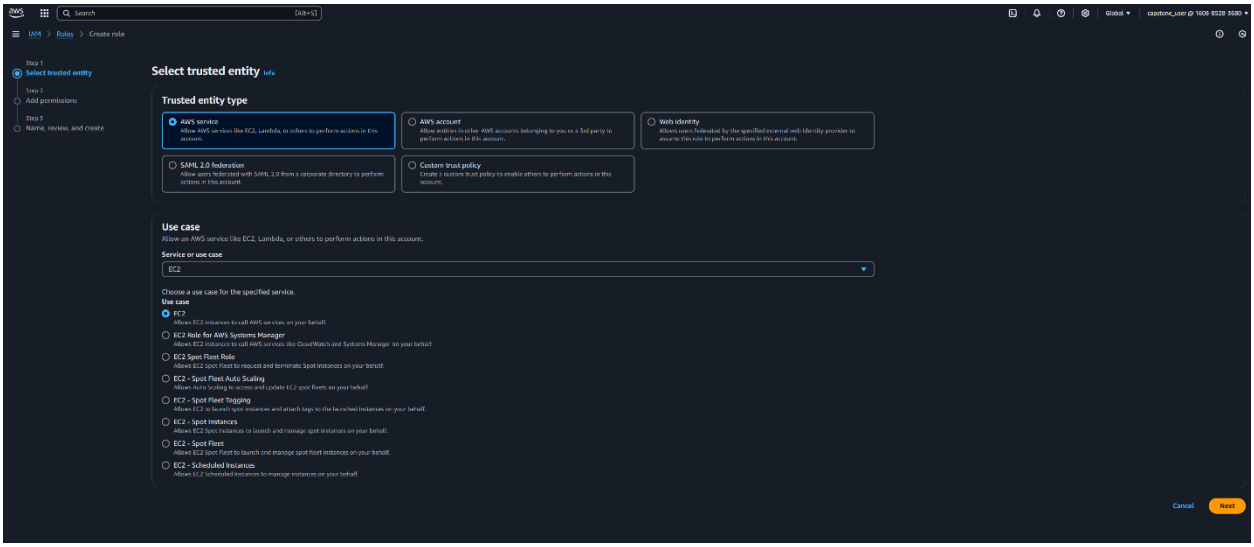


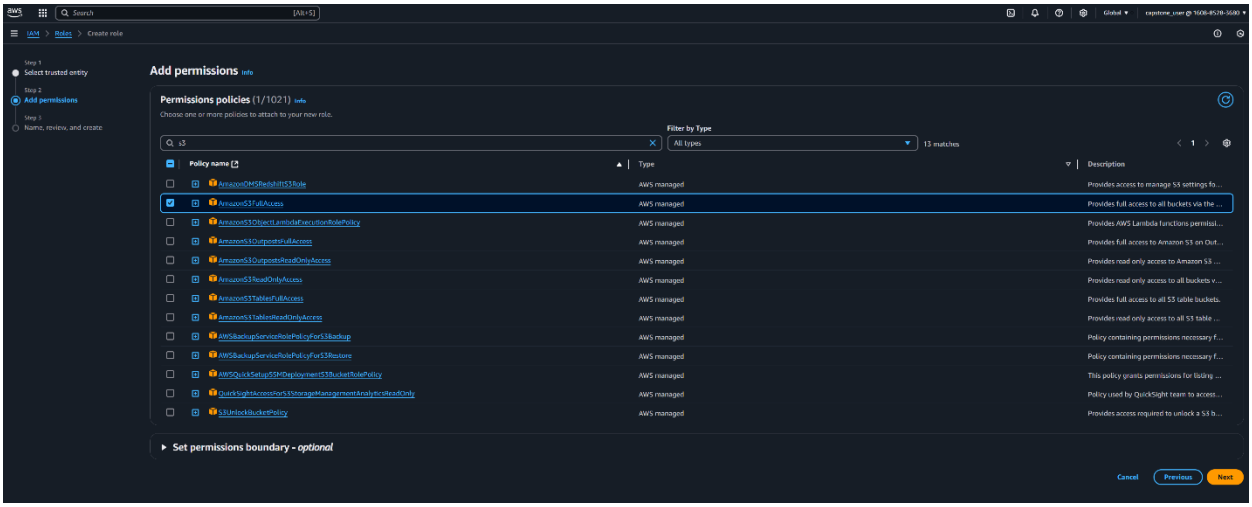


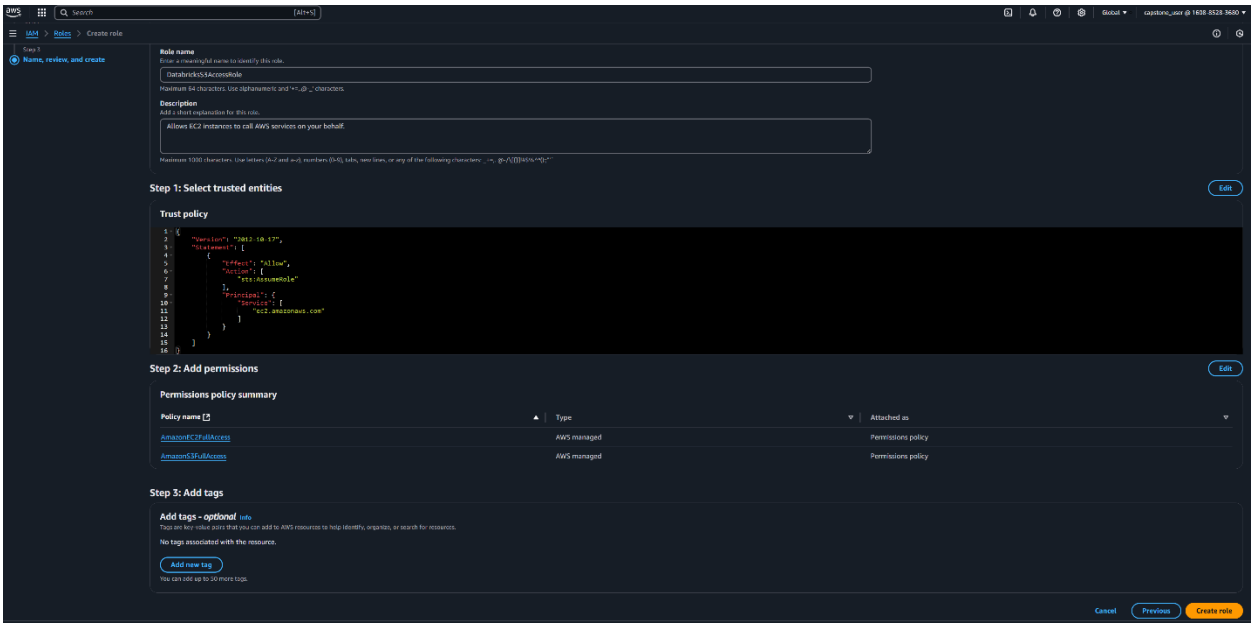




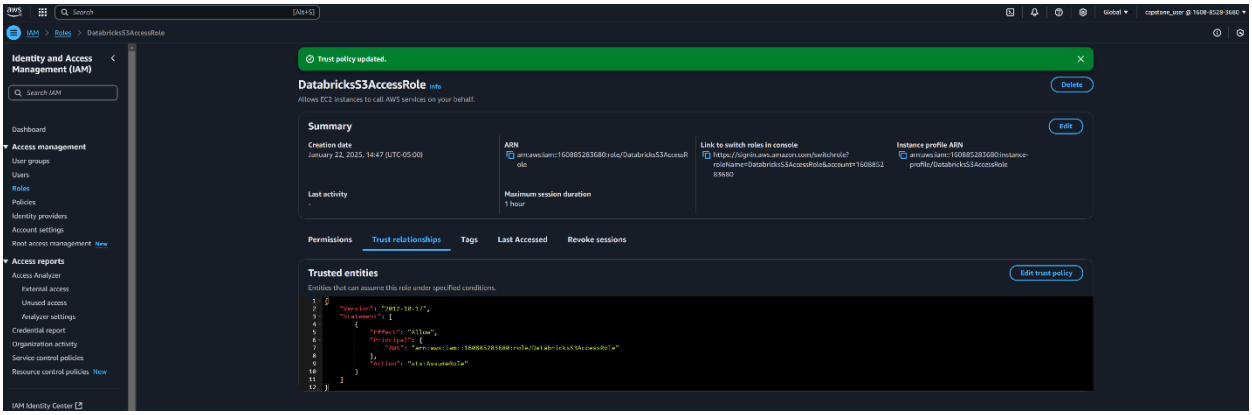
2.2 Creating IAM Roles for Databricks




IAM > Roles > Create role
Step 1
Select trusted entity
Step 2
Add permissions
Step 3
Name, review, and create
Add permissions Info
Permissions policies (1/1021) Info
Choose one or more policies to attach to your new role.
Filter by Type
s3
All types
13 matches
Policy name
Type
Description
AmazonDMSRedshiftS3Role
AWS managed
Provides access to manage S3 settings fo...
AmazonS3FullAccess
AWS managed
Provides full access to all buckets via the ...
AmazonS3ObjectLambdaExecutionRolePolicy
AWS managed
Provides AWS Lambda functions permissi...
AmazonS3OutpostsFullAccess
AWS managed
Provides full access to Amazon S3 on Out...
AmazonS3OutpostsReadOnlyAccess
AWS managed
Provides read only access to Amazon S3 ...
AmazonS3ReadOnlyAccess
AWS managed
Provides read only access to all buckets v...
AmazonS3TablesFullAccess
AWS managed
Provides full access to all S3 table buckets.
AmazonS3TablesReadOnlyAccess
AWS managed
Provides read only access to all S3 table ...
AWSBackupServiceRolePolicyForS3Backup
AWS managed
Policy containing permissions necessary f...
AWSBackupServiceRolePolicyForS3Restore
AWS managed
Policy containing permissions necessary f...
AWSQuickSetupSSMDeploymentS3BucketRolePolicy
AWS managed
This policy grants permissions for listing ...
QuickSightAccessForS3StorageManagementAnalyticsReadOnly
AWS managed
Policy used by QuickSight team to access...
S3UnlockBucketPolicy
AWS managed
Provides access required to unlock a S3 b...
Set permissions boundary - optional
Cancel
Previous
Next

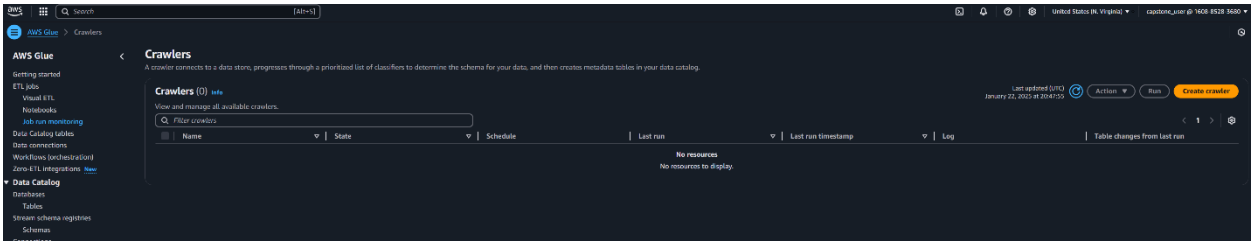

IAM > Roles > Create role
Step 3
Name, review, and create
Role name
Enter a meaningful name to identify this role.
DatabricksS3AccessRole
Maximum 64 characters. Use alphanumeric and '+=,.@-_' characters.
Description
Add a short explanation for this role.
Allows EC2 instances to call AWS services on your behalf.
Maximum 1000 characters. Use letters (A-Z and a-z), numbers (0-9), tabs, new lines, or any of the following characters: _+=,. @-/'(){}[]!#$%^*|;:"
Step 1: Select trusted entities
Edit
Trust policy
{
"Version": "2012-10-17",
"Statement": [
{
"Effect": "Allow",
"Action": [
"sts:AssumeRole"
],
"Principal": {
"Service": [
"ec2.amazonaws.com"
]
}
}
]
}
Step 2: Add permissions
Edit
Permissions policy summary
Policy name
Type
Attached as
AmazonEC2FullAccess
AWS managed
Permissions policy
AmazonS3FullAccess
AWS managed
Permissions policy
Step 3: Add tags
Add tags - optional Info
Tags are key-value pairs that you can add to AWS resources to help identify, organize, or search for resources.
No tags associated with the resource.
Add new tag
You can add up to 50 more tags.
Cancel
Previous
Create role

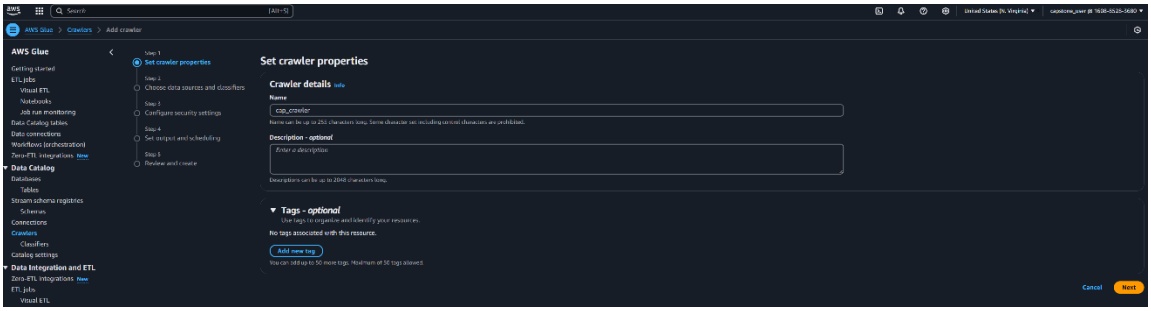
2.3 Editing the trust policy of data bricks



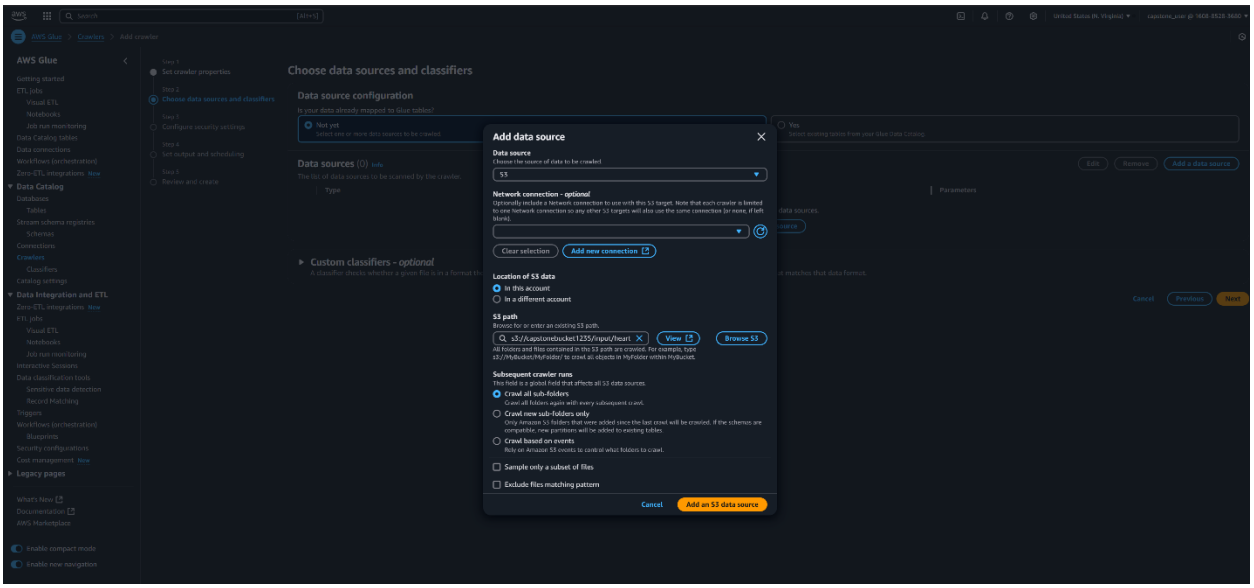
3. Setting Up AWS Glue ETL Job

3.1 Creating an AWS Glue Crawler

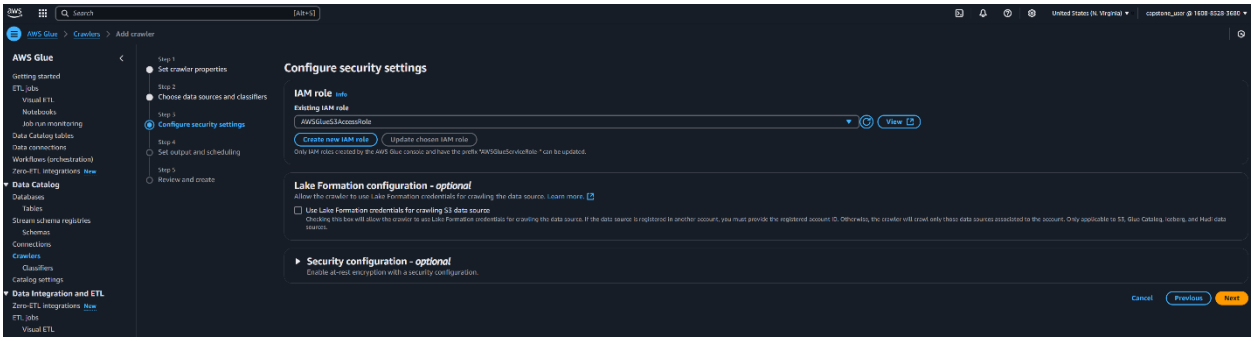




Creating an AWS Glue Crawler named (**cap_crawler**).

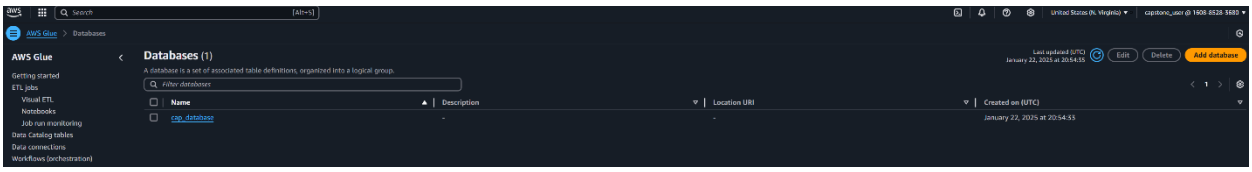


Setting up the S3 path (**s3://capstonebucket1235/input/heart_disease_sample.csv**).



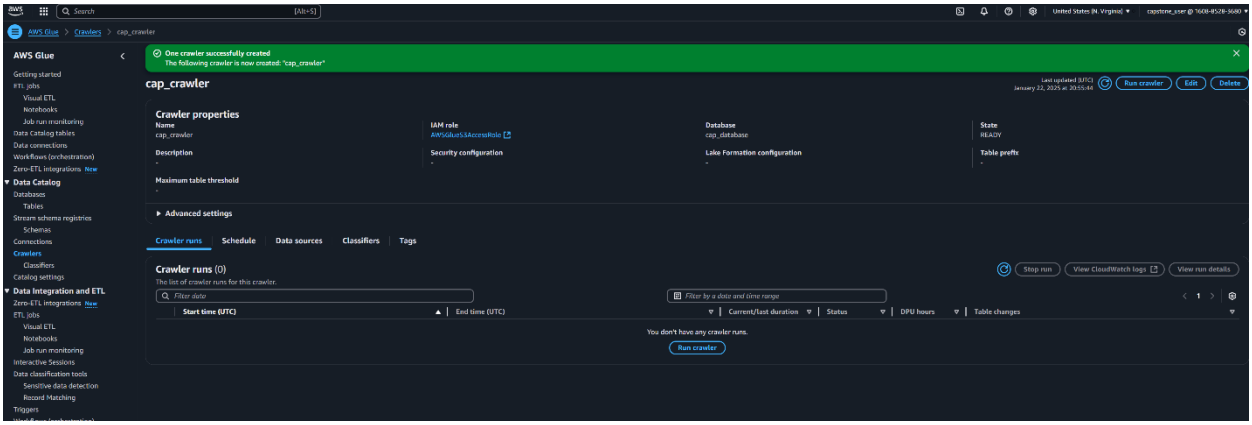
Selecting the created IAM role (**AWSGlueS3AccessRole**).

Creating the database named (**cap_database**).

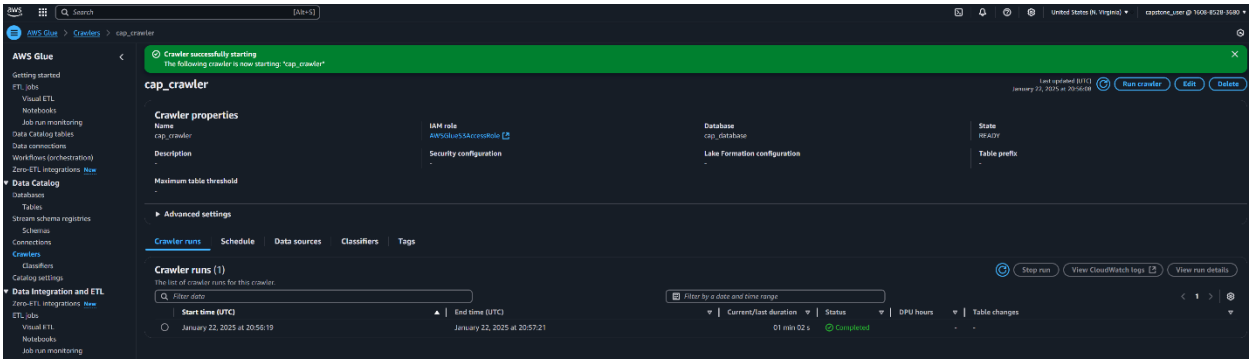


Successfully created the database.

Selecting the database named (**cap_database**).

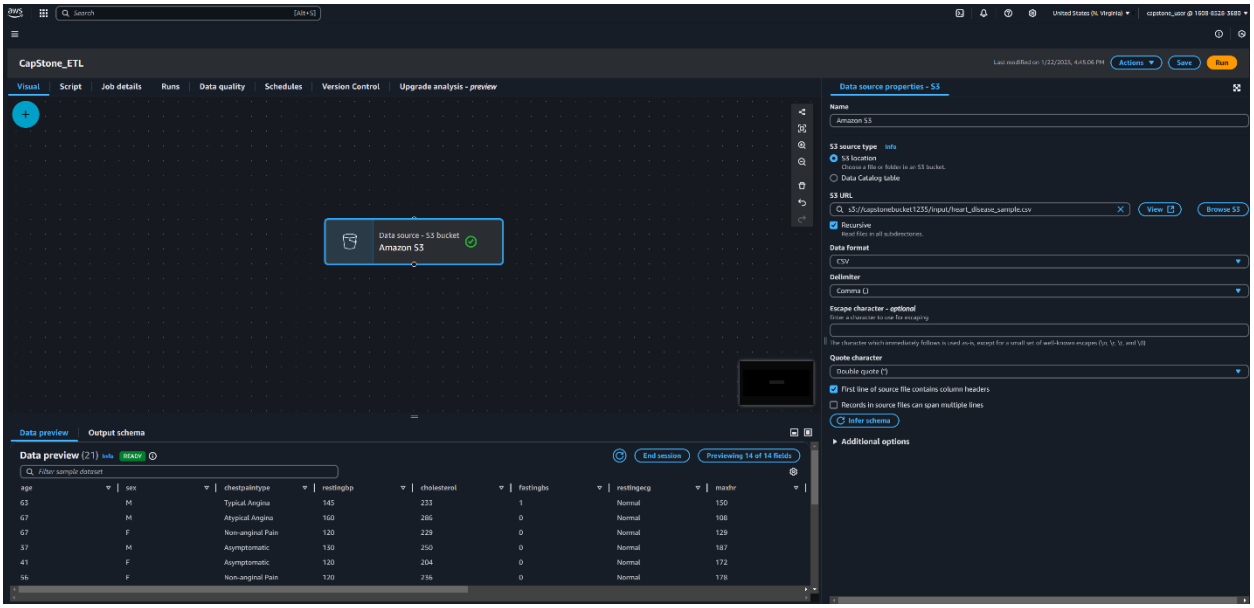


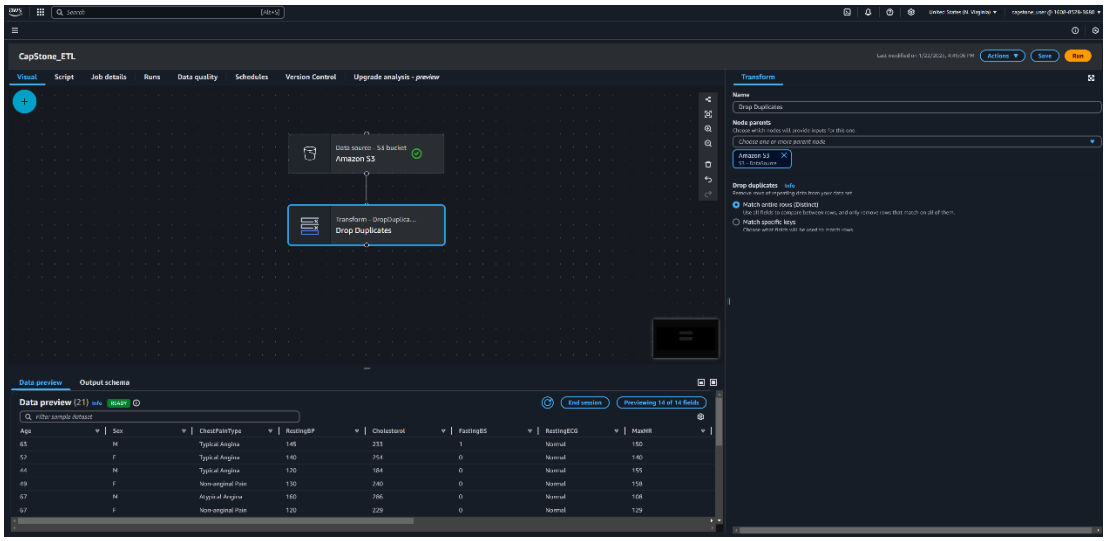
Successfully created the AWS crawler.

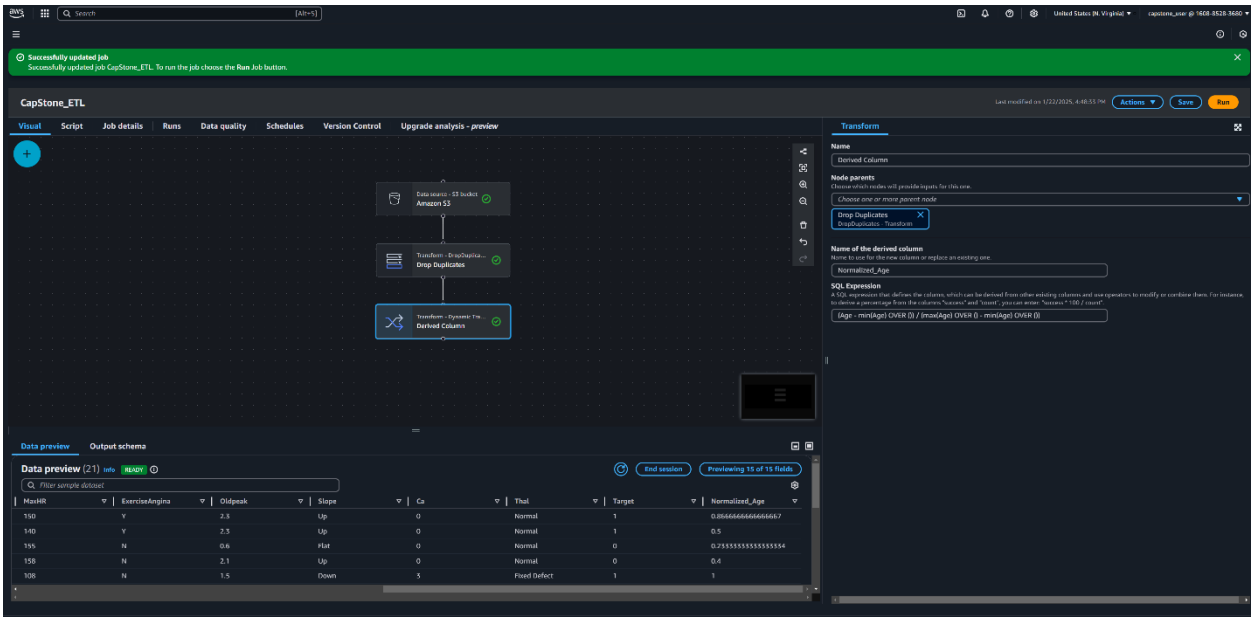


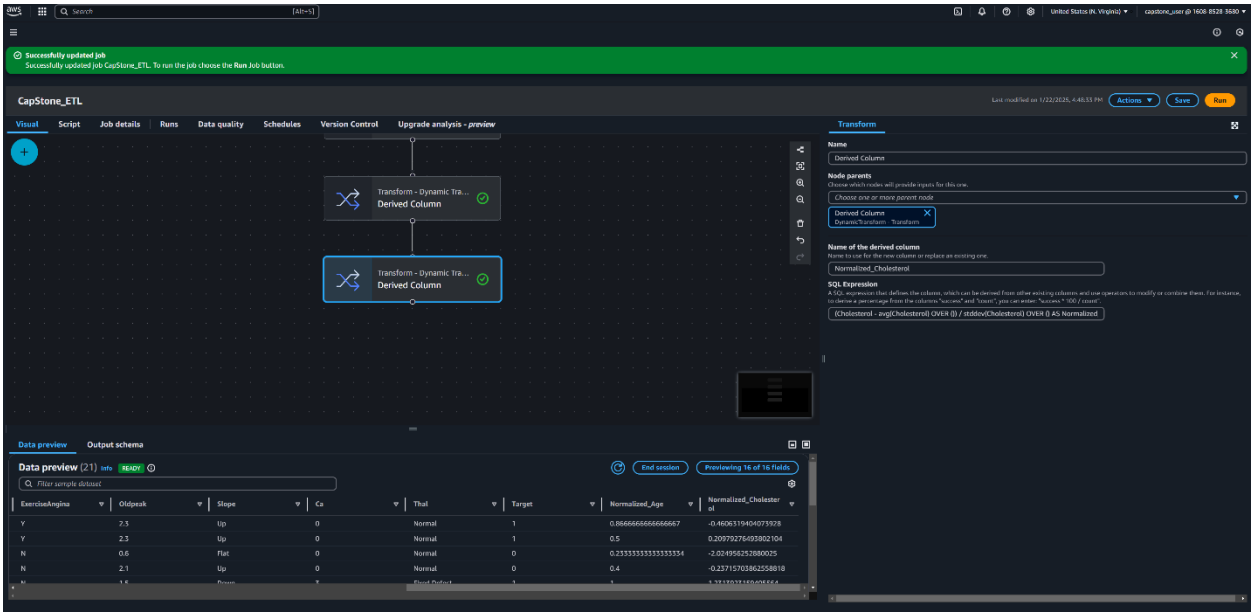
Successfully started the AWS crawler.

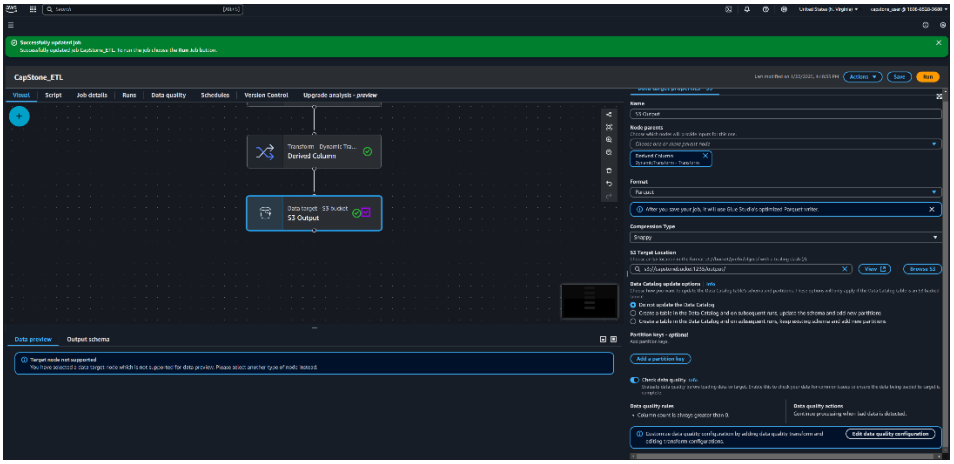
3.2 Starting the process of ETL Job in AWS Console

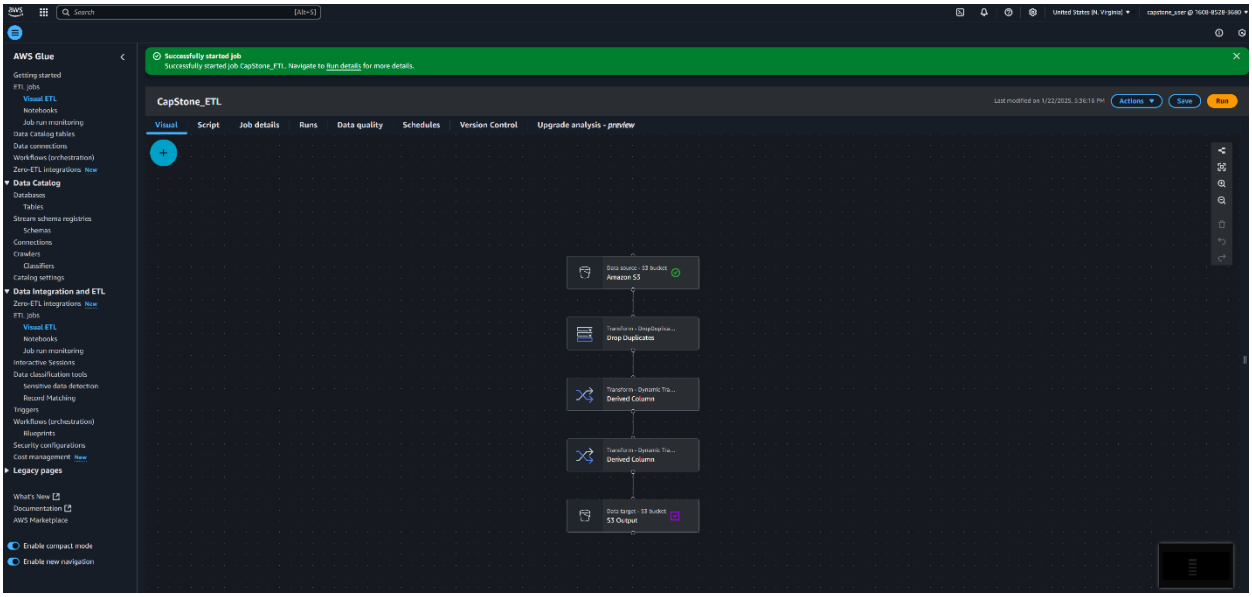


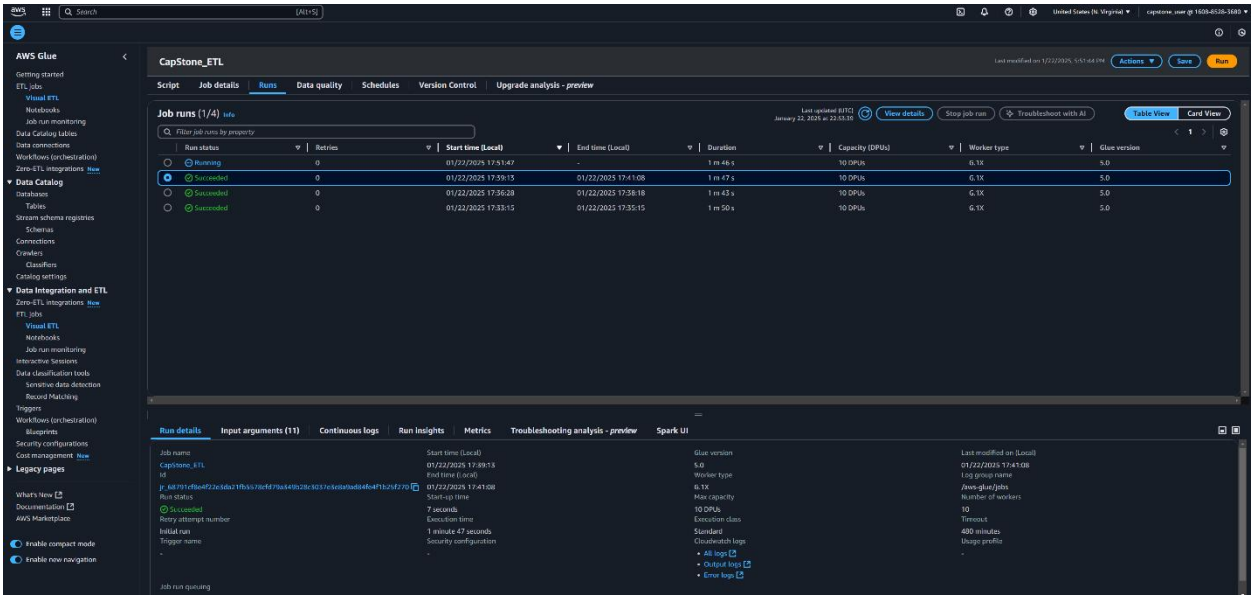


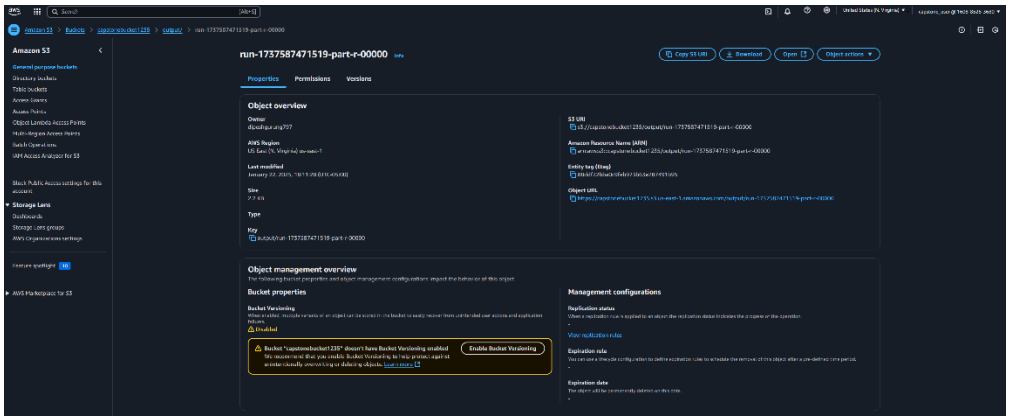

Successfully updated job
Successfully updated job CapStone_ETL. To run the job choose the Run Job button.
CapStone_ETL
Visual
Script
Job details
Runs
Data quality
Schedules
Version Control
Upgrade analysis - preview
Actions
Save
Run
Amazon S3
Drop Duplicates
Derived Column
Transform
Name
Derived Column
Node parents
Choose one or more parent node
Drop Duplicates
Name of the derived column
Normalized_Age
SQL Expression
(Age - min(Age) OVER ()) / (max(Age) OVER () - min(Age) OVER ())
Data preview
Output schema
Data preview (21)
Info
READY
End session
Previewing 15 of 15 fields
MaxHR
ExerciseAngina
Oldpeak
Slope
Ca
Thal
Target
Normalized_Age

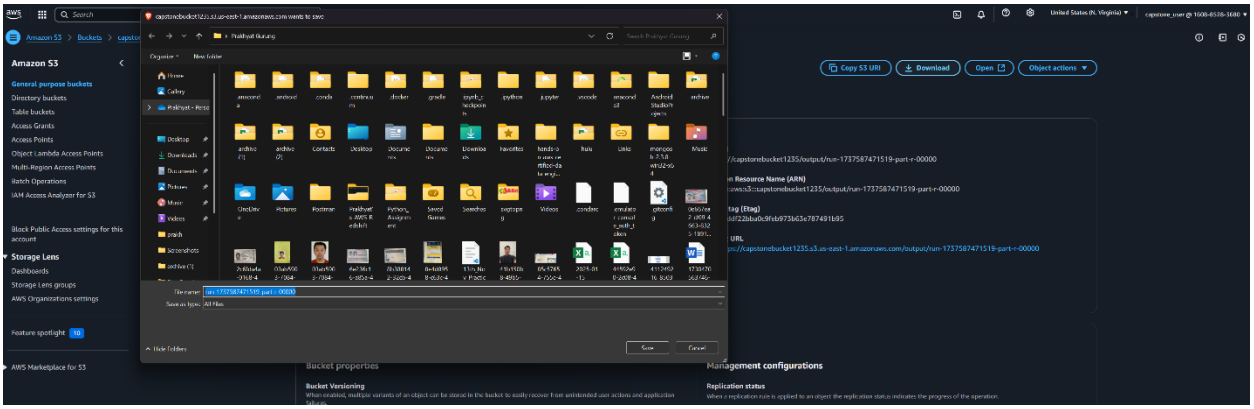

Successfully updated job
Successfully updated job CapStone_ETL. To run the job choose the Run Job button.
CapStone_ETL
Visual
Script
Job details
Runs
Data quality
Schedules
Version Control
Upgrade analysis - preview
Actions
Save
Run
Transform - Dynamic Tra...
Derived Column
Transform - Dynamic Tra...
Derived Column
Transform
Name
Derived Column
Node parents
Choose one or more parent node
Derived Column
Name of the derived column
Normalized_Cholesterol
SQL Expression
(Cholesterol - avg(Cholesterol) OVER ()) / stddev(Cholesterol) OVER () AS Normalized
Data preview
Output schema
Data preview (21)
Info
READY
End session
Previewing 16 of 16 fields
ExerciseAngina
Oldpeak
Slope
Ca
Thal
Target
Normalized_Age
Normalized_Cholesterol

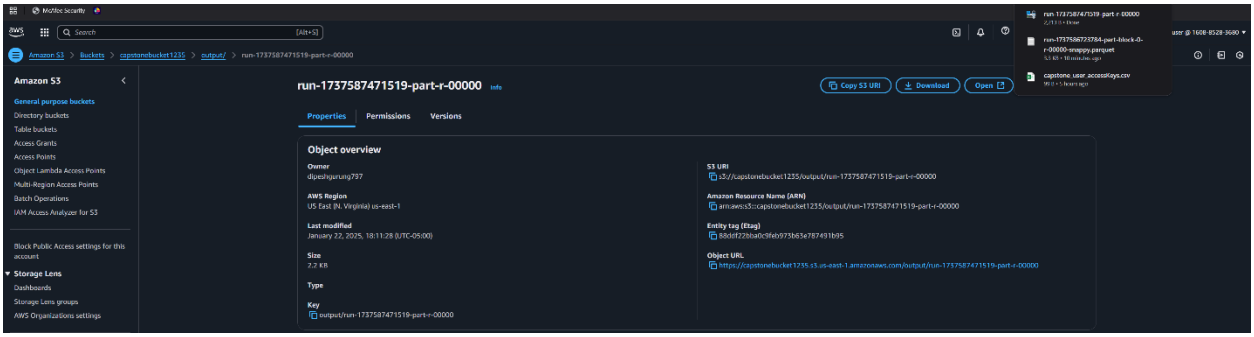


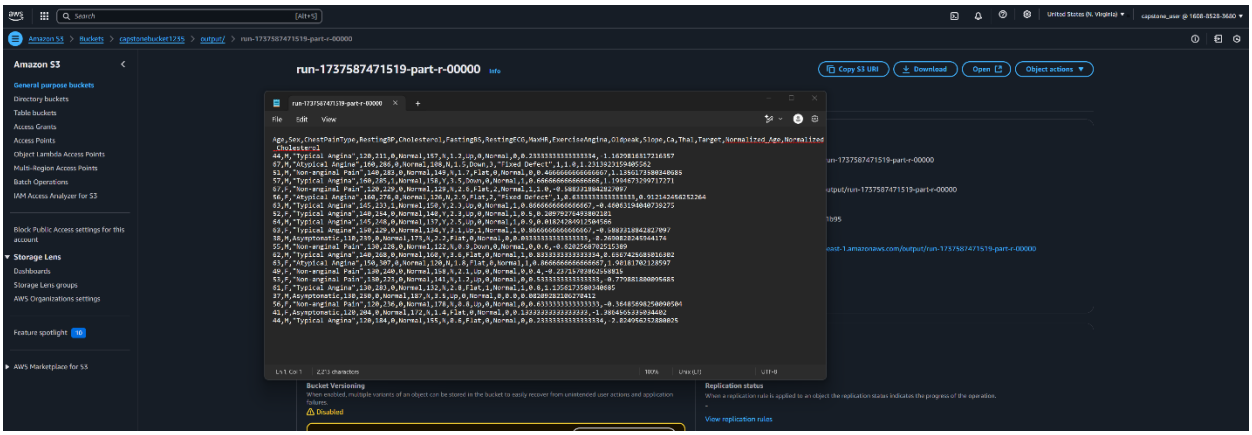




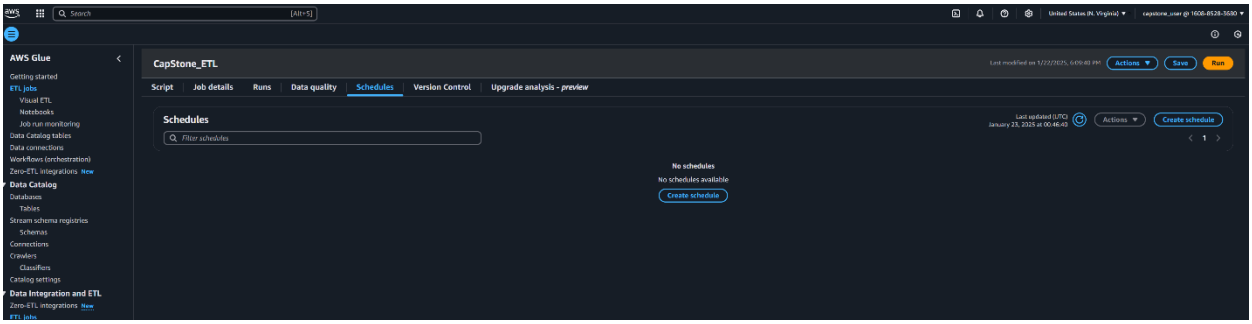

run-1737587471519-part-r-00000
Properties
Permissions
Versions
Object overview
Object management overview
Bucket properties
Management configurations

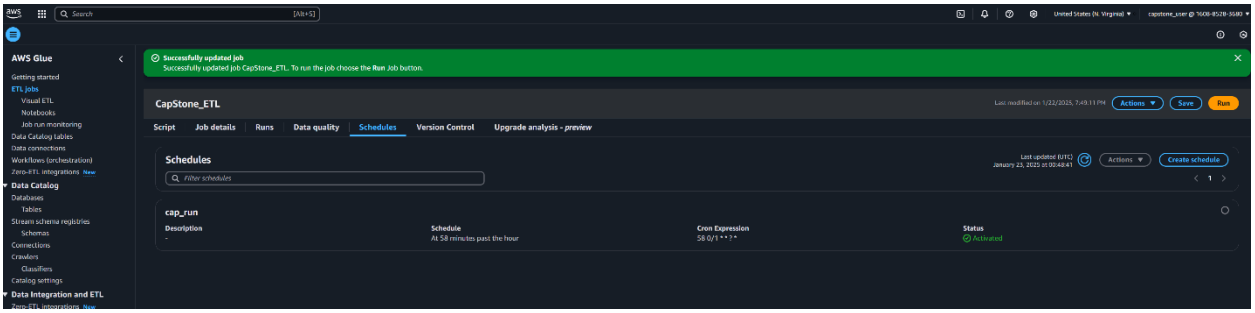

Amazon S3
Copy S3 URI
Download
Open
Object actions


run-1737587471519-part-r-00000
Properties
Permissions
Versions
Object overview

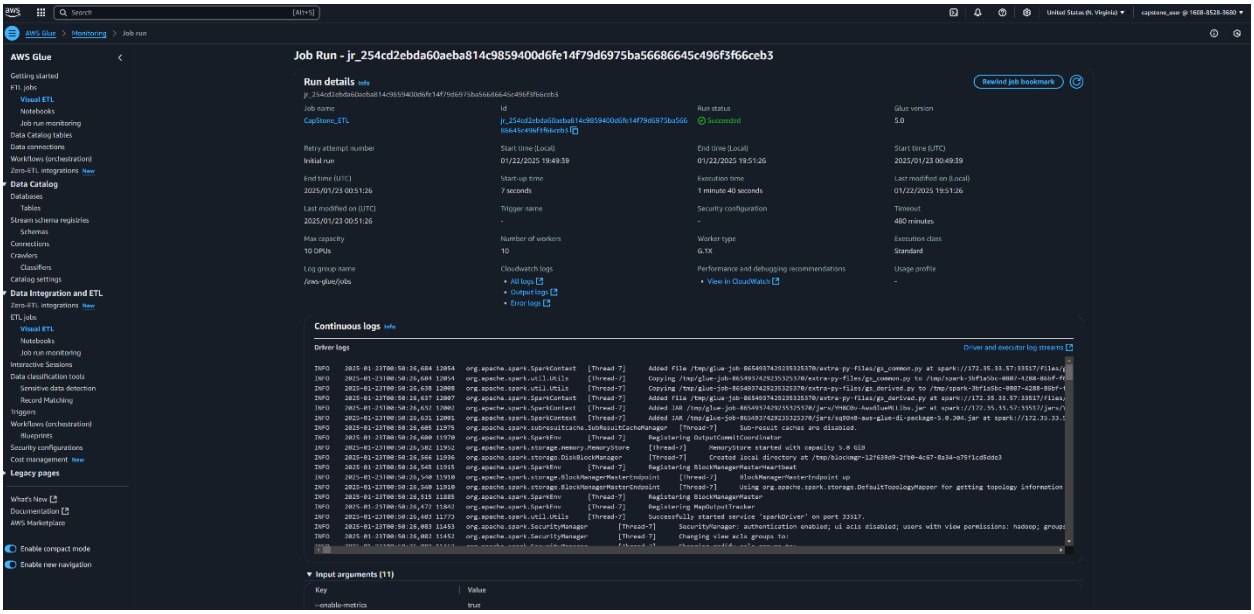

run-1737587471519-part-r-00000
Age,Sex,ChestPainType,RestingBP,Cholesterol,FastingBS,RestingECG,MaxHR,ExerciseAngina,Oldpeak,Slope,Ca,Thal,Target,Normalized_Age,Normalized_Cholesterol
Bucket Versioning
Replication status

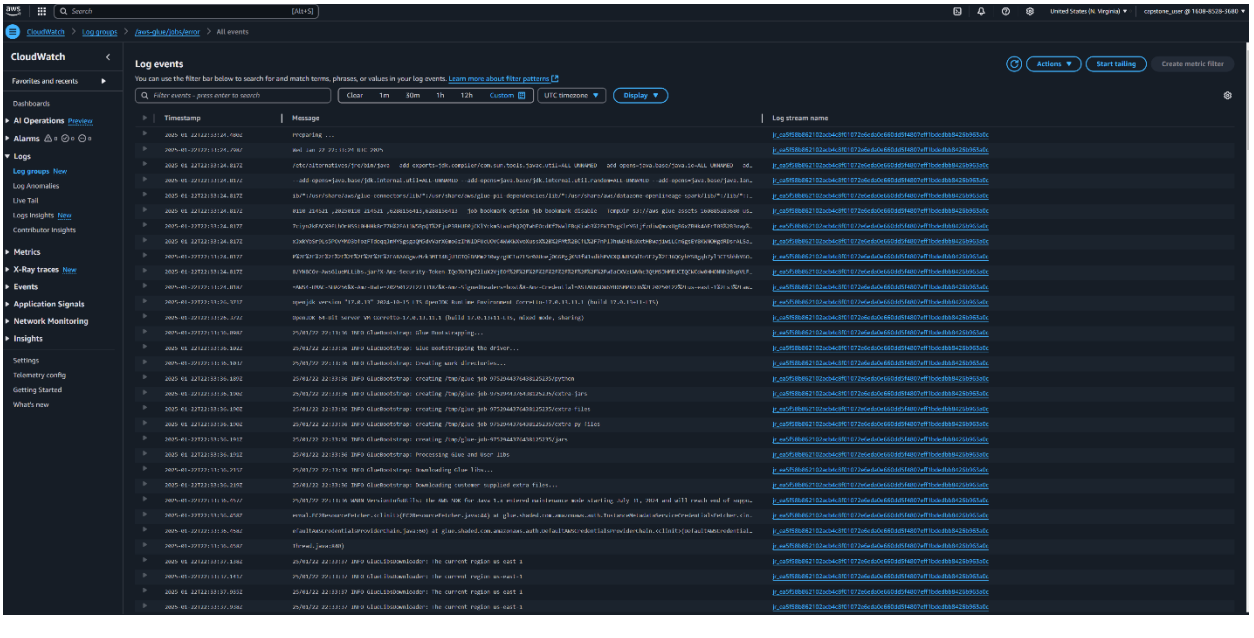
3.3 Scheduling the job run





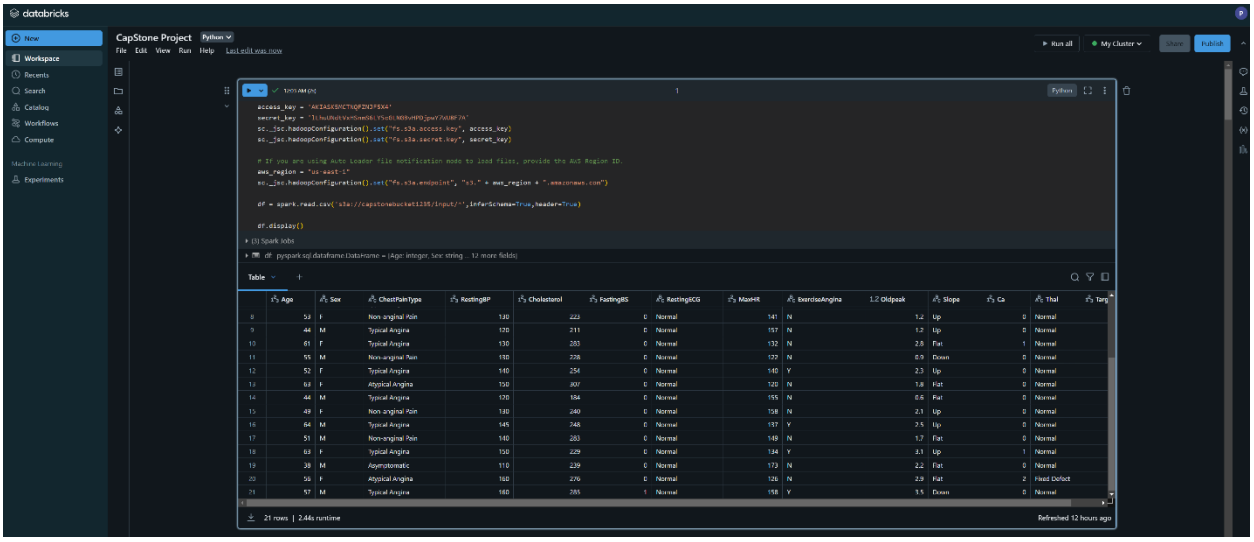
3.4 Displaying the CloudWatch Logs

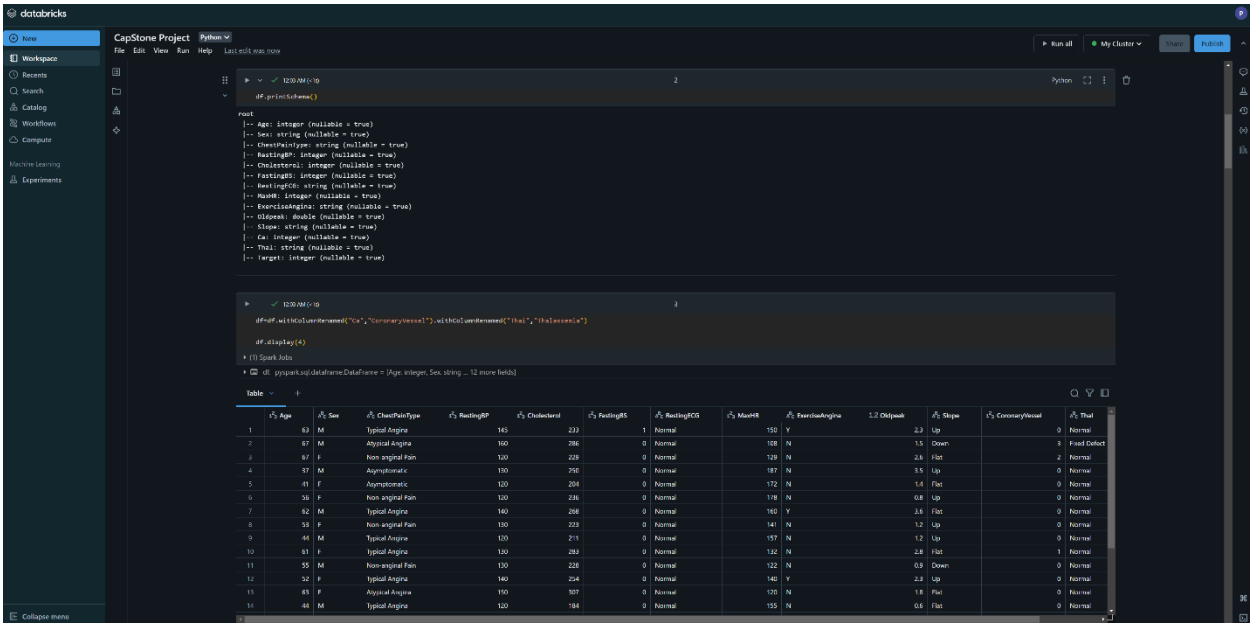


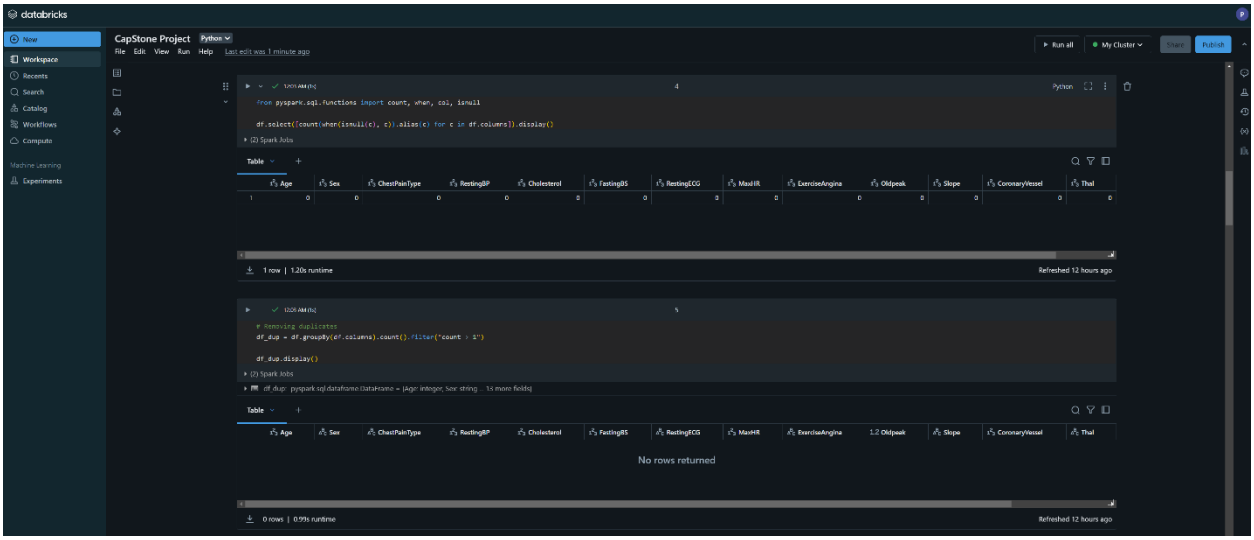


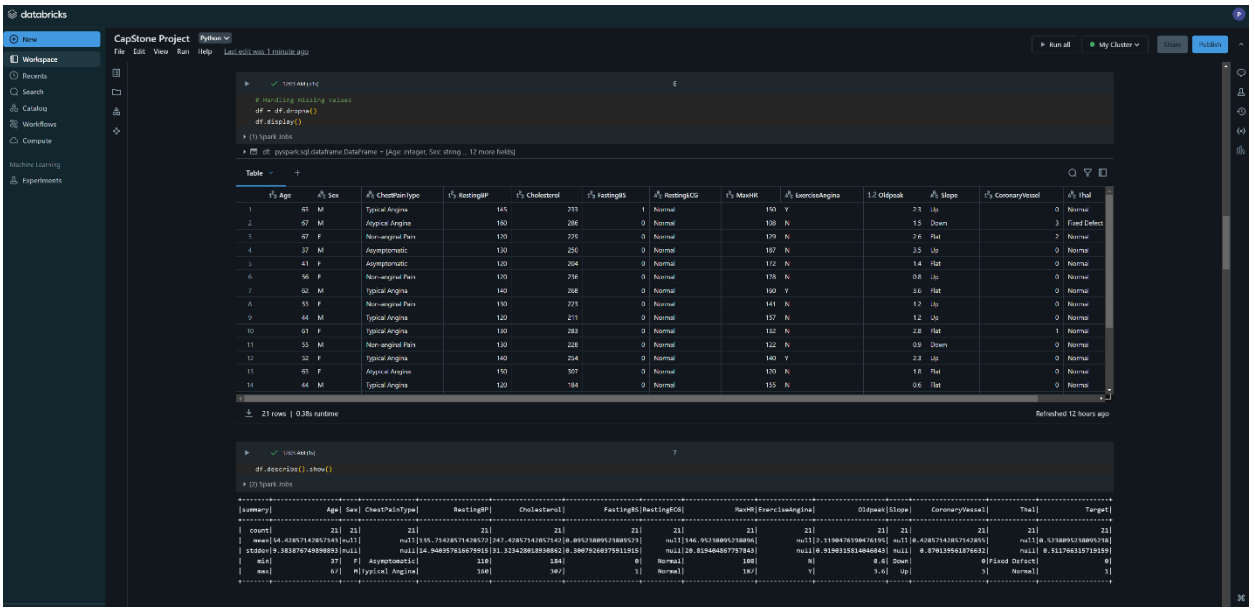
4. Analyzing Data Using Databricks (PySpark and SQL)

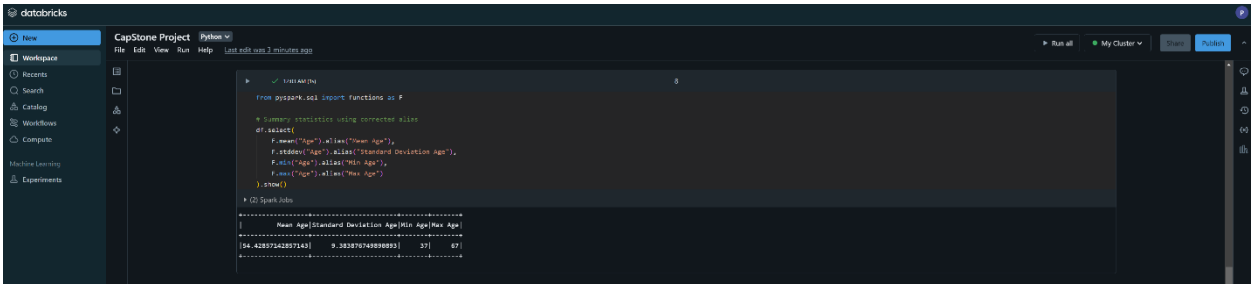
4.1 Strating the ETL process on Databricks

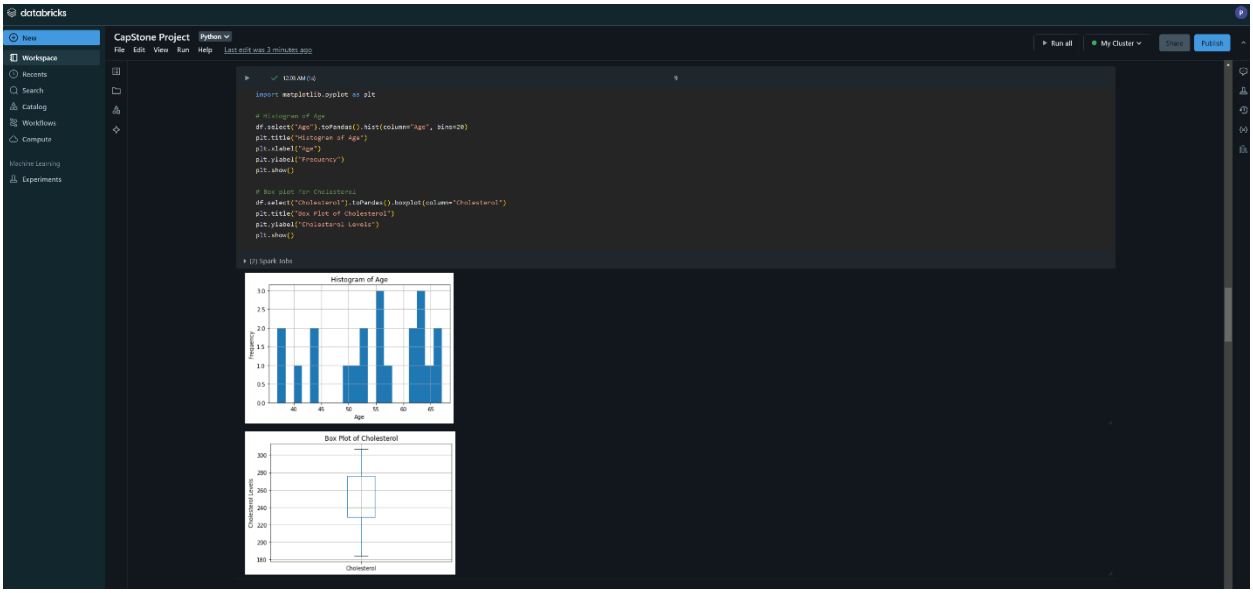


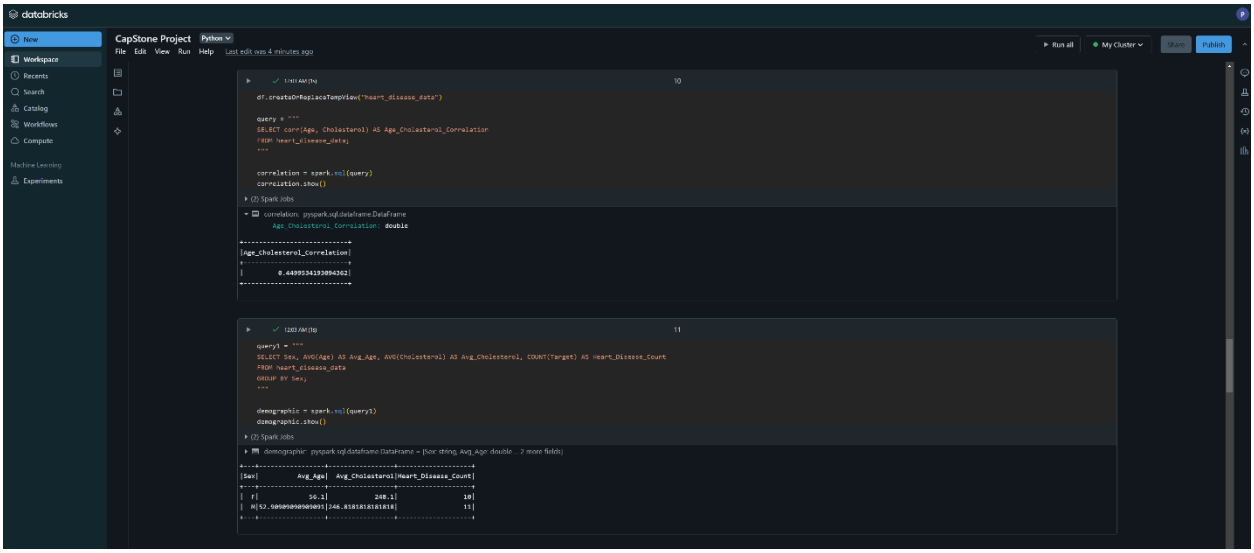


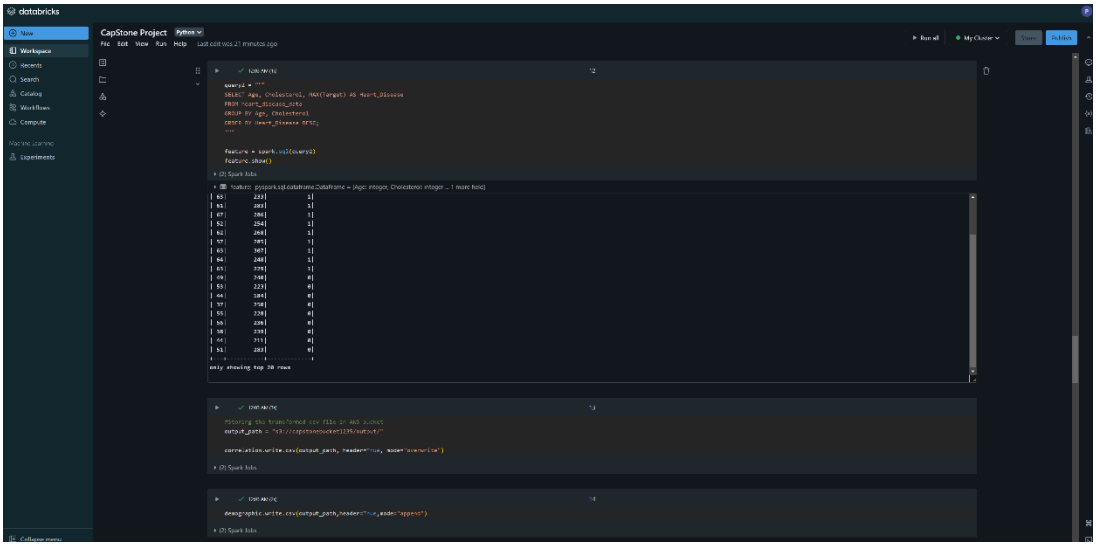

databricks
CapStone Project
Python
File Edit View Run Help Last edit was 1 minute ago
Run all
My Cluster
Share
Publish
New
Workspace
Recents
Search
Catalog
Workflows
Compute
Machine Learning
Experiments
from pyspark.sql.functions import count, when, col, isnull
df.select([count(when(isnull(c), c)).alias(c) for c in df.columns]).display()
(2) Spark Jobs
Table
Age Sex ChestPainType RestingBP Cholesterol FastingBS RestingECG MaxHR ExerciseAngina Oldpeak Slope CoronaryVessel Thal
1 0 0 0 0 0 0 0 0 0 0 0 0 0
1 row | 1.20s runtime
Refreshed 12 hours ago
# Removing duplicates
df_dup = df.groupBy(df.columns).count().filter("count > 1")
df_dup.display()
(2) Spark Jobs
df_dup: pyspark.sql.dataframe.DataFrame = [Age: integer, Sex: string ... 11 more fields]
Table
Age Sex ChestPainType RestingBP Cholesterol FastingBS RestingECG MaxHR ExerciseAngina 1.2 Oldpeak Slope CoronaryVessel Thal
No rows returned
0 rows | 0.99s runtime
Refreshed 12 hours ago

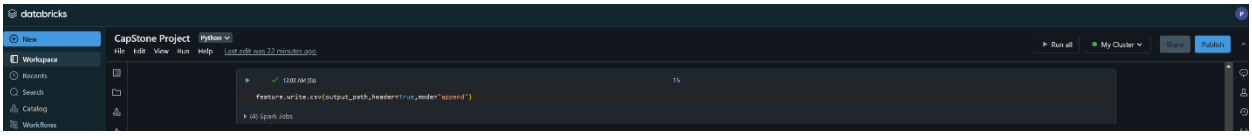

databricks
CapStone Project
Python
File Edit View Run Help Last edit was 1 minute ago
Run all
My Cluster
Share
Publish
# Handling missing values
df = df.dropna()
df.display()
(1) Spark Jobs
df: pyspark.sql.dataframe.DataFrame = [Age: integer, Sex: string ... 12 more fields]
Table
Age Sex ChestPainType RestingBP Cholesterol FastingBS RestingECG MaxHR ExerciseAngina Oldpeak Slope CoronaryVessel Thal
1 65 M Typical Angina 145 233 1 Normal 150 Y 2.3 Up 0 Normal
2 67 M Atypical Angina 160 286 0 Normal 108 N 1.5 Down 3 Fixed Defect
3 67 F Non-anginal Pain 120 229 0 Normal 129 N 2.6 Flat 2 Normal
4 37 M Asymptomatic 130 250 0 Normal 187 N 3.5 Up 0 Normal
5 41 F Asymptomatic 130 204 0 Normal 172 N 1.4 Flat 0 Normal
6 56 F Non-anginal Pain 120 236 0 Normal 178 N 0.8 Up 0 Normal
7 62 M Typical Angina 140 268 0 Normal 160 Y 3.6 Flat 0 Normal
8 53 F Non-anginal Pain 130 223 0 Normal 141 N 1.2 Up 0 Normal
9 44 M Typical Angina 120 215 0 Normal 157 N 1.2 Up 0 Normal
10 61 F Typical Angina 130 283 0 Normal 132 N 2.8 Flat 1 Normal
11 55 M Non-anginal Pain 130 228 0 Normal 122 N 0.9 Down 0 Normal
12 52 F Typical Angina 140 254 0 Normal 140 Y 2.3 Up 0 Normal
13 63 F Atypical Angina 150 407 0 Normal 130 N 1.8 Flat 0 Normal
14 44 M Typical Angina 120 184 0 Normal 155 N 0.6 Flat 0 Normal
21 rows | 0.38s runtime
Refreshed 12 hours ago
df.describe().show()
(2) Spark Jobs
|summary| Age| Sex| ChestPainType| RestingBP| Cholesterol| FastingBS|RestingECG| MaxHR|ExerciseAngina| Oldpeak|Slope| CoronaryVessel| Thal| Target|
count	21	21	21	21	21	21	21	21	21	21	21	21	21	21
mean	54.428571428571431	null	null	135.71428571428572	247.42857142857142	0.09523809523809523	null	146.95238095238096	null	0.4285714285714285	null	0.5238095238095238
stddev	9.383876749898893	null	null	14.940357616679915	31.323428818938862	0.30079260375911915	null	20.815404867757843	null	0.9190331814046043	null	0.8701395619876632	null	0.5117663157159159
min	37	F	Asymptomatic	110	184	0	Normal	108	N	0.6	Down	0	Fixed Defect	0
max	67	M	Typical Angina	160	307	1	Normal	167	Y	3.6	Up	3	Normal	1

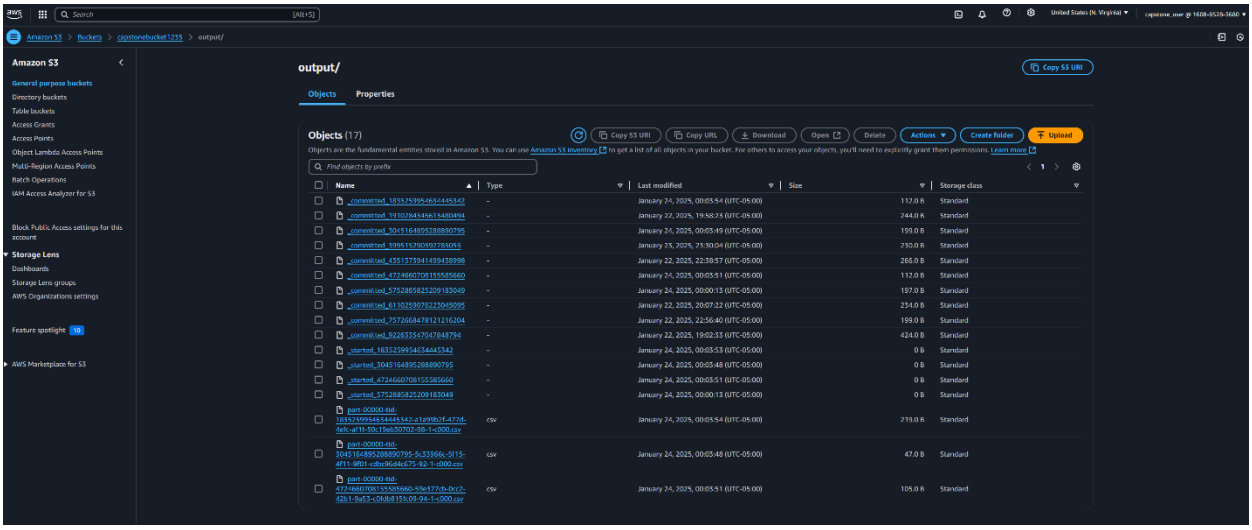

databricks
CapStone Project
Python
File Edit View Run Help Last edit was 3 minutes ago
Run all
My Cluster
Share
Publish
from pyspark.sql import functions as F
# Summary statistics using corrected alias
df.select(
F.mean("Age").alias("Mean Age"),
F.stddev("Age").alias("Standard Deviation Age"),
F.min("Age").alias("Min Age"),
F.max("Age").alias("Max Age")
).show()
(2) Spark Jobs
| Mean Age|Standard Deviation Age|Min Age|Max Age|
|54.428571428571431| 9.383876749898893| 37| 67|

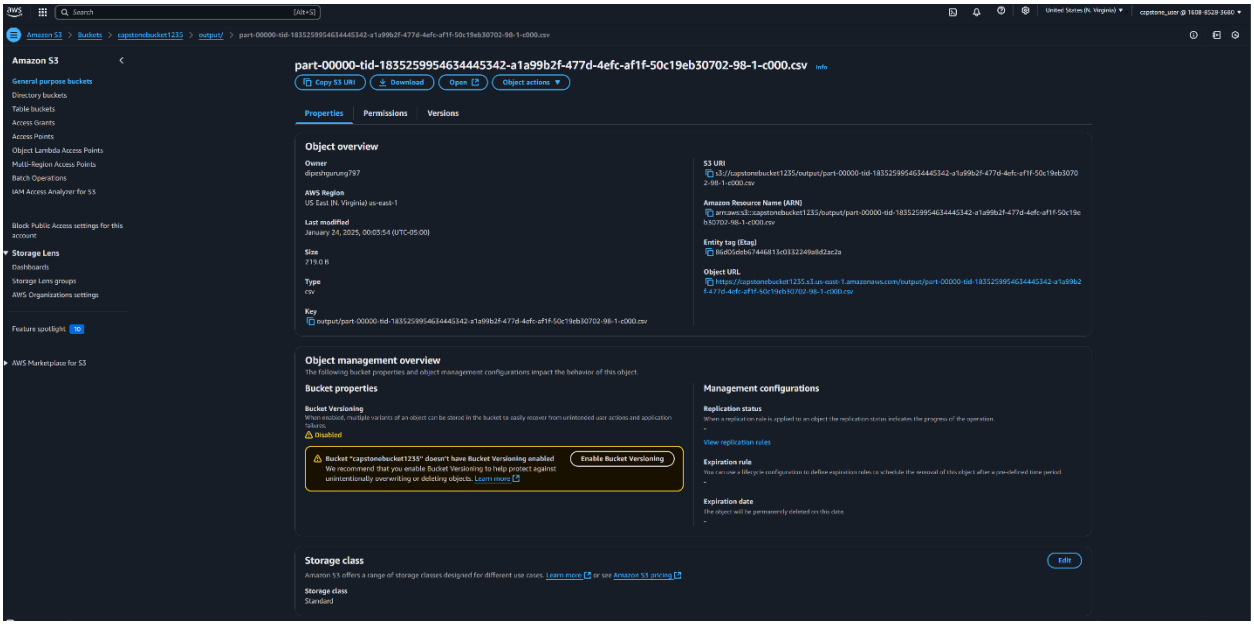

databricks
CapStone Project
Python
File Edit View Run Help Last edit was 3 minutes ago
Run all
My Cluster
Share
Publish
New
Workspace
Recents
Search
Catalog
Workflows
Compute
Machine Learning
Experiments
import matplotlib.pyplot as plt
# Histogram of Age
df.select("Age").toPandas().hist(column="Age", bins=20)
plt.title("Histogram of Age")
plt.xlabel("Age")
plt.ylabel("Frequency")
plt.show()
# Box plot for Cholesterol
df.select("Cholesterol").toPandas().boxplot(column="Cholesterol")
plt.title("Box Plot of Cholesterol")
plt.ylabel("Cholesterol Levels")
plt.show()
(7) Spark Jobs
Histogram of Age
Frequency
Age
Box Plot of Cholesterol
Cholesterol Levels
Cholesterol

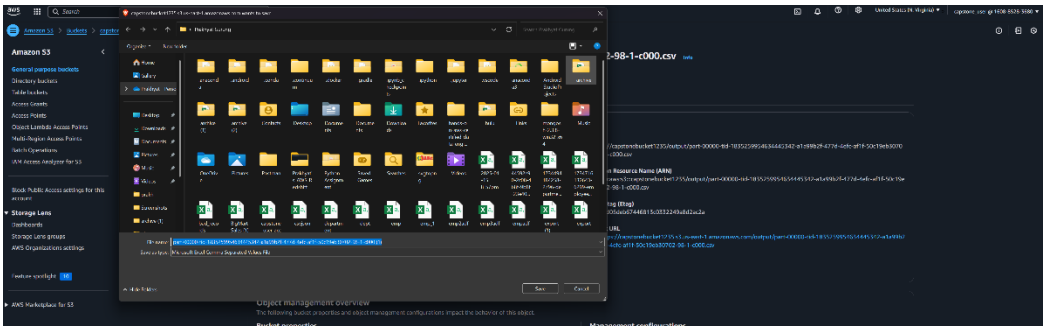

databricks
CapStone Project
Python
File Edit View Run Help Last edit was 4 minutes ago
Run all
My Cluster
Share
Publish
New
Workspace
Recents
Search
Catalog
Workflows
Compute
Machine Learning
Experiments
df.createOrReplaceTempView("heart_disease_data")
query = """
SELECT corr(Age, Cholesterol) AS Age_Cholesterol_Correlation
FROM heart_disease_data;
"""
correlation = spark.sql(query)
correlation.show()
(2) Spark Jobs
correlation: pyspark.sql.dataframe.DataFrame
Age_Cholesterol_Correlation: double
+---------------------------+
|Age_Cholesterol_Correlation|
+---------------------------+
| 0.4499534193094362|
+---------------------------+
query1 = """
SELECT Sex, AVG(Age) AS Avg_Age, AVG(Cholesterol) AS Avg_Cholesterol, COUNT(Target) AS Heart_Disease_Count
FROM heart_disease_data
GROUP BY Sex;
"""
demographic = spark.sql(query1)
demographic.show()
(2) Spark Jobs
demographic: pyspark.sql.dataframe.DataFrame = [Sex: string, Avg_Age: double ... 2 more fields]
+---+------------------+------------------+-----------------+
|Sex| Avg_Age| Avg_Cholesterol|Heart_Disease_Count|
+---+------------------+------------------+-----------------+
| F| 56.1| 248.1| 10|
| M|52.909090909090914|246.81818181818182| 11|
+---+------------------+------------------+-----------------+

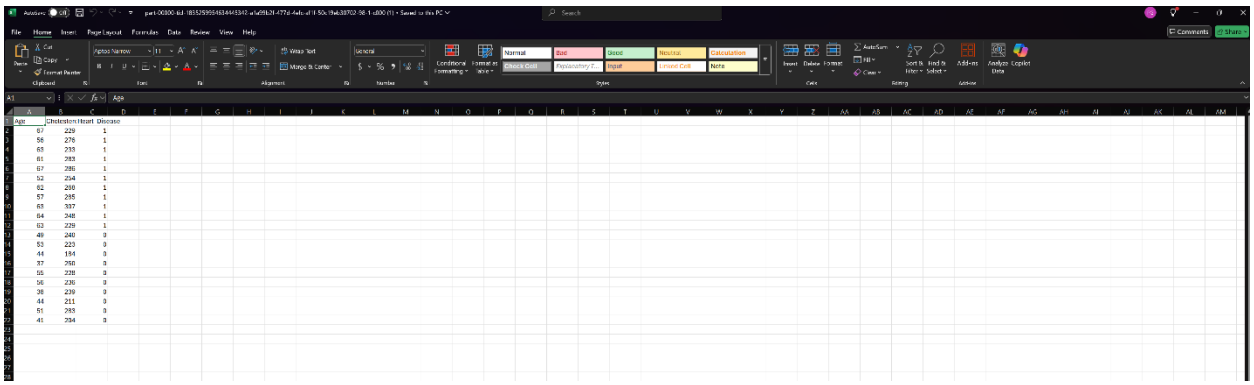

databricks
CapStone Project
Python
File Edit View Run Help
Last edit was 21 minutes ago
Run all
My Cluster
Share
Publish
New
Workspace
Recents
Search
Catalog
Workflows
Compute
Machine Learning
Experiments
query2 = """
SELECT Age, Cholesterol, MAX(Target) AS Heart_Disease
FROM Heart_Disease_Info
GROUP BY Age, Cholesterol
ORDER BY Heart_Disease DESC;
"""
feature = spark.sql(query2)
feature.show()
(2) Spark Jobs
feature: pyspark.sql.dataframe.DataFrame = [Age: integer, Cholesterol: integer ... 1 more field]
65	233	1
61	282	1
67	286	1
52	254	1
62	268	1
57	785	1
63	307	1
64	248	1
65	779	1
69	748	0
53	223	0
44	184	0
57	354	0
55	228	0
56	229	0
58	339	0
44	311	0
51	283	0
only showing top 20 rows
Storing the transformed csv file in AWS bucket
output_path = "s3://capstonebucket1235/output/"
correlation.write.csv(output_path, header="true", mode="overwrite")
(2) Spark Jobs
demographic.write.csv(output_path,header=True,mode="append")
(2) Spark Jobs

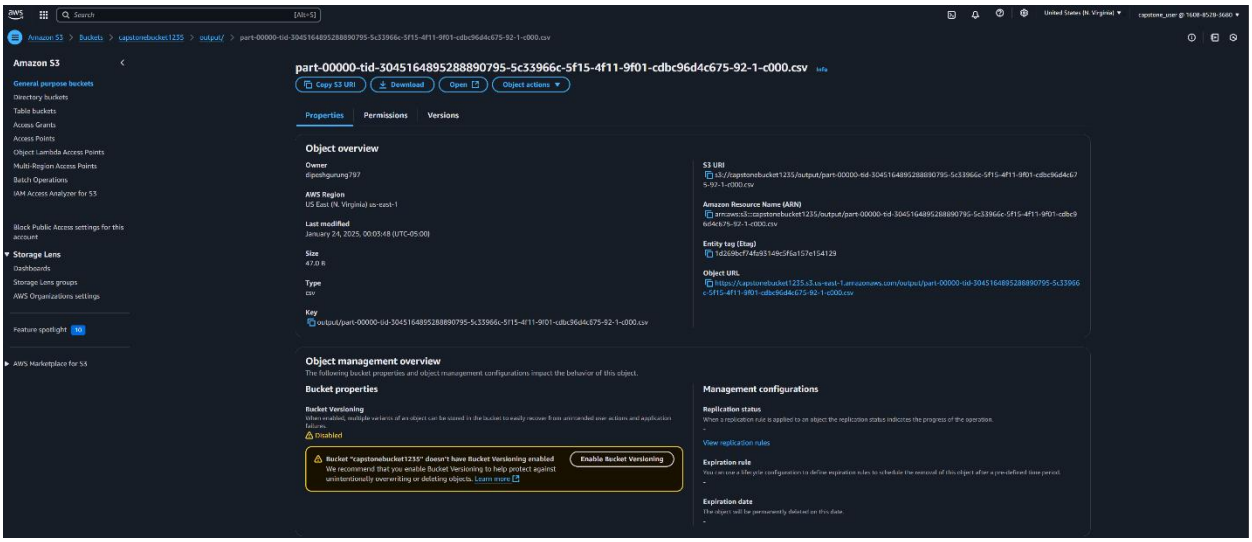

databricks
CapStone Project
Python
File Edit View Run Help
Last edit was 22 minutes ago
Run all
My Cluster
Share
Publish
New
Workspace
Recents
Search
Catalog
Workflows
12:00 AM (9s)
feature.write.csv(output_path,header=True,mode="append")
(4) Spark Jobs

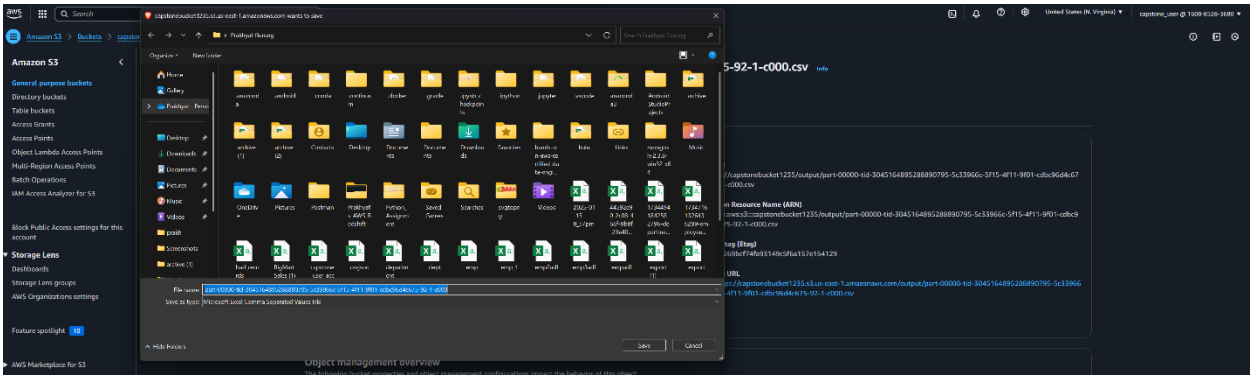

aws
Search
[Alt+S]
United States (N. Virginia)
Amazon S3 > Buckets > capstonebucket1235 > output/
Amazon S3
General purpose buckets
Directory buckets
Table buckets
Access Grants
Access Points
Object Lambda Access Points
Multi-Region Access Points
Batch Operations
IAM Access Analyzer for S3
Block Public Access settings for this account
Storage Lens
Dashboards
Storage Lens groups
AWS Organizations settings
Feature spotlight 10
AWS Marketplace for S3
output/
Copy S3 URI
Objects
Properties
Objects (17)
Copy S3 URI
Copy URL
Download
Open
Delete
Actions
Create folder
Upload
Find objects by prefix
Name
Type
Last modified
Size
Storage class
_committed_1835259954634443432 - January 24, 2025, 00:03:54 (UTC-05:00) 112.0 B Standard
_committed_1910284344613480494 - January 22, 2025, 19:58:21 (UTC-05:00) 244.0 B Standard
_committed_3045164895288890795 - January 24, 2025, 00:03:49 (UTC-05:00) 199.0 B Standard
_committed_1995152905927850S5 - January 23, 2025, 21:30:04 (UTC-05:00) 210.0 B Standard
_committed_4551373941499438998 - January 22, 2025, 22:38:57 (UTC-05:00) 266.0 B Standard
_committed_4724660708155585660 - January 24, 2025, 00:03:51 (UTC-05:00) 112.0 B Standard
_committed_5752385825209183049 - January 24, 2025, 00:00:13 (UTC-05:00) 197.0 B Standard
_committed_6110259076223045095 - January 22, 2025, 20:07:22 (UTC-05:00) 254.0 B Standard
_committed_7572068478121216204 - January 22, 2025, 22:56:40 (UTC-05:00) 199.0 B Standard
_committed_9226335470474848794 - January 22, 2025, 19:02:33 (UTC-05:00) 424.0 B Standard
_started_1835259954634443432 - January 24, 2025, 00:03:53 (UTC-05:00) 0 B Standard
_started_3045164895288890795 - January 24, 2025, 00:03:48 (UTC-05:00) 0 B Standard
_started_4724660708155585660 - January 24, 2025, 00:03:51 (UTC-05:00) 0 B Standard
_started_5752385825209183049 - January 24, 2025, 00:00:13 (UTC-05:00) 0 B Standard
part-00000-tid-1835259954634443432-a1a99b2f-477d-4efc-af1f-50c10eb30702-98-1-c000.csv csv January 24, 2025, 00:03:54 (UTC-05:00) 219.0 B Standard
part-00000-tid-3045164895288890795-5c33964c-5115-4f11-9f01-cdbc9fd46c75-92-1-c000.csv csv January 24, 2025, 00:03:48 (UTC-05:00) 47.0 B Standard
part-00000-tid-4724660708155585660-59e577cb-0cc7-42b1-9a53-c0fdb8115fc0-94-1-c000.csv csv January 24, 2025, 00:03:51 (UTC-05:00) 105.0 B Standard

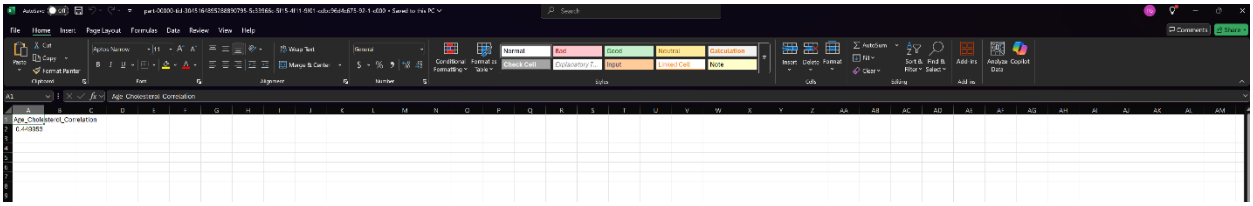

Amazon S3
part-00000-tid-1835259954634445342-a1a99b2f-477d-4efc-af1f-50c19eb30702-98-1-c000.csv
Copy S3 URI
Download
Open
Object actions
Properties
Permissions
Versions
Object overview
Owner
dipeshgurung797
AWS Region
US East (N. Virginia) us-east-1
Last modified
January 24, 2025, 00:03:54 (UTC-05:00)
Size
219.0 B
Type
csv
Key
output/part-00000-tid-1835259954634445342-a1a99b2f-477d-4efc-af1f-50c19eb30702-98-1-c000.csv
S3 URI
s3://capstonebucket1235/output/part-00000-tid-1835259954634445342-a1a99b2f-477d-4efc-af1f-50c19eb30702-98-1-c000.csv
Amazon Resource Name (ARN)
Entity tag (Etag)
Object URL
Object management overview
Bucket properties
Bucket Versioning
Disabled
Bucket "capstonebucket1235" doesn't have Bucket Versioning enabled
We recommend that you enable Bucket Versioning to help protect against unintentionally overwriting or deleting objects.
Enable Bucket Versioning
Management configurations
Replication status
View replication rules
Expiration rule
Expiration date
Storage class
Amazon S3 offers a range of storage classes designed for different use cases.
Storage class
Standard
Edit

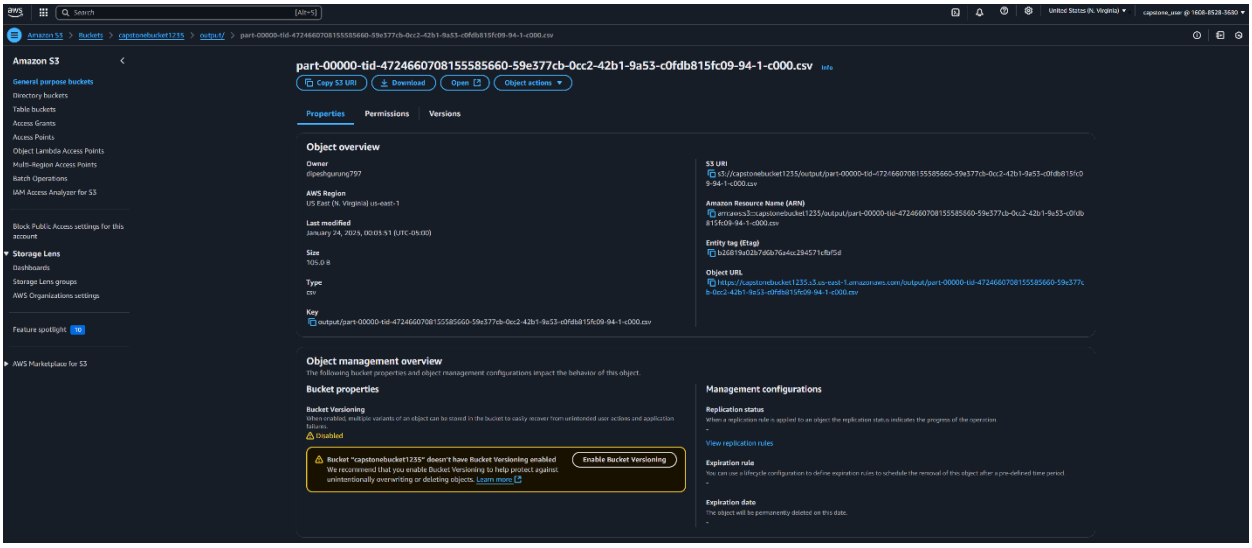


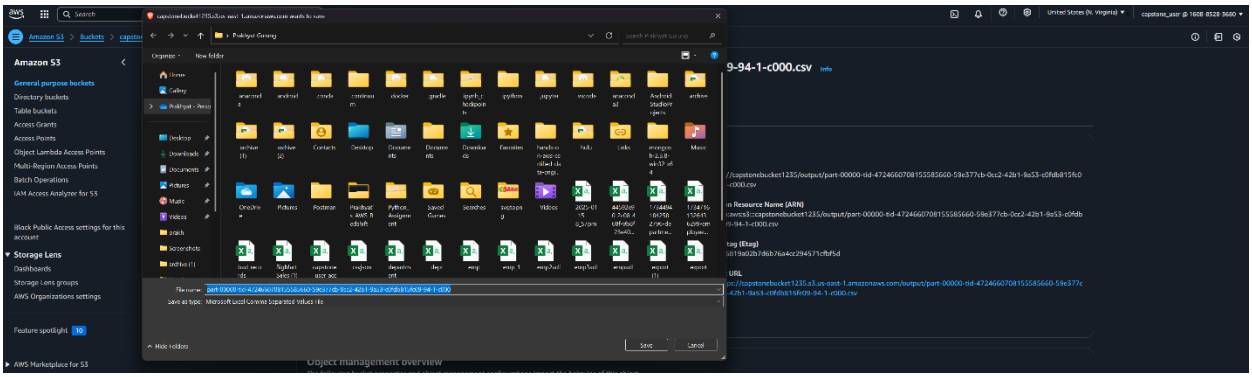

Age
Cholesterol
Heart Disease

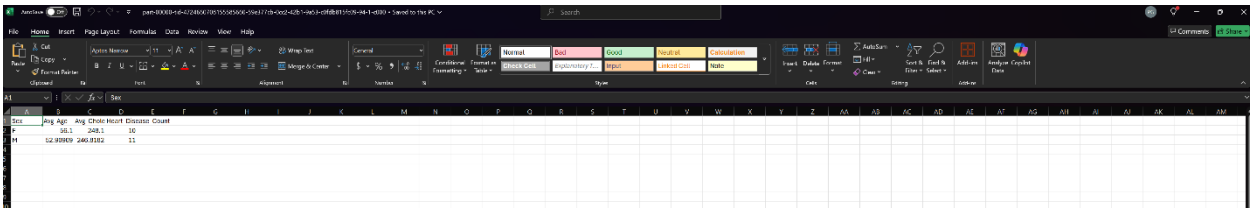

Amazon S3
part-00000-tid-3045164895288890795-5c33966c-5f15-4f11-9f01-cdbc96d4c675-92-1-c000.csv
Copy S3 URI
Download
Open
Object actions
Properties
Permissions
Versions
Object overview
Owner
dipeshgurung797
AWS Region
US East (N. Virginia) us-east-1
Last modified
January 24, 2025, 00:03:48 (UTC-05:00)
Size
47.0 B
Type
csv
Key
output/part-00000-tid-3045164895288890795-5c33966c-5f15-4f11-9f01-cdbc96d4c675-92-1-c000.csv
S3 URI
Amazon Resource Name (ARN)
Entity tag (Etag)
Object URL
Object management overview
Bucket properties
Bucket Versioning
Disabled
Bucket "capstonebucket1235" doesn't have Bucket Versioning enabled
Enable Bucket Versioning
Management configurations
Replication status
View replication rules
Expiration rule
Expiration date

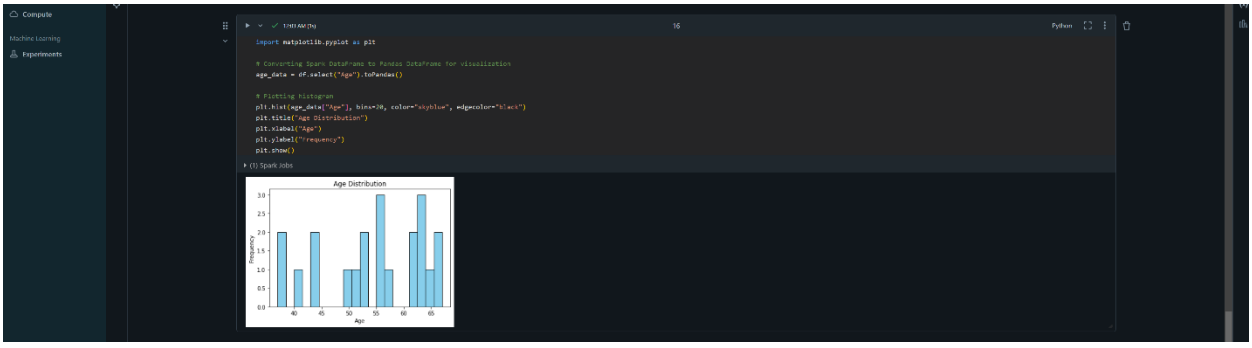

Amazon S3
Buckets
Object management overview

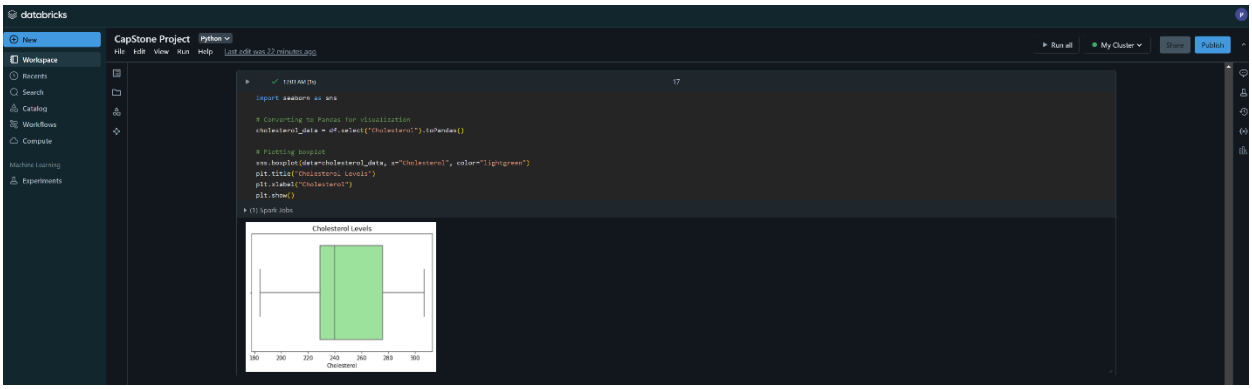

Age_Cholesterol_Correlation
0.4439853

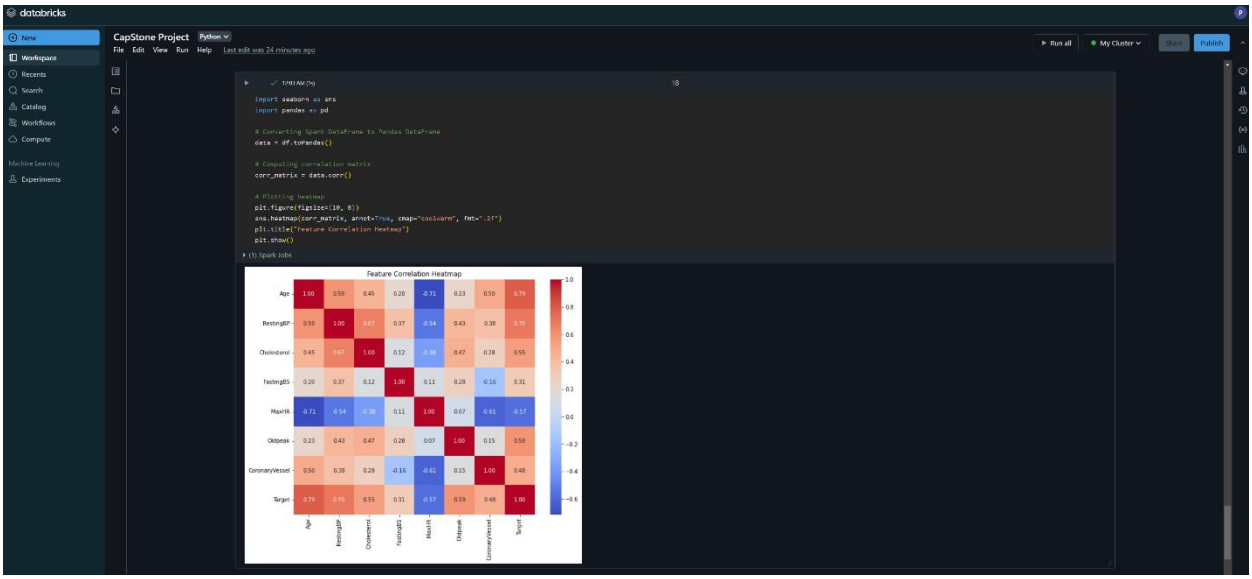

part-00000-tid-4724660708155585660-59e377cb-0cc2-42b1-9a53-c0fdb815fc09-94-1-c000.csv
Copy S3 URI
Download
Open
Object actions
Properties
Permissions
Versions
Object overview
Owner
dipeshgurung797
AWS Region
US East (N. Virginia) us-east-1
Last modified
January 24, 2025, 00:03:51 (UTC-05:00)
Size
105.0 B
Type
csv
Key
output/part-00000-tid-4724660708155585660-59e377cb-0cc2-42b1-9a53-c0fdb815fc09-94-1-c000.csv
S3 URI
s3://capstonebucket1235/output/part-00000-tid-4724660708155585660-59e377cb-0cc2-42b1-9a53-c0fdb815fc09-94-1-c000.csv
Amazon Resource Name (ARN)
arn:aws:s3:::capstonebucket1235/output/part-00000-tid-4724660708155585660-59e377cb-0cc2-42b1-9a53-c0fdb815fc09-94-1-c000.csv
Entity tag (Etag)
b26819a02b7d6b76a4cc294571cfbf5d
Object URL
https://capstonebucket1235.s3.us-east-1.amazonaws.com/output/part-00000-tid-4724660708155585660-59e377cb-0cc2-42b1-9a53-c0fdb815fc09-94-1-c000.csv
Object management overview
Bucket properties
Bucket Versioning
Disabled
Bucket "capstonebucket1235" doesn't have Bucket Versioning enabled
Enable Bucket Versioning
Management configurations
Replication status
View replication rules
Expiration rule
Expiration date

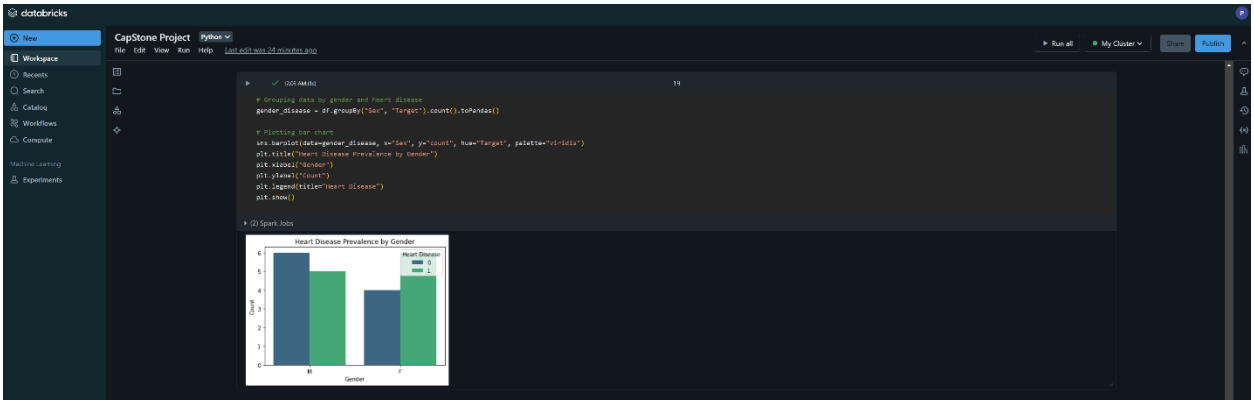





import matplotlib.pyplot as plt
# Converting Spark DataFrame to Pandas DataFrame for visualization
age_data = df.select("Age").toPandas()
# Plotting Histogram
plt.hist(age_data["Age"], bins=20, color="skyblue", edgecolor="black")
plt.title("Age Distribution")
plt.xlabel("Age")
plt.ylabel("Frequency")
plt.show()
Age Distribution
Frequency
Age


databricks
CapStone Project
Python
File Edit View Run Help
Last edit was 22 minutes ago
Run all
My Cluster
Share
Publish
New
Workspace
Recents
Search
Catalog
Workflows
Compute
Machine Learning
Experiments
17
import seaborn as sns
# Converting to Pandas for visualization
cholesterol_data = df.select("Cholesterol").toPandas()
# Plotting boxplot
sns.boxplot(data=cholesterol_data, x="Cholesterol", color="lightgreen")
plt.title('Cholesterol Levels')
plt.xlabel("Cholesterol")
plt.show()
(1) Spark Jobs
Cholesterol Levels
180
200
220
240
260
280
300
Cholesterol


databricks
CapStone Project
Python
File Edit View Run Help
Last edit was 24 minutes ago
Run all
My Cluster
Share
Publish
New
Workspace
Recents
Search
Catalog
Workflows
Compute
Machine Learning
Experiments
18
import seaborn as sns
import pandas as pd
# Converting Spark DataFrame to Pandas DataFrame
data = df.toPandas()
# Computing correlation matrix
corr_matrix = data.corr()
# Plotting heatmap
plt.figure(figsize=(10, 8))
sns.heatmap(corr_matrix, annot=True, cmap="coolwarm", fmt=".2f")
plt.title("Feature Correlation Heatmap")
plt.show()
(1) Spark Jobs
Feature Correlation Heatmap
Age
RestingBP
Cholesterol
FastingBS
MaxHR
Oldpeak
CoronaryVessel
Target


databricks
CapStone Project
Python
File Edit View Run Help
Last edit was 24 minutes ago
Run all
My Cluster
Share
Publish
New
Workspace
Recents
Search
Catalog
Workflows
Compute
Machine Learning
Experiments
19
# Grouping data by gender and heart disease
gender_disease = df.groupBy("Sex", "Target").count().toPandas()
# Plotting bar chart
sns.barplot(data=gender_disease, x="Sex", y="count", hue="Target", palette="viridis")
plt.title("Heart Disease Prevalence by Gender")
plt.xlabel("Gender")
plt.ylabel("Count")
plt.legend(title="Heart Disease")
plt.show()
(2) Spark Jobs
Heart Disease Prevalence by Gender
Heart Disease
0
1
Count
Gender