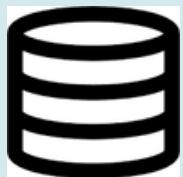
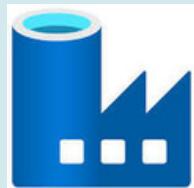


OLYMPIC DATA ANALYTICS



DATA SOURCE



DATA FACTORY



DATA LAKE GEN 2
RAW DATA



DATABRICKS



DATA LAKE GEN 2
TRANSFORMED
DATA



AZURE SYNAPSE
ANALYTICS

RESOURCE GROUPS

OLYMPIC

The screenshot shows the Microsoft Azure Resource Groups blade for the 'Olympic' resource group. The left sidebar contains navigation links for Overview, Activity log, Access control (IAM), Tags, Resource visualizer, Events, and Settings (Deployments, Security, Deployment stacks, Policies, Properties, Locks). The main area is titled 'Essentials' and shows a table of resources. The table has columns for Name, Type, and Location. The resources listed are:

Name	Type	Location
dataolympic	Azure Databricks Service	Southeast Asia
olympictanishq	Storage account	Southeast Asia
tokyo-olympic-tanishq	Synapse workspace	Southeast Asia
tokyo-olympictanishq	Data factory (V2)	Southeast Asia

STORAGE ACCOUNTS

The screenshot shows the Microsoft Azure Storage Accounts blade for the 'olympictanishq' container. The top navigation bar includes links for Home, Storage accounts, and olympictanishq | Containers. The container name 'olympictanishq' is displayed with a 'Container' label. The toolbar includes actions for Upload, Add Directory, Refresh, Rename, Delete, Change tier, Acquire lease, Break lease, and Give feedback. A note indicates the Authentication method is Access key (Switch to Microsoft Entra user account) and the Location is olympictanishq. A search bar allows searching by blob prefix. The main table lists blobs with columns for Name, Modified, Access tier, Archive status, Blob type, Size, and Lease state. Two blobs are listed: 'raw-data' and 'transformed-data'.

Name	Modified	Access tier	Archive status	Blob type	Size	Lease state
raw-data					-	...
transformed-data					-	...

RAW DATA

Home > Storage accounts > olympictanishq | Containers >

olympictanishq ...

Container

» [Upload](#) [Add Directory](#) [Refresh](#) | [Rename](#) [Delete](#) [Change tier](#) [Acquire lease](#) [Break lease](#) [Give feedback](#)

Authentication method: Access key ([Switch to Microsoft Entra user account](#))
Location: [olympictanishq](#) / raw-data

Search blobs by prefix (case-sensitive) Show deleted objects

Name	Modified	Access tier	Archive status	Blob type	Size	Lease state
[..]						...
Athletes.csv	28/04/2024, 19:21:33	Hot (Inferred)		Block blob	408.68 KiB	Available
Coaches.csv	28/04/2024, 19:21:47	Hot (Inferred)		Block blob	16.49 KiB	Available
EntriesGender.csv	28/04/2024, 19:22:06	Hot (Inferred)		Block blob	1.1 KiB	Available
Medals.csv	28/04/2024, 19:22:20	Hot (Inferred)		Block blob	2.36 KiB	Available
Teams.csv	28/04/2024, 19:22:35	Hot (Inferred)		Block blob	34.44 KiB	Available

TRANSFORMED-DATA

Microsoft Azure [Upgrade](#) Search resources, services, and docs (G+/-)

Home > Storage accounts > olympictanishq | Containers >

olympictanishq ...

Container

» [Upload](#) [Add Directory](#) [Refresh](#) | [Rename](#) [Delete](#) [Change tier](#) [Acquire lease](#) [Break lease](#) [Give feedback](#)

Authentication method: Access key ([Switch to Microsoft Entra user account](#))
Location: [olympictanishq](#) / transformed-data / athletes

Search blobs by prefix (case-sensitive) Show deleted objects

Name	Modified	Access tier	Archive status	Blob type	Size	Lease state
[..]						...
_committed_2277468800143869808	29/04/2024, 09:32:16	Hot (Inferred)		Block blob	112 B	Available
_started_2277468800143869808	29/04/2024, 09:32:15	Hot (Inferred)		Block blob	0 B	Available
_SUCCESS	29/04/2024, 09:32:16	Hot (Inferred)		Block blob	0 B	Available
part-00000-tid-2277468800143869808-8bc99c...	29/04/2024, 09:32:15	Hot (Inferred)		Block blob	397.91 KiB	Available

DATA FACTORIES

DATA PIPELINE



DATABRICKS

The screenshot shows the Microsoft Azure Databricks workspace interface. On the left, the sidebar includes options like New, Workspace, Recents, Catalog, Workflows, Compute, SQL, SQL Editor, Queries, Dashboards, Alerts, Query History, and SQL Warehouses. The main area displays a Python notebook titled 'olympic'. The notebook contains the following code:

```
# Convert the list to a DataFrame
mounts_df = spark.createDataFrame(dbutils.fs.mounts())

# Check if the mount point already exists
if mounts_df.filter(mounts_df.mountPoint == "/mnt/tokyoolympic").count() > 0:
    dbutils.fs.unmount("/mnt/tokyoolympic")

configs = {
    "fs.azure.account.auth.type": "OAuth",
    "fs.azure.account.oauth.provider.type": "org.apache.hadoop.fs.azurebfs.oauth2.ClientCredsTokenProvider",
    "fs.azure.account.oauth2.client.id": "39e1c726-816c-4eee-8917-19361c0bc667",
    "fs.azure.account.oauth2.client.secret": "ub08Q~oBtCKfJUaAsGdczo6FLXucIETxZliZqbZj",
    "fs.azure.account.oauth2.client.endpoint": "https://login.microsoftonline.com/7e3fb8b0-353b-4478-992c-53c7ec3135e9/oauth2/token"
}

# Mount the directory
dbutils.fs.mount(
    source="abfss://olympictanishq@olympictanishq.dfs.core.windows.net", # container@storageacc
    mount_point="/mnt/tokyoolympic",
    extra_configs=configs)
```

The URL at the bottom of the screen is: <https://adb-1200210281944828.8.azuredata.databricks.net/browse/folders/4361584012429293?o=1200210281944828>

```
%fs  
ls "/mnt/tokyoolympic"
```

Table +

New result table: ON

Search



	A ^B _C path	A ^B _C name	1 ² ₃ size	1 ² ₃ modificationTime
1	dbfs:/mnt/tokyoolympic/raw-data/	raw-data/	0	1714308047000
2	dbfs:/mnt/tokyoolympic/transformed-dat...	transformed-dat...	0	1714308116000

▶ v ✓ Yesterday (6s) 4 Python 🗑 ⚙ ⌂ ⏰

```
athletes = spark.read.format("csv").option("header","true").option("inferSchema","true").load("/mnt/tokyoolympic/raw-data/Athletes.csv")
coaches = spark.read.format("csv").option("header","true").option("inferSchema","true").load("/mnt/tokyoolympic/raw-data/Coaches.csv")
entriesgender = spark.read.format("csv").option("header","true").option("inferSchema","true").load("/mnt/tokyoolympic/raw-data/EntriesGender.csv")
medals = spark.read.format("csv").option("header","true").option("inferSchema","true").load("/mnt/tokyoolympic/raw-data/Medals.csv")
teams = spark.read.format("csv").option("header","true").option("inferSchema","true").load("/mnt/tokyoolympic/raw-data/Teams.csv")
```

▶ (10) Spark Jobs

- ▶ [] athletes: pyspark.sql.dataframe.DataFrame = [PersonName: string, Country: string ... 1 more field]
- ▶ [] coaches: pyspark.sql.dataframe.DataFrame = [Name: string, Country: string ... 2 more fields]
- ▶ [] entriesgender: pyspark.sql.dataframe.DataFrame = [Discipline: string, Female: integer ... 2 more fields]
- ▶ [] medals: pyspark.sql.dataframe.DataFrame = [Rank: integer, Team_Country: string ... 5 more fields]
- ▶ [] teams: pyspark.sql.dataframe.DataFrame = [TeamName: string, Discipline: string ... 2 more fields]

▶ v ✓ Yesterday (1s) 5 Python 🗑 ⚙ ⌂ ⏰

```
athletes.show()
```

▶ (1) Spark Jobs

PersonName	Country	Discipline
AALERUD Katrine	Norway	Cycling Road
ABAD Nestor	Spain	Artistic Gymnastics
ABAGNALE Giovanni	Italy	Rowing
ABALDE Alberto	Spain	Basketball
ABALDE Tamara	Spain	Basketball
ABALO Luc	France	Handball
ABAROA Cesar	Chile	Rowing

▶ ✓ Yesterday (<1s) 6

```
athletes.printSchema()
```

root

- PersonName: string (nullable = true)
- Country: string (nullable = true)
- Discipline: string (nullable = true)

▶ ✓ Yesterday (1s) 7 Python

```
top_gold_medal_countries = medals.orderBy("Gold", ascending=False).select("Team_Country","Gold").show()
```

▶ (1) Spark Jobs

Team_Country Gold
United States of ... 39
People's Republic... 38
Japan 27
Great Britain 22
ROC 20
Australia 17
Netherlands 10
France 10
Germany 10
Italy 10

▶ ✓ Yesterday (1s) 8 Python

```
# Calculate the average number of entries by gender for each discipline
average_entries_by_gender = entriesgender.withColumn(
    'Avg_Female', entriesgender['Female'] / entriesgender['Total']
).withColumn(
    'Avg_Male', entriesgender['Male'] / entriesgender['Total']
)
average_entries_by_gender.show()
```

▶ (1) Spark Jobs

Discipline Female Male Total Avg_Female Avg_Male
3x3 Basketball 32 32 64 0.5 0.5
Archery 64 64 128 0.5 0.5
Artistic Gymnastics 98 98 196 0.5 0.5
Artistic Swimming 105 0 105 1.0 0.0
Athletics 969 1072 2041 0.4747672709456149 0.5252327290543851
Badminton 86 87 173 0.49710982658959535 0.5028901734104047

TRANSFORMED-DATA

```
✓ Yesterday (1s) 10 Python ⚡

# Read the athletes CSV file into a Spark DataFrame
athletes_transformed = spark.read.csv("/mnt/tokyoolympic/transformed-data/athletes", header=True, inferSchema=True)

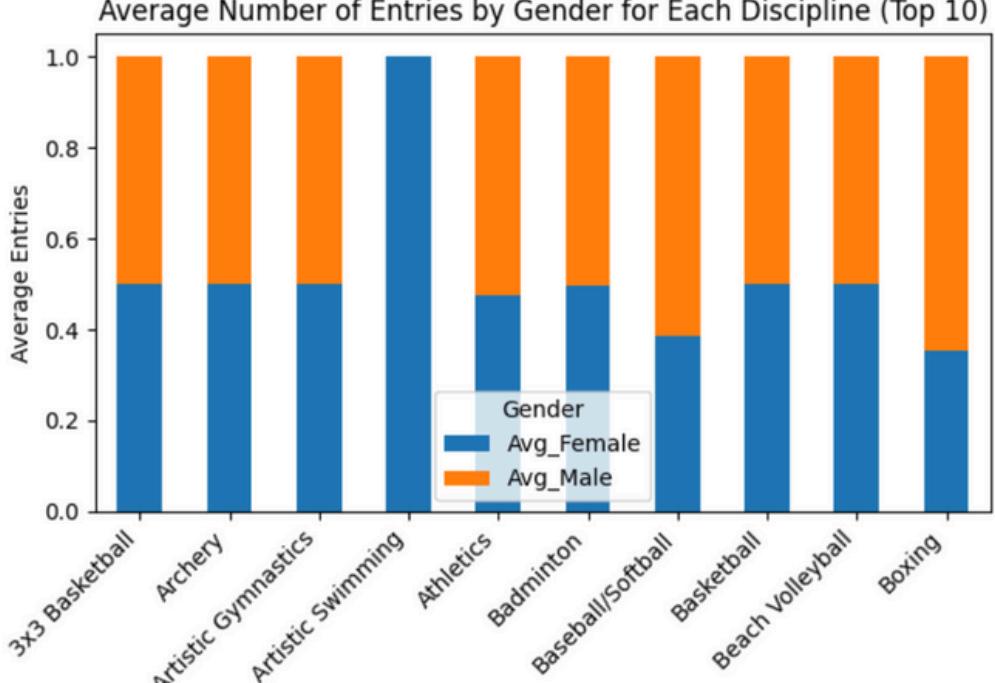
# Show the first few rows of the DataFrame
athletes_transformed.show()
```

PersonName	Country	Discipline
AALERUD Katrine	Norway	Cycling Road
ABAD Nestor	Spain	Artistic Gymnastics
ABAGNALE Giovanni	Italy	Rowing
ABALDE Alberto	Spain	Basketball
ABALDE Tamara	Spain	Basketball
ABALO Luc	France	Handball

```
▶ ✓ Yesterday (1s) 12

# Convert Spark DataFrame to Pandas DataFrame
average_entries_by_gender_pd = average_entries_by_gender.limit(10).toPandas()

# Plot the average number of entries by gender for each discipline
plt.figure(figsize=(12, 8))
average_entries_by_gender_pd.plot(kind='bar', x='Discipline', y=['Avg_Female', 'Avg_Male'], stacked=True)
plt.xlabel('Discipline')
plt.ylabel('Average Entries')
plt.title('Average Number of Entries by Gender for Each Discipline (Top 10)')
plt.xticks(rotation=45, ha='right')
plt.legend(title='Gender')
plt.tight_layout()
plt.show()
```

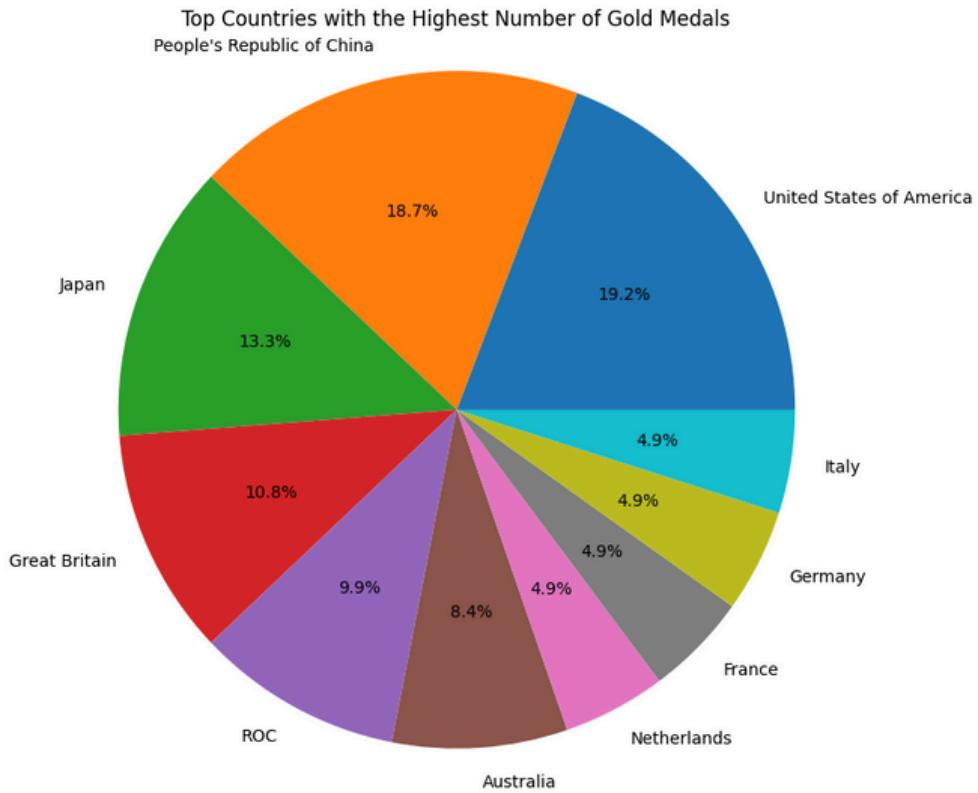
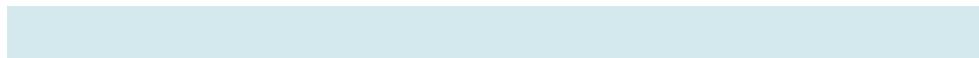


Yesterday (1s) 13 Python

```
# Find the top countries with the highest number of gold medals
top_gold_medal_countries = medals.orderBy("Gold", ascending=False).select("Team_Country", "Gold")

# Convert Spark DataFrame to Pandas DataFrame
top_gold_medal_countries_pd = top_gold_medal_countries.limit(10).toPandas()

# Plot the top countries with the highest number of gold medals using a pie chart
plt.figure(figsize=(8, 8))
plt.pie(top_gold_medal_countries_pd['Gold'], labels=top_gold_medal_countries_pd['Team_Country'], autopct='%.1f%%')
plt.title('Top Countries with the Highest Number of Gold Medals')
plt.axis('equal')
plt.show()
```

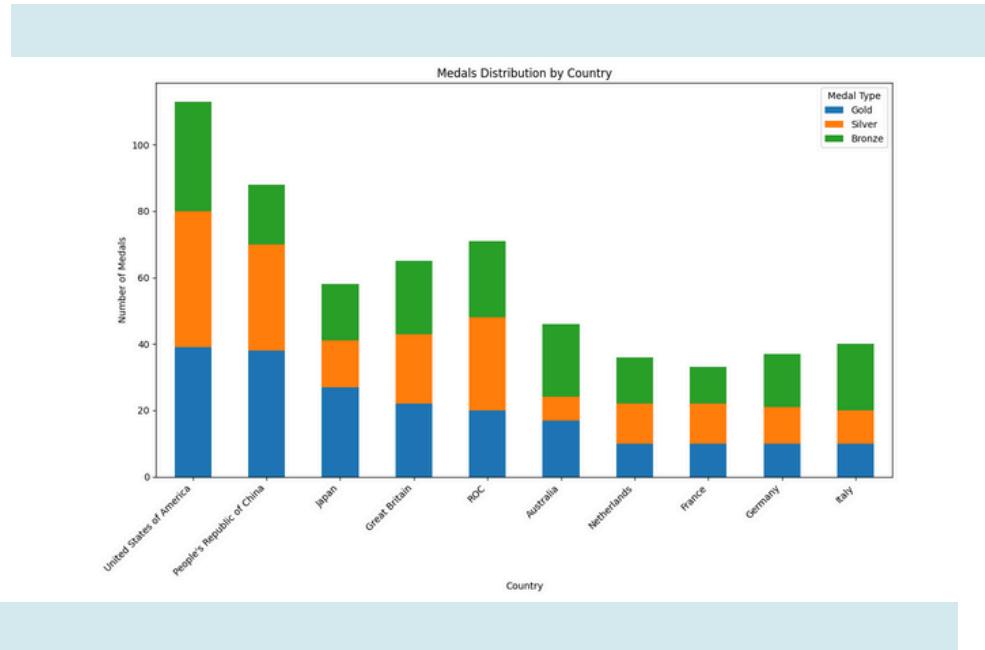


Yesterday (1s) 14 Python

```
# Convert Spark DataFrame to Pandas DataFrame
medals_pd = medals.limit(10).toPandas()

# Plot the distribution of medals by country using a stacked bar chart
medals_pd.set_index('Team_Country')[['Gold', 'Silver', 'Bronze']].plot(kind='bar', stacked=True, figsize=(12, 8))
plt.xlabel('Country')
plt.ylabel('Number of Medals')
plt.title('Medals Distribution by Country')
plt.xticks(rotation=45, ha='right')
plt.legend(title='Medal Type')
plt.tight_layout()
plt.show()
```

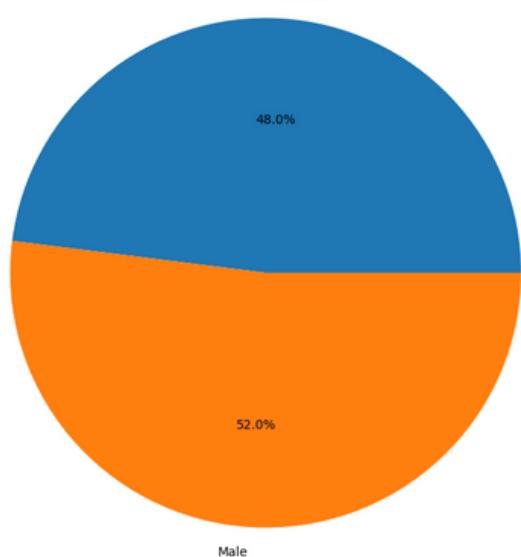
(1) Spark Jobs



```
▶ ▾ ✓ Yesterday (1s) 15 Python ⚡ ⚡ ⚡ ⚡
# Calculate total entries by gender
total_female_entries = average_entries_by_gender.selectExpr("sum(Female) as TotalFemale").collect()[0]['TotalFemale']
total_male_entries = average_entries_by_gender.selectExpr("sum(Male) as TotalMale").collect()[0]['TotalMale']

# Plot the number of entries by gender using a pie chart
plt.figure(figsize=(8, 8))
plt.pie([total_female_entries, total_male_entries], labels=['Female', 'Male'], autopct='%1.1f%')
plt.title('Number of Entries by Gender')
plt.axis('equal')
plt.show()
```

Number of Entries by Gender



SYNAPSE STUDIO

Microsoft Azure | Synapse Analytics > tokyo-olympic-tanish | Search

Synapse live | Validate all | Publish all

Data

Workspace

Lake database

Filter resources by name

tokyoolympic

Tables

- athletes
- coaches
- entriesgender
- medals
- teams

SQL script 1

Other users in your workspace may have access to modify this item. Do not use this item unless you trust all users who may have access to the workspace.

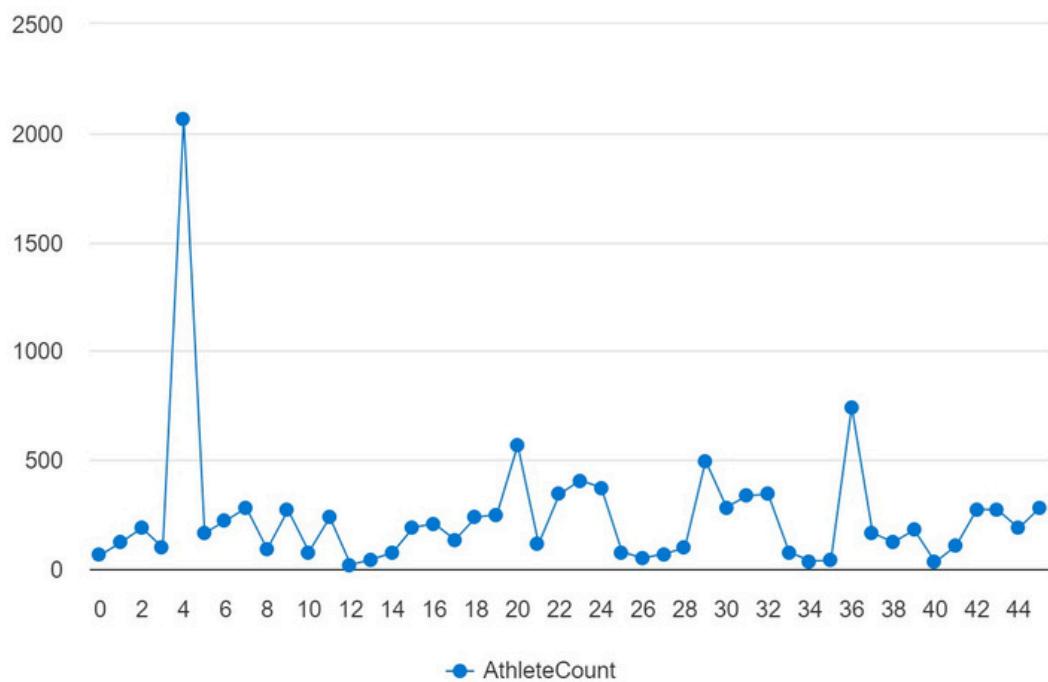
Run Undo Publish Query plan Connect to Built-in ...

```
1 SELECT Discipline, COUNT(*) AS AthleteCount
2 FROM athletes
3 GROUP BY Discipline;
4
5
6 SELECT * FROM athletes
7 WHERE Country = 'Chile';
8
9 SELECT *
10 FROM athletes
11 WHERE Country IN (
12     SELECT Country
13     FROM athletes
14     GROUP BY Country
15     HAVING COUNT(*) > 2
16 );
17
18 SELECT *
19 FROM athletes
20 WHERE Country IN /
```

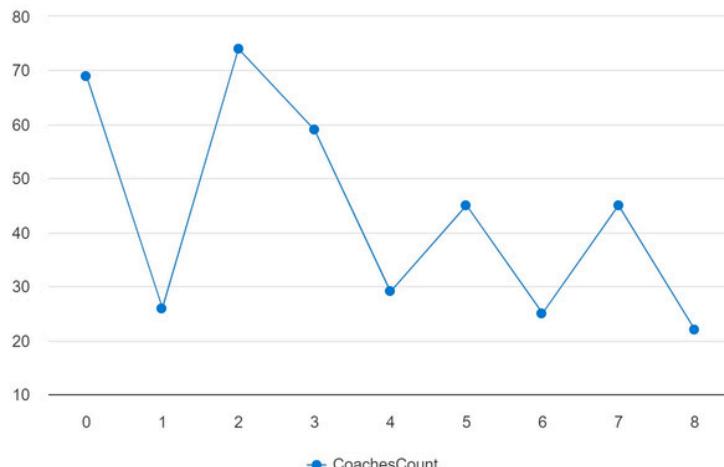
Messages

A 000001 Query compiled with errors

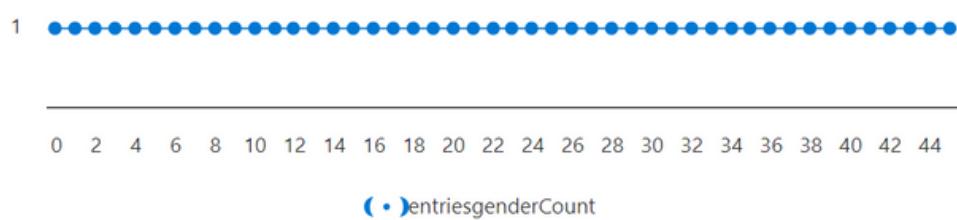
ATHLETES



COACHES



ENTRIESGENDER



TEAMS

