Academy Data Engineering Capstone Project

Overview and objectives

Throughout this course, you have completed hands-on labs, where you used the features of different AWS services to practice ingesting large datasets, transforming them, and extracting information from them.

In this capstone project, you are challenged to build a solution that uses many AWS services that are familiar to you *without* being given step-by-step guidance. Specific sections of the assignment are meant to be challenging.

This capstone will challenge you to do the following:

- Launch and configure an AWS Cloud9 integrated development environment (IDE) instance.
- Transform CSV-formatted data files to the Apache Parquet format and upload them to Amazon S3.
- Create an AWS Glue crawler to infer the structure of the data.
- Use Amazon Athena to query the data.
- Create an Athena view.
- Run queries on Athena view to analyse data.

Duration and monitoring your budget

This capstone is anticipated to require approximately **4 hours** to complete.

This environment is long lived. When the session timer runs to 0:00, the session will end, but any data and resources that you created in the AWS account will be retained. If you later launch a new session (for example, the next day), you will find that your work is still in the lab environment. Also, at any point before the session timer reaches 0:00, you can choose **Start Lab** again to extend the lab session time.

ℹ **Important:** Monitor your lab budget in the lab interface. When you have an active lab session, the latest known remaining budget information displays at the top of this screen. This data comes from AWS Budgets, which typically updates every 8 to 12 hours. Therefore, *the remaining budget that you see might not reflect your most recent account activity*. **If you exceed your lab budget, your lab account will be disabled, and all progress and resources will be lost**. Therefore, it is important for you to manage your spending.

AWS service restrictions

In this lab environment, access to AWS services and service actions might be restricted to the ones that are needed to complete the lab instructions. You might encounter errors if you attempt to access other services or perform actions beyond the ones that are described in this lab.

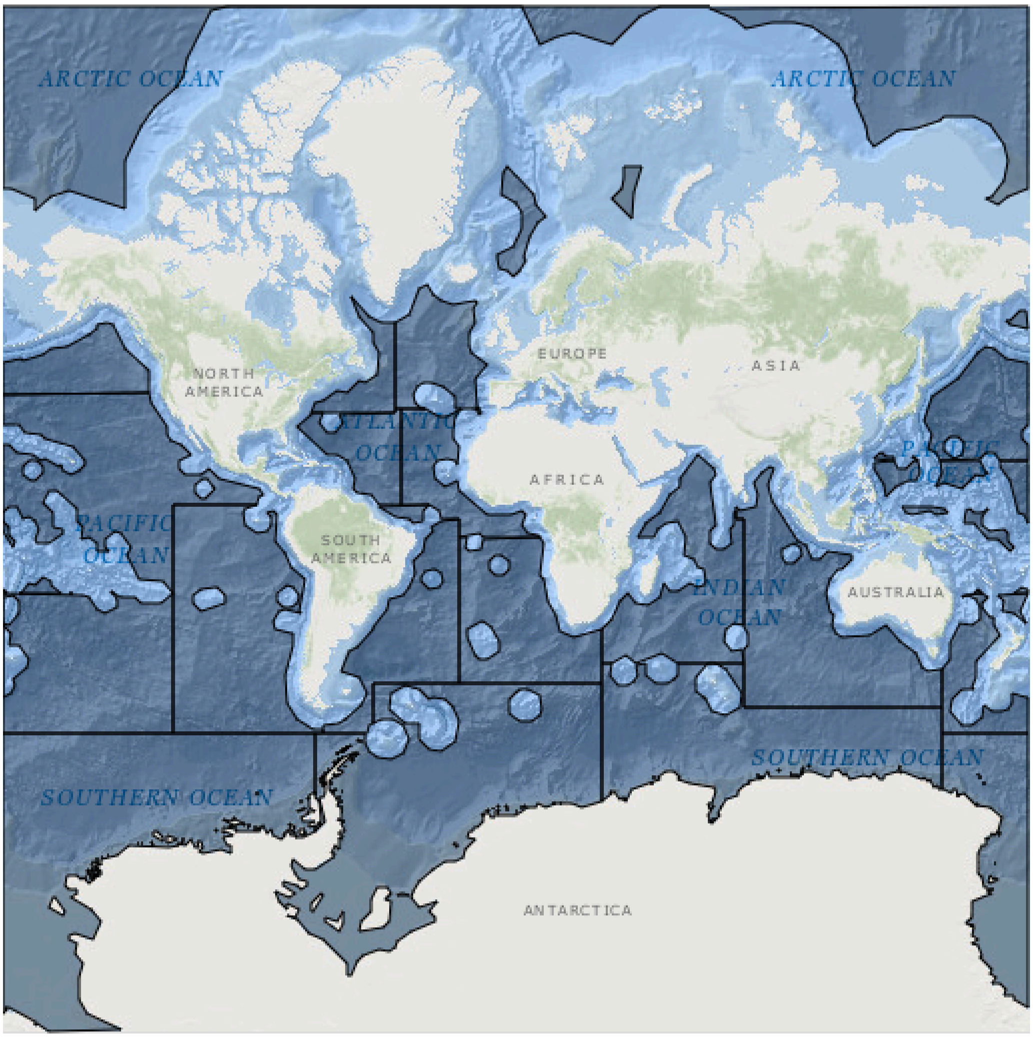
The dataset

The Sea Around Us website provides a dataset with extensive historical information about fisheries in all parts of every ocean globally. The data includes information about yearly fishery catches from 1950 to 2018.

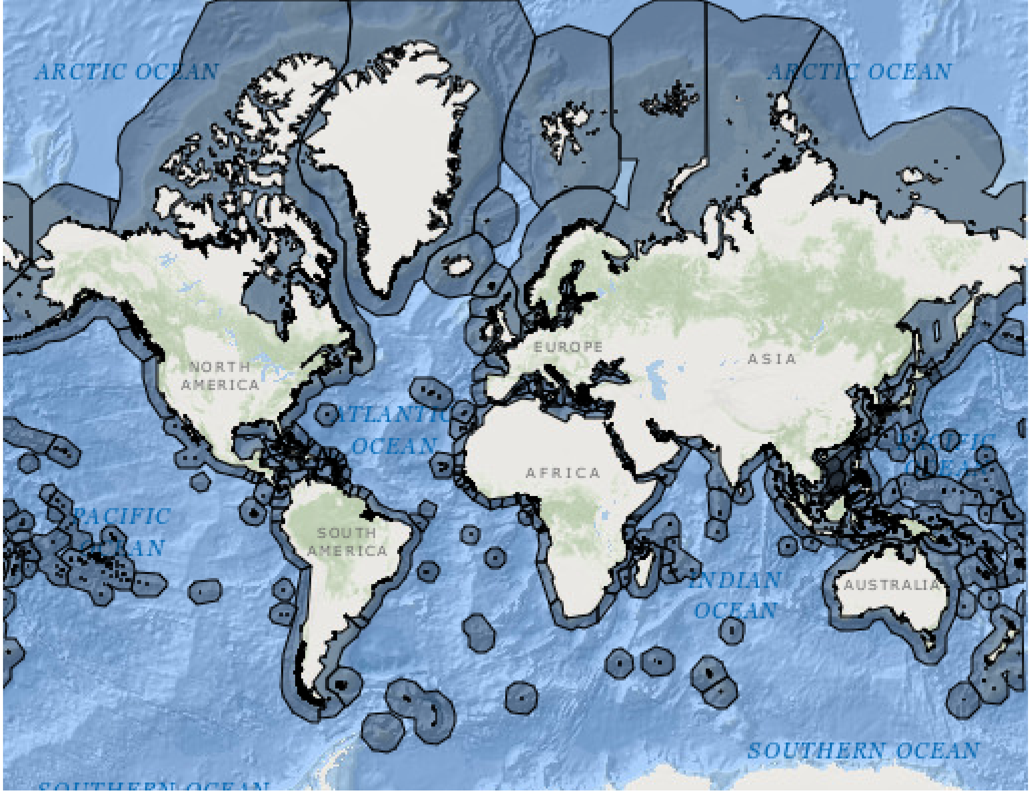
The data can be downloaded in CSV format from the Sea Around Us website. The dataset includes columns of information for each year, including which countries caught which types of fish in which areas. The data also indicates how many tonnes of fish were caught and what the value of the catch was, measured in 2010 US dollars.

To understand the data, it will be helpful to understand what is meant by *open seas* areas and *EEZ* areas:

- **Open seas (also called high seas):** Areas of the ocean that are at least 200 nautical miles away from any single country's shoreline. The resources, including the fish, in these areas are generally accepted as not belonging to any one country. The following map, which is a screen capture from the Sea Around Us website, shows how the dataset divides the high seas (areas highlighted in dark gray) into unique high seas areas.



- **Exclusive Economic Zones (EEZs):** Areas within 200 nautical miles of a country's shoreline. Each country typically claims exclusive access to the resources in the zones, including the fish within them. The following map, which is a screen capture from the Sea Around Us website, shows the EEZs of the world.



The Sea Around Us data are (where not otherwise regulated) under a Creative Commons Attribution Non-Commercial 4.0 International License (https://creativecommons.org/licenses/by-nc/4.0/). Notices regarding copyrights (© The University of British Columbia), license and disclaimer can be found under http://www.seaaroundus.org/terms-and-conditions/. Pauly D., Zeller D., Palomares M.L.D. (Editors), 2020. Sea Around Us Concepts, Design and Data (seaaroundus.org).

Scenario

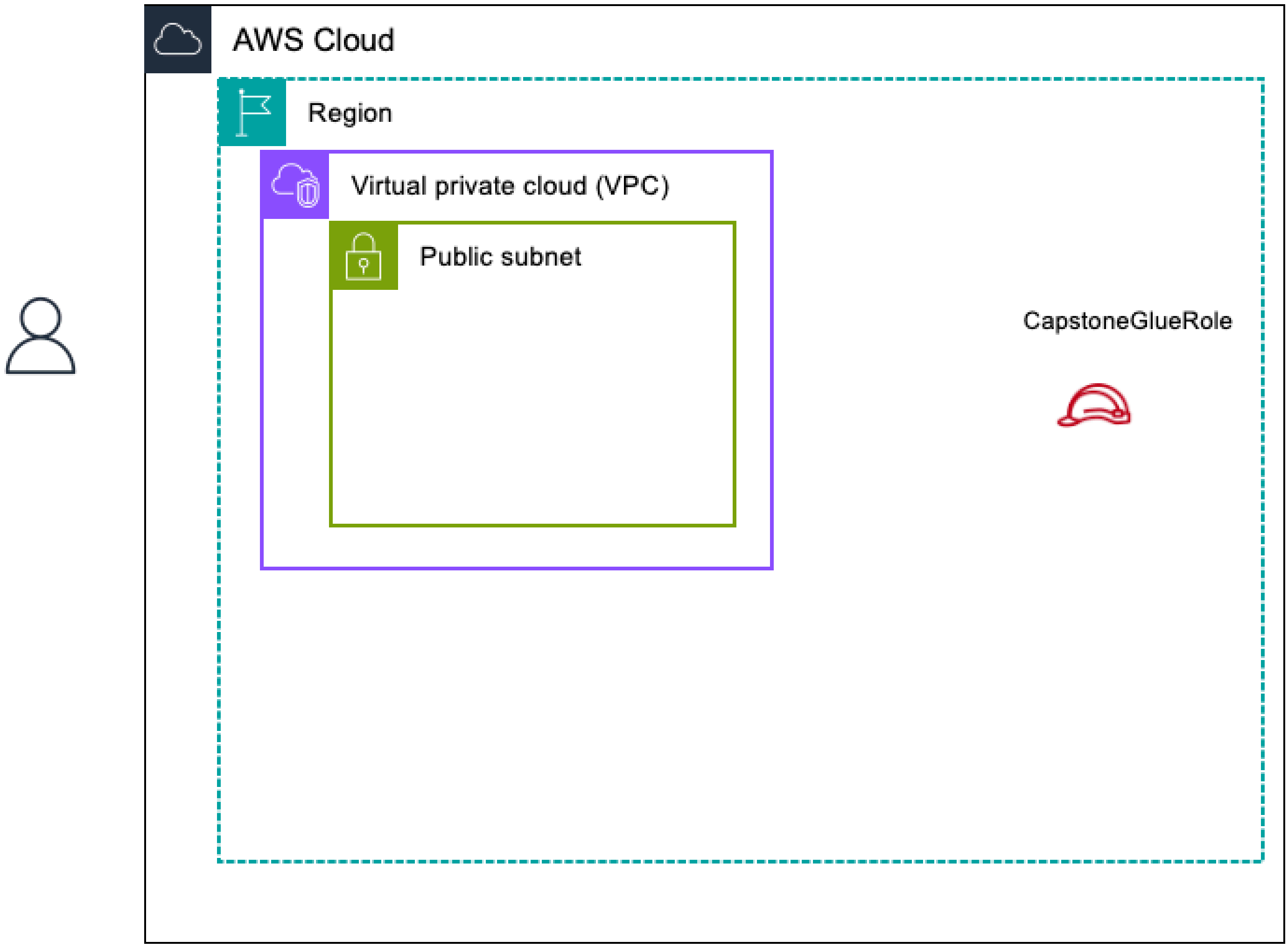
You have been tasked to create the infrastructure to host fishing data so that data analysts in your organization can create reports about fishing impact in the open seas. You have decided to build the infrastructure in your AWS account and test it by using three data files from the Sea Around Us dataset.

In this capstone project, you will work with three data files from the Sea Around Us website:

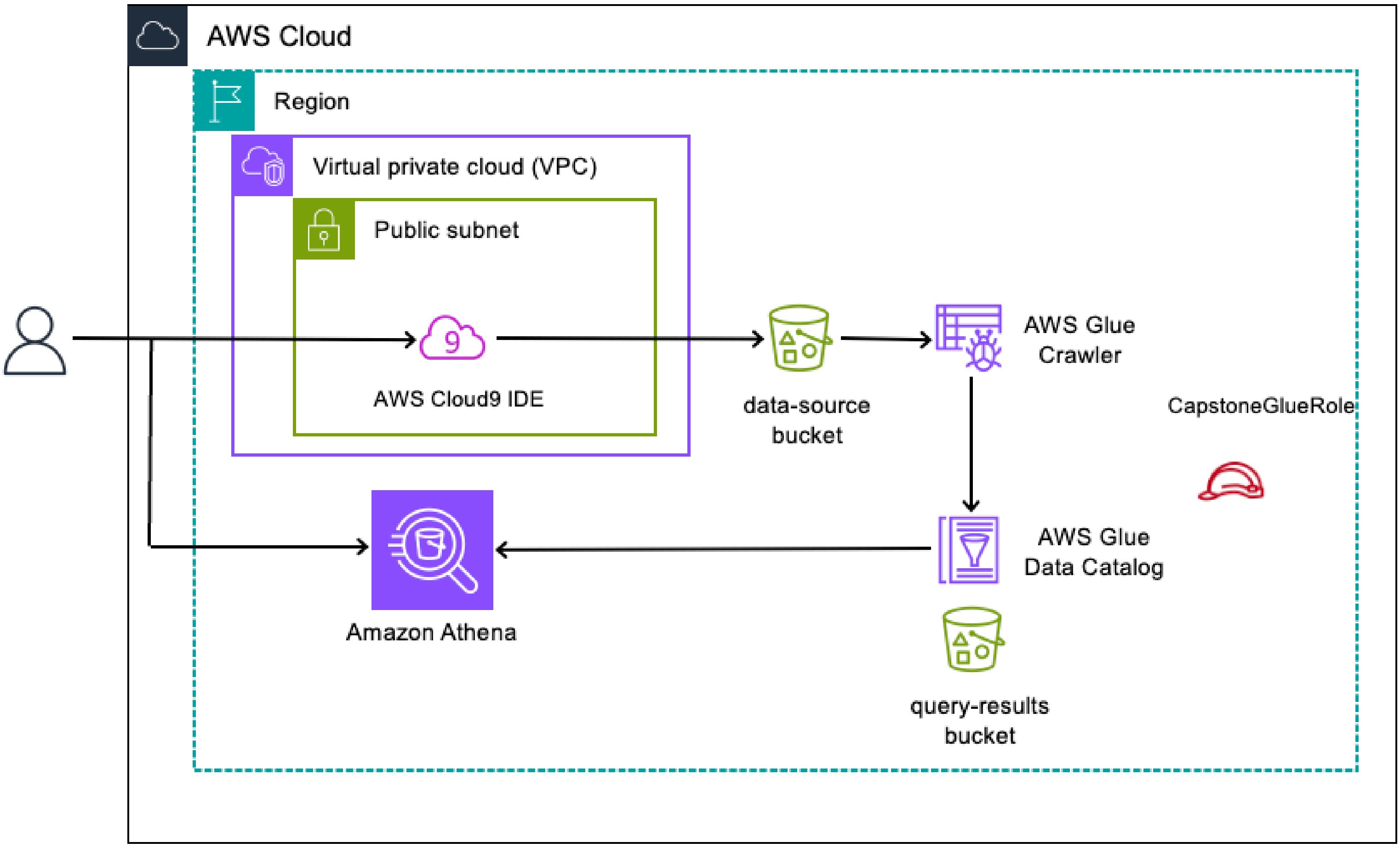
- The first file contains data from *all open seas areas*.
- The second file contains data from a *single open seas area* in the Pacific ocean, referred to as *Pacific, Western Central*, which is not far from Fiji and many other countries.
- The third file contains data from the *EEZ* of a single country (Fiji), which is near the Pacific, Western Central open seas area.

By working with this variety of sample data files, you will be able to test whether the solution that you build can support a much larger dataset.

When you start the capstone, the environment will contain the resources that are shown in the following diagram.



By the end of the capstone, you will have created an architecture that is similar to what is shown in the following diagram.



Accessing the AWS Management Console

1. At the top of these instructions, choose ▶ **Start Lab**.
   - The lab session starts.
   - A timer displays at the top of the page and shows the time remaining in the session.

     💡 **Tip:** To refresh the session length at any time, choose ▶ **Start Lab** again before the timer reaches 0:00.
   - Before you continue, wait until the circle icon to the right of the AWS ● link in the upper-left corner turns green.

2. To connect to the AWS Management Console, choose the **AWS** link in the upper-left corner.
   - A new browser tab opens and connects you to the console.

     💡 **Tip:** If a new browser tab does not open, a banner or icon is usually at the top of your browser with the message that your browser is preventing the site from opening pop-up windows. Choose the banner or icon, and then choose **Allow pop-ups**.

Task 1: Configuring the development environment

In this first part of the capstone, you will set up your development environment.

3. Observe the details of the *CapstoneGlueRole* AWS Identity and Access Management (IAM) role that has been created for you.

4. Create an AWS Cloud9 environment with the following settings:
   - Name the environment `CapstoneIDE`
   - Create a new EC2 instance for the environment, and use a t2.micro instance.
   - Deploy the instance to support *SSH* connections to the *Capstone VPC*, in the *Capstone public subnet*.
   - Keep all other default settings.

5. Create two S3 buckets with the following settings:
   - Create the buckets in the us-east-1 Region.
   - Name the first bucket `data-source-#####` where ##### is a random number.
   - Name the second bucket `query-results-#####` where ##### is also a random number.
   - Keep all other default settings.

6. To download the three .csv source data files, run the following commands in the terminal of your AWS Cloud9 IDE:

```
wget https://aws-tc-largeobjects.s3.us-west-2.amazonaws.com/CUR-TF-200-ACDENG-1-91570/lab-capstone/s3/SAU-GLOBAL-1-v48-0.csv
wget https://aws-tc-largeobjects.s3.us-west-2.amazonaws.com/CUR-TF-200-ACDENG-1-91570/lab-capstone/s3/SAU-HighSeas-71-v48-0.csv
wget https://aws-tc-largeobjects.s3.us-west-2.amazonaws.com/CUR-TF-200-ACDENG-1-91570/lab-capstone/s3/SAU-EEZ-242-v48-0.csv
```

7. To observe the column header row and the first five rows of data in the SAU-GLOBAL-1-v48-0.csv file, run the following command:

   ```
   head -6 SAU-GLOBAL-1-v48-0.csv
   ```

   Analysis: Among other details, each line of data in this dataset includes:

   - The *year* that the fishing occurred
   - The country (*fishing_entity*) that did the fishing
   - The *tonnes* of fish caught that year by that country
   - The value in 2010 US dollars (*landed_value*) of the fish caught that year by that country

   Note:

   - This dataset contains 561,675 lines.

     Tip: To confirm this, run run `wc -l <filename>`.

     The dataset includes reported and "best guess" data for all fishing that occurred in the global high seas (meaning, *not* in any one country's EEZ) between 1950 and 2018.

   - EEZ areas include the ocean waters within 200 nautical miles of the shoreline of a country. Therefore, the fishing reported in this dataset occurred at least 200 miles offshore from any country.

8. Convert the SAU-GLOBAL-1-v48-0.csv file to Parquet format.

   First, you need to install some tools on your AWS Cloud9 IDE. Run the following command:

   ```
   sudo pip3 install pandas pyarrow fastparquet
   ```

   Next, to convert the file to Parquet format, run the following code:

   ```
   # Start the python interactive shell
   python3
   # Use pandas to convert the file to parquet format
   import pandas as pd
   df = pd.read_csv('SAU-GLOBAL-1-v48-0.csv')
   df.to_parquet('SAU-GLOBAL-1-v48-0.parquet')
   exit()
   ```

   Note: Pandas is a useful tool for working with data files. For more information, see the pandas website.

9. To upload the *SAU-GLOBAL-1-v48-0.parquet* file to the *data-source* bucket, use an AWS Command Line Interface (AWS CLI) command in your AWS Cloud9 terminal.

Task 2: Using an AWS Glue crawler and querying multiple files with Athena

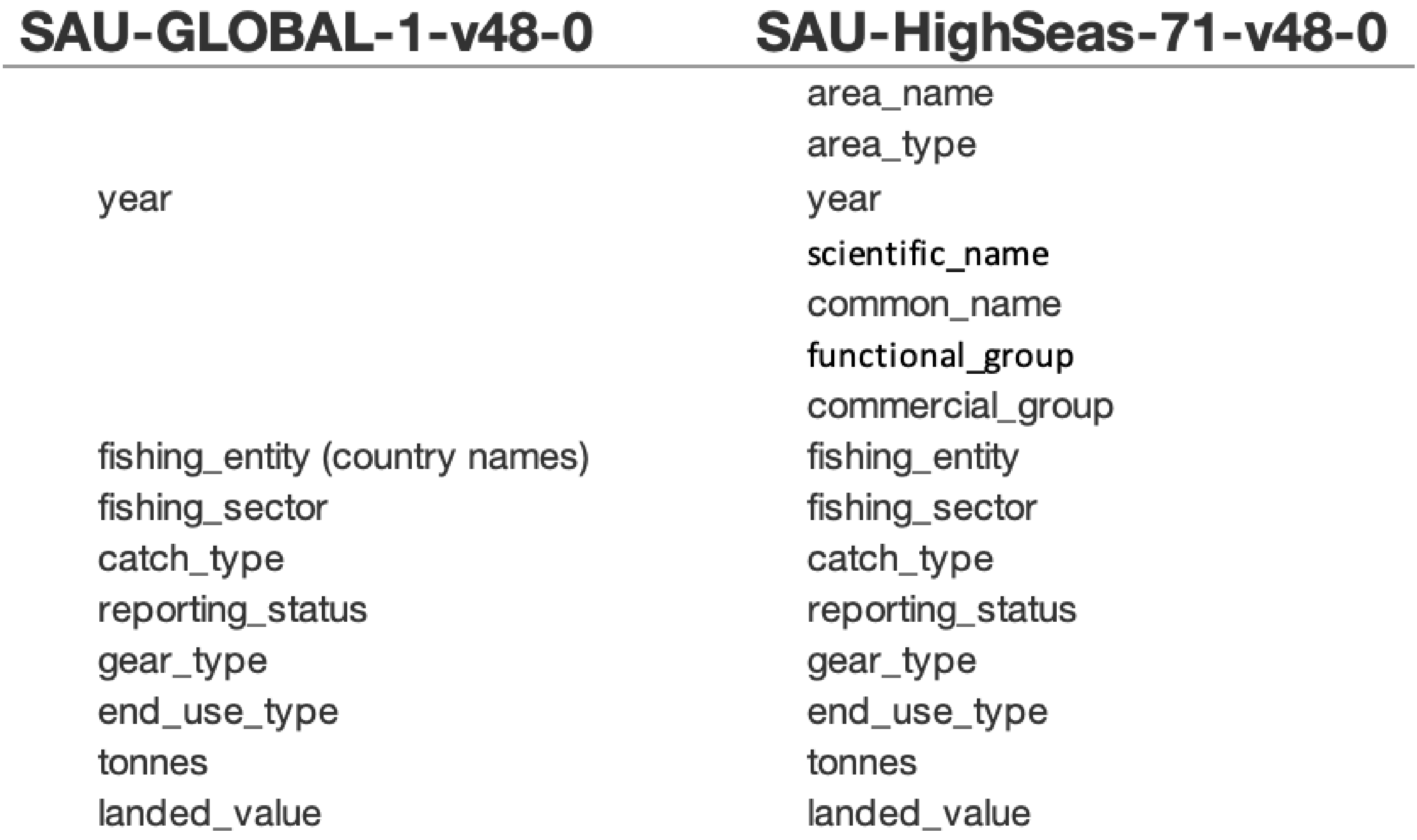
The query that you ran in the previous task works well for a single data file. However, what if you need to query a larger dataset that consists of more than one file?

In this second part of the capstone, you will query data that is stored in multiple files. To do this, you will configure an AWS Glue crawler to discover the structure of the data and then use Athena to query the data.

10. To observe the column header row and first few lines of data from the *SAU-HighSeas-71-v48-0.csv* file, use the *head* command.

    Recall that you already downloaded the file to your AWS Cloud9 IDE.

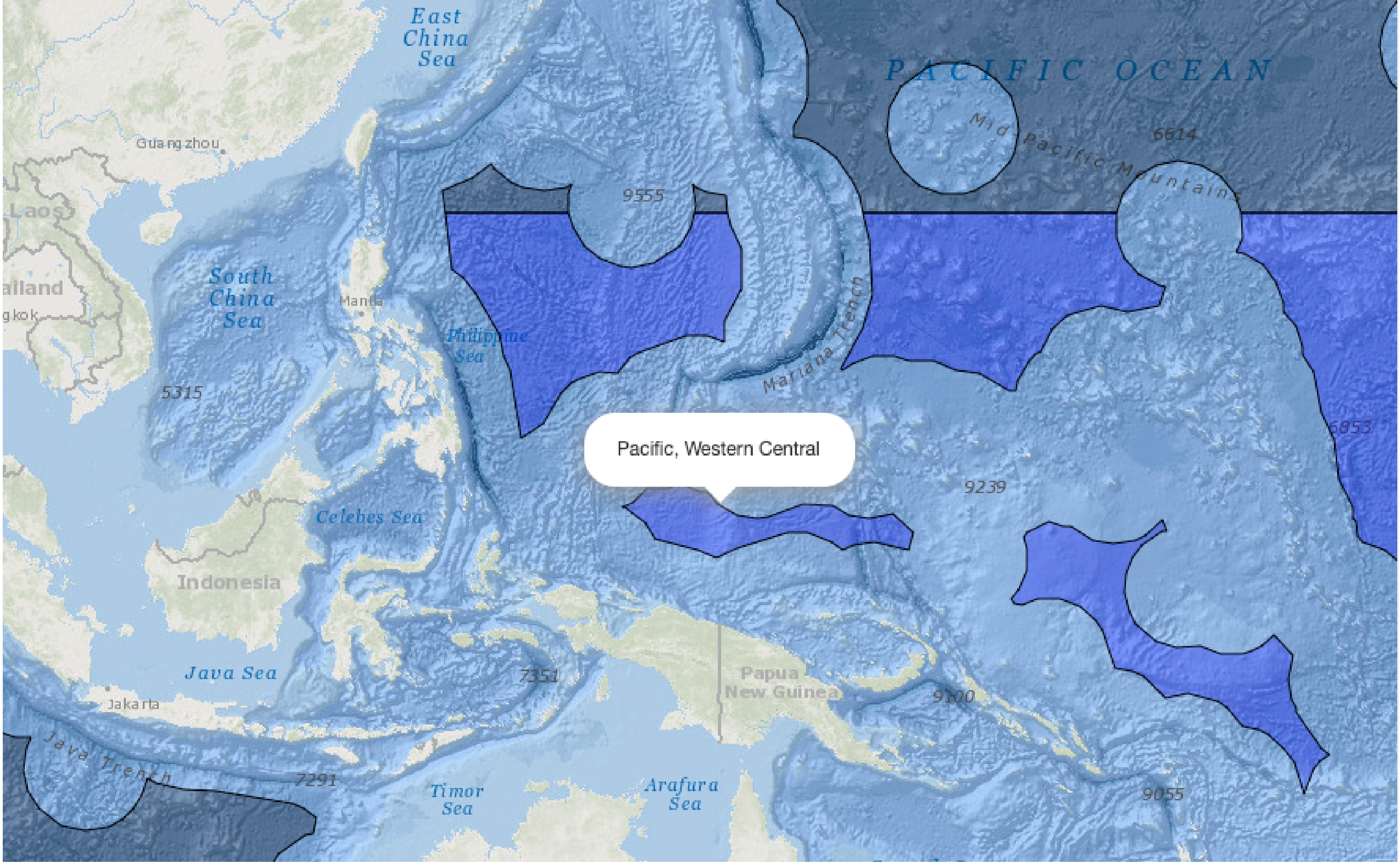
    The file contains the same columns as the *SAU-GLOBAL-1-v48-0.csv* file but has additional columns. The following chart shows the columns that are contained in each file.



SAU-GLOBAL-1-v48-0	SAU-HighSeas-71-v48-0
	area_name
	area_type
year	year
	scientific_name
	common_name
	functional_group
	commercial_group
fishing_entity (country names)	fishing_entity
fishing_sector	fishing_sector
catch_type	catch_type
reporting_status	reporting_status
gear_type	gear_type
end_use_type	end_use_type
tonnes	tonnes
landed_value	landed_value

Analysis:

- Like the *SAU-GLOBAL-1-v48-0* dataset that you already uploaded to Amazon S3 and queried, the *SAU-HighSeas-71-v48-0* dataset also describes fish catches in the high seas. However, the HighSeas dataset includes data only from one high seas area, known as *Pacific, Western Central*. This area is highlighted in the following screen capture from the Sea Around Us website.



- Of the additional columns in the HighSeas dataset, two are of particular interest:
  - The *area_name* column contains the "Pacific, Western Central" value in every row.
  - The *common_name* column contains values that describe certain types of fish (for example, "Mackerels, tunas, bonitos").

11. Convert the *SAU-HighSeas-71-v48-0.csv* file to Parquet format and upload it to the *data-source* bucket.

12. Create an AWS Glue database and an AWS Glue crawler with the following settings:
    - Name the database `fishdb`
    - Name the crawler `fishcrawler`
    - Configure the crawler to use the *CapstoneGlueRole* IAM role to crawl the contents of the *data-source* S3 bucket.
    - Output the results of the crawler to the *fishdb* database.
    - Set the crawler frequency to *On demand*.

13. Run the crawler to create a table that contains metadata in the AWS Glue database.

    Verify that the expected table is created.

14. To confirm that the table properly categorized the data, use Athena to run SQL queries against each column in the new table.

    Important: Before you run the first query in Athena, configure the Athena Query Editor to output data to the *query-results* bucket.

    Example query:

```
SELECT DISTINCT area_name FROM fishdb.data_source_xxxxx;
```

Note: The example query returns two results. For this column, every row in the dataset contains either the value "Pacific, Western Central" (for rows pulled from *SAU-HighSeas-71-v48-0.parquet*) or a null value (for rows pulled from *SAU-GLOBAL-1-v48-0.parquet*).

15. Now that your data table is defined, run queries to confirm that it provides useful results.
    - To find the value in US dollars of all fish caught by the country Fiji from the Pacific, Western Central high seas area since 2001, organized by year, use the following query (be sure to replace `<FMI_1>` and `<FMI_2>` with the proper values) :

      ```
      SELECT year, fishing_entity AS Country, CAST(CAST(SUM(landed_value) AS DOUBLE) AS DECIMAL(38,2)) AS
      ValuePacificWCSeasCatch
      FROM <FMI_1>
      WHERE area_name LIKE '%Pacific%' and fishing_entity='Fiji' AND year > <FMI_2>
      GROUP BY year, fishing_entity
      ORDER By year
      ```

      Note: The *CAST(CAST(sum(landed_value) AS DOUBLE) AS DECIMAL(38,2))* part of the query ensures that the format of the returned data from the *landed_value* column displays in a reader-friendly format (dollars and cents) instead of scientific format.
    - **Challenge:** Find the value in US dollars of all fish caught by the country Fiji from all high seas areas since 2001, organized by year. In your output results, name the US dollar value column `ValueAllHighSeasCatch`

      Helpful hints for the challenge:
      - Your WHERE clause should include two `AND` keywords.
      - To return rows that don't contain an entry for a particular column, use `IS NULL`
    - After you create and run the correct query and see the results displayed, create a view based on the query:
      - Choose **Create** > **View from query**.
      - Name the view `challenge`

Task 3: Transforming a new file and adding it to the dataset

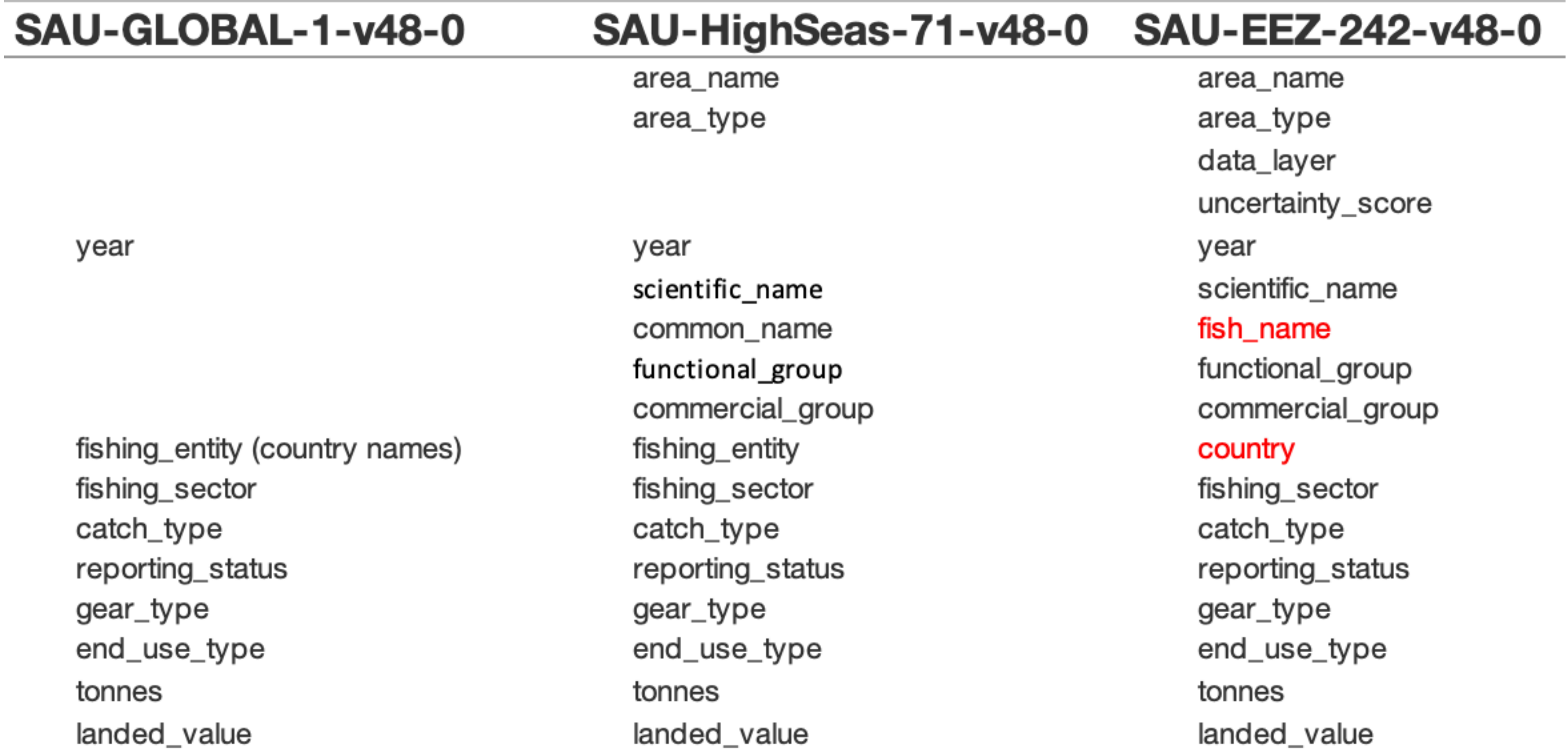
In this part of the capstone, you will add the *SAU-EEZ-242-v48-0.csv* data file to your dataset in Amazon S3.

This file has a few column names that don't match the data that you already added to your S3 bucket. However, the data in these columns *does* align with the existing data. You will need to modify the column names before you add the data to your bucket.

16. Analyze the data structure of *SAU-EEZ-242-v48-0.csv* file. Compare the columns that it contains with the columns that the other data files contain.

Use the same technique that you used earlier in the lab to discover the column names for the EEZ file.

Tip: Most of the column names match between the three files. However, two of the column names in the EEZ file are *not* an exact match. The following chart shows the columns that are contained in each file.



SAU-GLOBAL-1-v48-0	SAU-HighSeas-71-v48-0	SAU-EEZ-242-v48-0
	area_name	area_name
	area_type	area_type
		data_layer
		uncertainty_score
year	year	year
	scientific_name	scientific_name
	common_name	fish_name
	functional_group	functional_group
	commercial_group	commercial_group
fishing_entity (country names)	fishing_entity	country
fishing_sector	fishing_sector	fishing_sector
catch_type	catch_type	catch_type
reporting_status	reporting_status	reporting_status
gear_type	gear_type	gear_type
end_use_type	end_use_type	end_use_type
tonnes	tonnes	tonnes
landed_value	landed_value	landed_value

Analysis: The data in the *fish_name* column needs to be merged with the data in the *common_name* column from the HighSeas dataset. Likewise, the data in the *country* column needs to be merged with the data in the *fishing_entity* column from the HighSeas dataset.

17. Use the Python data analyst library, which is called pandas, to fix the column names. In addition, convert the EEZ file to the Parquet format.

    To accomplish these tasks, run all of the commands in the following code block.

    However, before you run the `df.rename` command, replace the **<FMI_#>** placeholders with the correct values.

    Tips:

    - For example `<FMI_1>` should be set to one of the column names you want to change and `<FMI_2>` should be set to what you want to change it to.
    - It will be easier to read the output of the **print** lines if you make your browser window as wide as possible.

```
# Make a backup of the file before you modify it in place
cp SAU-EEZ-242-v48-0.csv SAU-EEZ-242-v48-0-old.csv

# Start the python interactive shell
python3
import pandas as pd

# Load the backup version of the file
data_location = 'SAU-EEZ-242-v48-0-old.csv'

# Use Pandas to read the CSV into a dataframe
df = pd.read_csv(data_location)

# View the current column names
print(df.head(1))

# Change the names of the 'fish_name' and 'country' columns to match the column names where this data appears in the other data files already in your data-source bucket
df.rename(columns = {"<FMI_1>": "<FMI_2>", "<FMI_3>": "<FMI_4>"}, inplace = True)

# Verify the column names have been changed
```

```
print(df.head(1))

# Write the changes to disk
df.to_csv('SAU-EEZ-242-v48-0.csv', header=True, index=False)
df.to_parquet('SAU-EEZ-242-v48-0.parquet')
exit()
```

18. Upload the new EEZ data file to the *data-source* bucket.

19. To update the table metadata with the additional columns that are now part of your dataset, run the AWS Glue crawler again.

20. Run some queries in Athena.

    Note: For all of these queries, replace **data_source_#####** with the name of your *data_source* table.

    - To verify the values in the *area_name* column, as you did before, use the following query:

      ```
      SELECT DISTINCT area_name FROM fishdb.data_source_#####;
      ```

      With the addition of the EEZ file to the dataset, this query now returns three results, including the result for rows where the *area_name* column doesn't have any data. (Recall that this query returned only two results previously.)

    - To find the value in US dollars of all fish caught by Fiji *from the open seas* since 2001, organized by year, use the following query:

      ```
      SELECT year, fishing_entity AS Country, CAST(CAST(SUM(landed_value) AS DOUBLE) AS DECIMAL(38,2)) AS
      ValueOpenSeasCatch
      FROM fishdb.data_source_#####
      WHERE area_name IS NULL AND fishing_entity='Fiji' AND year > 2000
      GROUP BY year, fishing_entity
      ORDER By year
      ```

    - To find the value in US dollars of all fish caught by Fiji *from the Fiji EEZ* since 2001, organized by year, use the following query:

      ```
      SELECT year, fishing_entity AS Country, CAST(CAST(SUM(landed_value) AS DOUBLE) AS DECIMAL(38,2)) AS
      ValueEEZCatch
      FROM fishdb.data_source_#####
      WHERE area_name LIKE '%Fiji%' AND fishing_entity='Fiji' AND year > 2000
      GROUP BY year, fishing_entity
      ORDER By year
      ```

    - To find the value in US dollars of all fish caught by Fiji *from either the Fiji EEZ or the open seas* since 2001, organized by year, use the following query:

      ```
      SELECT year, fishing_entity AS Country, CAST(CAST(SUM(landed_value) AS DOUBLE) AS DECIMAL(38,2)) AS
      ValueEEZAndOpenSeasCatch
      FROM fishdb.data_source_#####
      WHERE (area_name LIKE '%Fiji%' OR area_name IS NULL) AND fishing_entity='Fiji' AND year > 2000
      GROUP BY year, fishing_entity
      ORDER By year
      ```

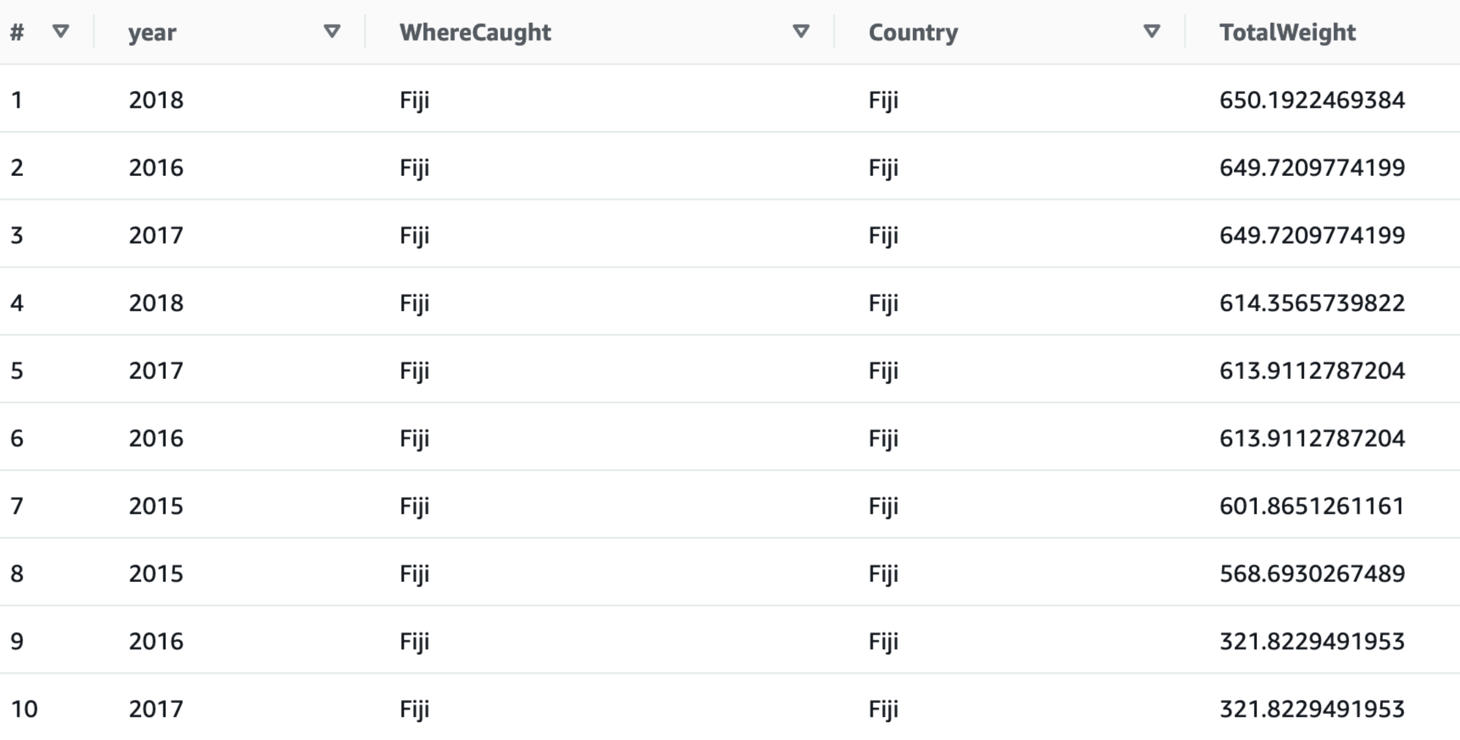
      Analysis: If your data is formatted well and the AWS Glue crawler properly updated the metadata table, then the results that you get from the first two queries in this step should add up to the results that you get from the third query.

For example, if you add the *2001 ValueOpenSeasCatch* value and the *2001 ValueEEZCatch* value, the total should equal the *2001 ValueEEZAndOpenSeasCatch* value. If your results are consistent with this description, then it is a good indication that your solution is working as intended.

21. Create a view in Athena, which will be useful to review the data in the next section of this capstone.
    - Run the following query. Replace **data_source_#####** with the name of your *data_source* table:

```
CREATE OR REPLACE VIEW MackerelsCatch AS
SELECT year, area_name AS WhereCaught, fishing_entity as Country, SUM(tonnes) AS TotalWeight
FROM fishdb.data_source_#####
WHERE common_name LIKE '%Mackerels%' AND year > 2014
GROUP BY year, area_name, fishing_entity, tonnes
ORDER BY tonnes DESC
```

    - To verify that the view has data, in the **Data** panel, under **Views**, choose the ellipsis (three dot) icon to the right of the **mackerelscatch** view, and choose **Preview View**. You should see the output similar to the following.



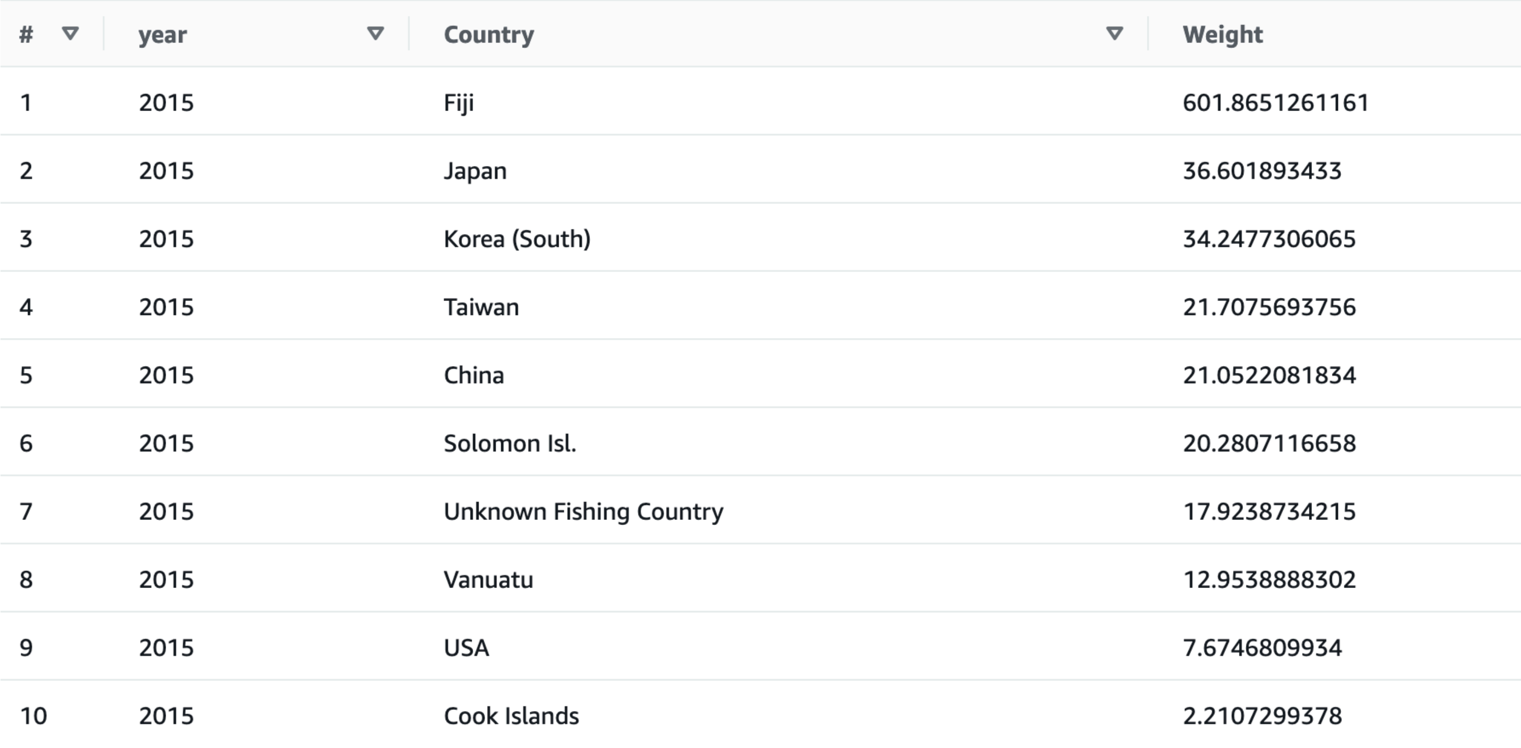
#	year	WhereCaught	Country	TotalWeight
1	2018	Fiji	Fiji	650.1922469384
2	2016	Fiji	Fiji	649.7209774199
3	2017	Fiji	Fiji	649.7209774199
4	2018	Fiji	Fiji	614.3565739822
5	2017	Fiji	Fiji	613.9112787204
6	2016	Fiji	Fiji	613.9112787204
7	2015	Fiji	Fiji	601.8651261161
8	2015	Fiji	Fiji	568.6930267489
9	2016	Fiji	Fiji	321.8229491953
10	2017	Fiji	Fiji	321.8229491953

Next you run few queries on the Athena view to analyze the data.

22. To view the following data *Tonnes of mackerel caught by year by country*
    - Return to the Athena query editor, and run the following SQL query to identify the countries with the highest mackerel catch each year.

```
SELECT year, Country, MAX(TotalWeight) AS Weight
FROM fishdb.mackerelscatch
GROUP BY year, Country
ORDER BY year, Weight DESC;
```

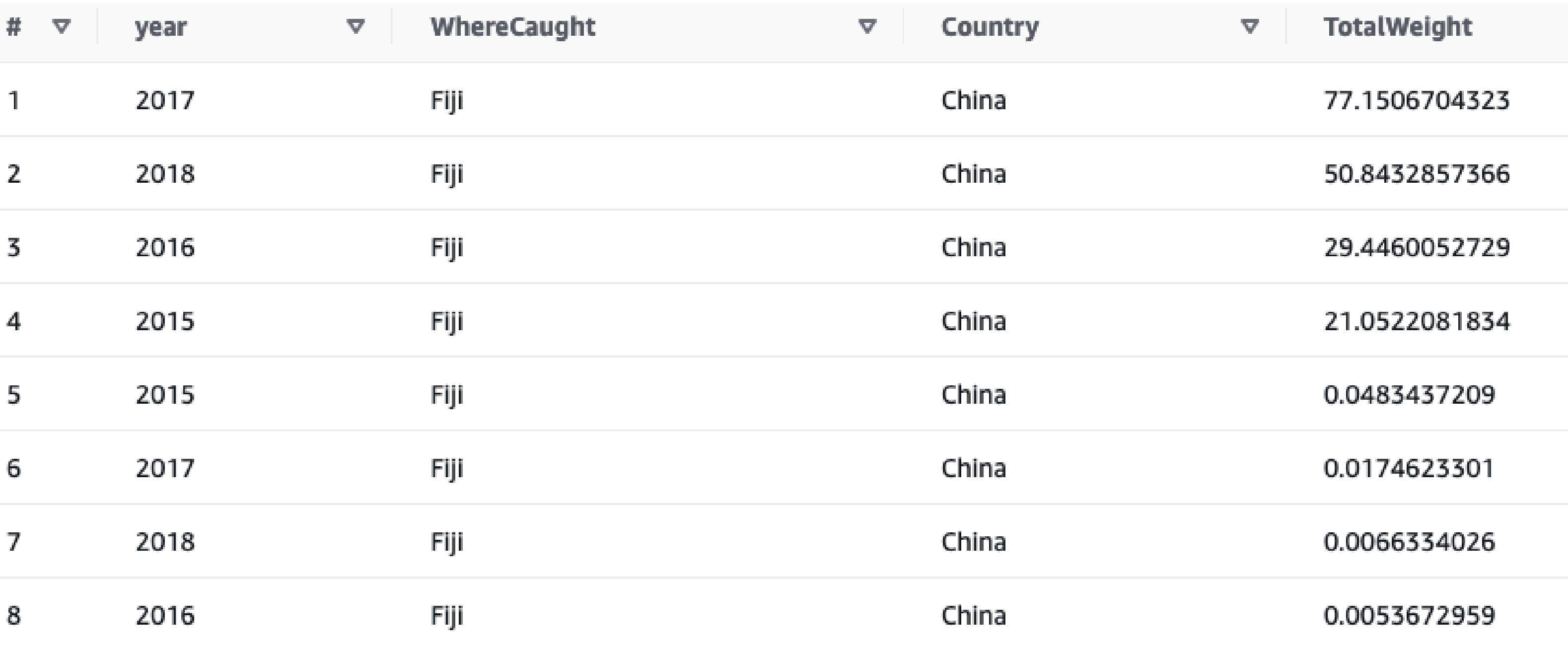
You should see the output similar to the following.



#	year	Country	Weight
1	2015	Fiji	601.8651261161
2	2015	Japan	36.601893433
3	2015	Korea (South)	34.2477306065
4	2015	Taiwan	21.7075693756
5	2015	China	21.0522081834
6	2015	Solomon Isl.	20.2807116658
7	2015	Unknown Fishing Country	17.9238734215
8	2015	Vanuatu	12.9538888302
9	2015	USA	7.6746809934
10	2015	Cook Islands	2.2107299378

23. To view the *MackerelsCatch* for a particular country, e.g `China`, run the following query.

```
SELECT * FROM "fishdb"."mackerelscatch"
where country in ('China')
```



#	year	WhereCaught	Country	TotalWeight
1	2017	Fiji	China	77.1506704323
2	2018	Fiji	China	50.8432857366
3	2016	Fiji	China	29.4460052729
4	2015	Fiji	China	21.0522081834
5	2015	Fiji	China	0.0483437209
6	2017	Fiji	China	0.0174623301
7	2018	Fiji	China	0.0066334026
8	2016	Fiji	China	0.0053672959

Note: You may run additional queries on the view or table to get more insights on the data.

Submitting your work

24. To record your progress, choose **Submit** at the top of these instructions.
25. When prompted, choose **Yes**.

    After a couple of minutes, the grades panel appears and shows you how many points you earned for each task. If the results don't display after a couple of minutes, choose **Grades** at the top of these instructions.

    Tip: You can submit your work multiple times. After you change your work, choose **Submit** again. Your last submission is recorded for this lab.

26. To find detailed feedback about your work, choose **Submission Report**.

Ending your session

Reminder: This is a long-lived lab environment. Data is retained until you either use the allocated budget or the course end date is reached (whichever occurs first).

To preserve your budget when you are finished for the day, or when you are finished actively working on the assignment for the time being, do the following:

27. At the top of this page, choose ■ **End Lab**, and then choose **Yes** to confirm that you want to end the lab.

    A message panel indicates that the lab is terminating.

    Note: Choosing **End lab** in this capstone environment will *not* delete the resource you have created. They will still be there the next time you choose Start lab (for example, on another day).

28. To close the panel, choose **Close** in the upper-right corner.