# **Chinook Database project**

## **Исследуйте данные музыкального магазина с помощью SQL!**

### **Обзор проекта**

В этом проекте вы будете выполнять запросы для базы данных Chinook. Chinook содержит информацию о музыкальном магазине.

Ваши задачи:

* использовать основные функционалы SQL
* отвечать на вопросы с помощью правильных запросов
* найти интересные инсайты в данных используя JOIN-ы и подзапросы
* переносить данные с одной платформы в другую
* получить удовольствие от процесса

### **Ваша роль**

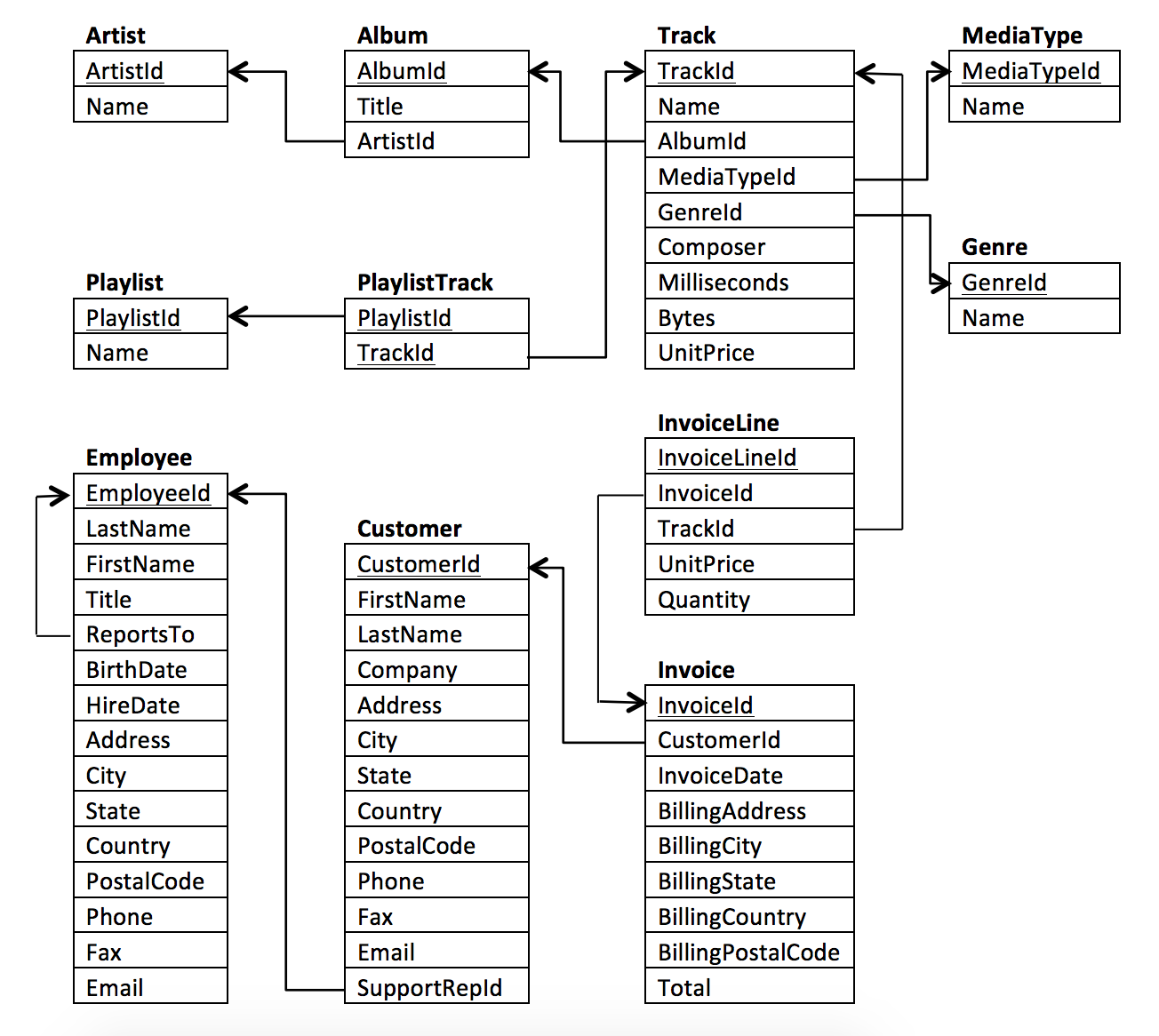
В рамках этого проекта вы будете помогать команде Chinook разобраться в товарах в их магазине, их покупателях и сотрудниках, а также в информации об их счетах.

### **База данных**

Чтобы помочь вам с дальнейшими запросами, ниже представлена схема базы данных Chinook. Вы можете увидеть столбцы, которые связывают таблицы вместе с помощью стрелок. (Hint: База данных Chinook уже есть в BigQuery, ничего создавать не нужно).

Для работы с базой данных, перенесите все данные с Google Sheets в BigQuery.

Изучите таблицы и данные в таблицах перед тем как начать работать над проектом.



## **Инструкции по сдаче проекта**

* После выполнения задания нажмите на кнопку "+ Добавить или создать" в Google Classroom и сохранить файл Google sheets + Google docs
* После загрузки файлов нажмите на кнопку "Сдать".
* Ожидайте обратную связь от менторов.

**Часть 1**

#### **Вопрос 1: В каких странах больше всего счетов (Invoices)?**

Используйте таблицу **Invoice**, чтобы определить страны с наибольшим количеством счетов. Предоставьте таблицу со столбцами **BillingCountry** и **Invoices**, упорядоченную по количеству счетов для каждой страны. Страна с наибольшим количеством счетов должна отображаться первой.

Ваш запрос:

| SELECT BillingCountry,  COUNT(\*) AS Invoices  FROM da-nfactorial.chinook.invoice  GROUP BY BillingCountry  ORDER BY Invoices DESC; |
| --- |

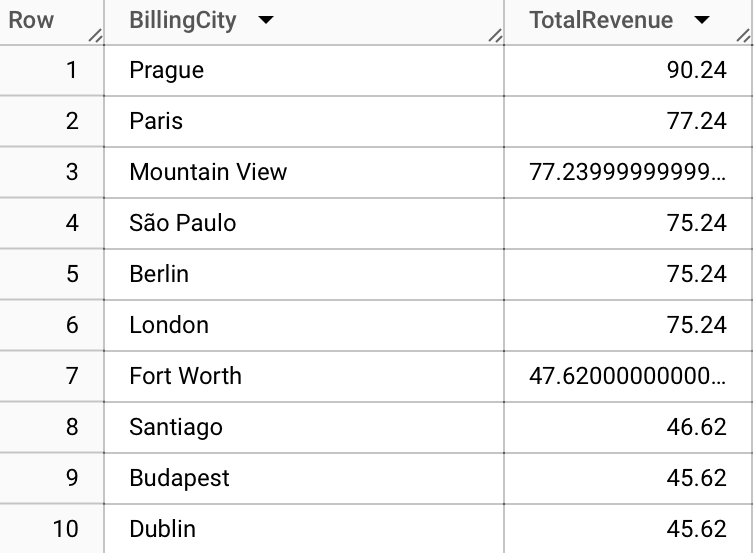


#### **Вопрос 2: В каком городе самые лучшие клиенты?**

Мы хотели бы устроить рекламный музыкальный фестиваль в городе, в котором мы заработали больше всего денег. Напишите запрос, который возвращает город с наибольшей суммой итоговых счетов (invoice totals). Предоставьте название города и общую сумму итоговых счетов.

Ваш запрос:

| SELECT BillingCity, SUM(Total) AS TotalRevenue  FROM `da-nfactorial.chinook.invoice`  GROUP BY BillingCity  ORDER BY TotalRevenue DESC; |
| --- |



#### **Вопрос 3: Кто является лучшим клиентом?**

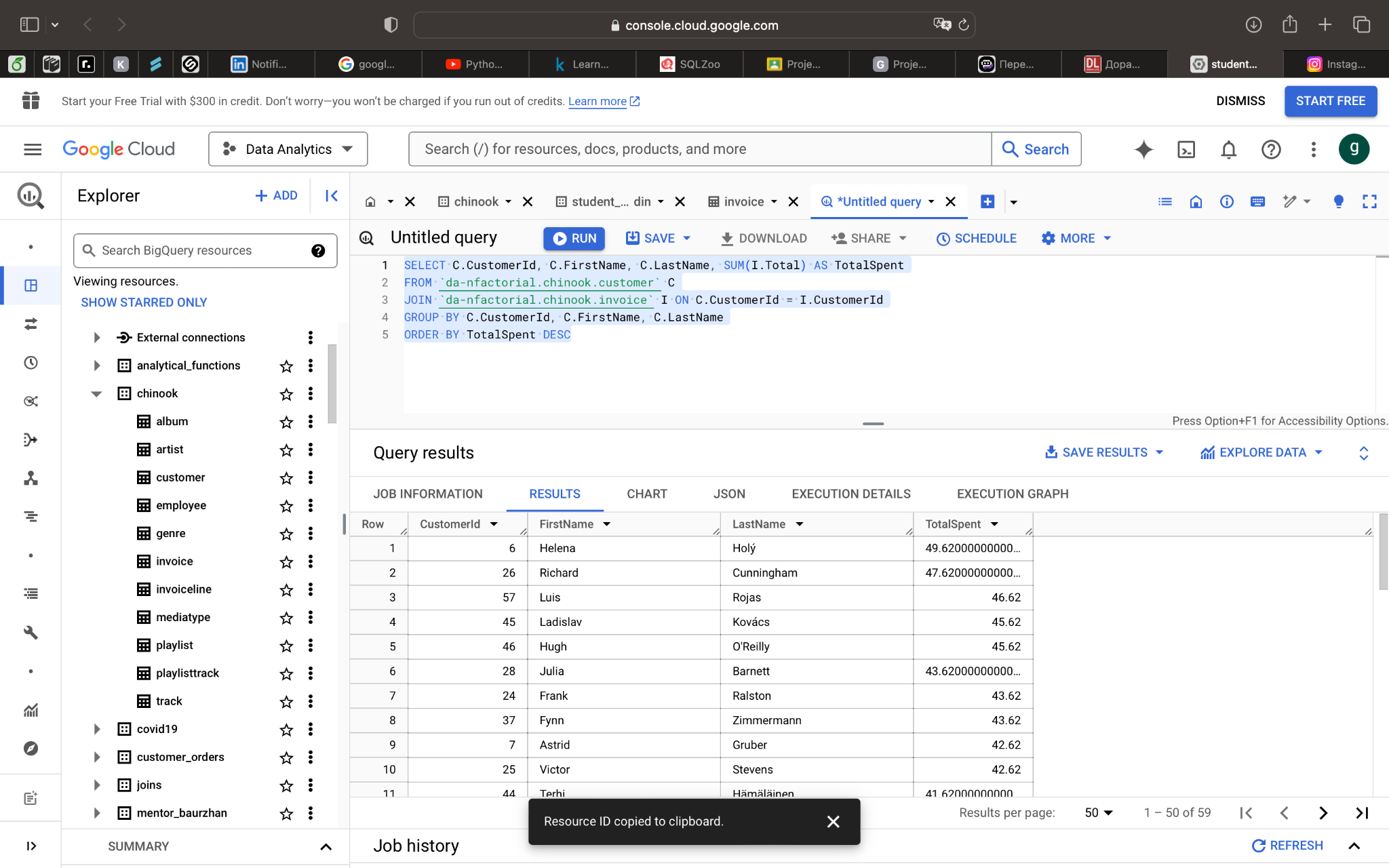
Клиент, потративший больше всего денег, будет объявлен лучшим клиентом. Напишите запрос, который возвращает человека, который потратил больше всего денег. Мы нашли решение, связав следующие три таблицы: **Invoice, InvoiceLine** и **Customer**, чтобы получить эту информацию, но вы, вероятно, можете сделать это с меньшим количеством таблиц.

Ваш запрос:

| SELECT C.CustomerId, C.FirstName, C.LastName, SUM(I.Total) AS TotalSpent  FROM `da-nfactorial.chinook.customer` C  JOIN `da-nfactorial.chinook.invoice` I ON C.CustomerId = I.CustomerId  GROUP BY C.CustomerId, C.FirstName, C.LastName  ORDER BY TotalSpent DESC |
| --- |

#### 

#### 



#### **Вопрос 4: Кто слушает рок-музыку?**

Напишите запрос, чтобы вернуть адрес электронной почты, имя, фамилию и жанр всех слушателей рок-музыки. Верните свой список в алфавитном порядке по адресу электронной почты.

| SELECT C.Email, C.FirstName, C.LastName, G.Name AS Genre  FROM `da-nfactorial.chinook.customer` C  JOIN `da-nfactorial.chinook.invoice` I ON C.CustomerId = I.CustomerId  JOIN `da-nfactorial.chinook.invoiceline` IL ON I.InvoiceId = IL.InvoiceId  JOIN `da-nfactorial.chinook.track` T ON IL.TrackId = T.TrackId  JOIN `da-nfactorial.chinook.genre` G ON T.GenreId = G.GenreId  WHERE G.Name = 'Rock'  ORDER BY C.Email; |
| --- |

#### 

#### 

#### 

#### 

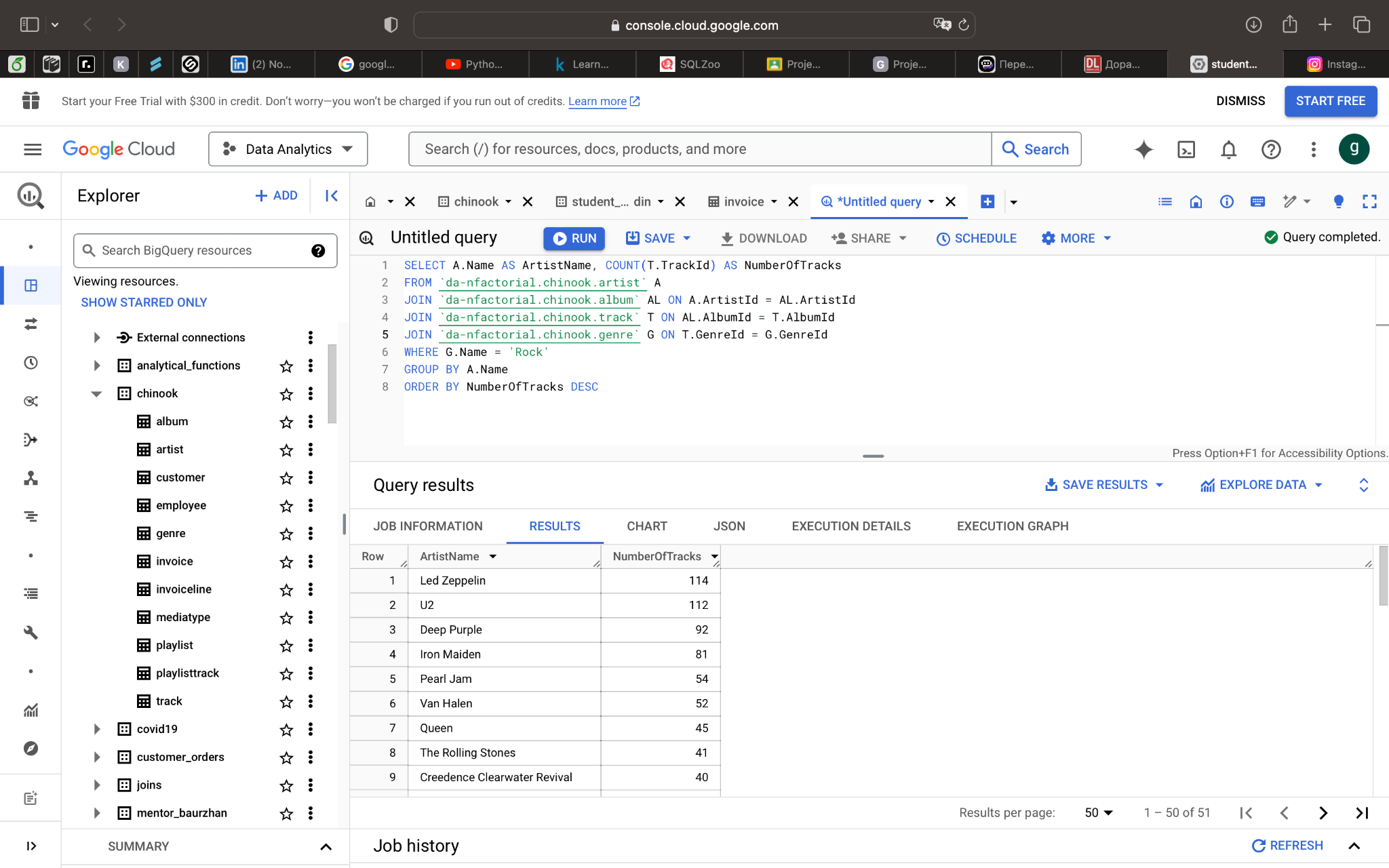
#### **Вопрос 5: Кто пишет рок-музыку?**

Теперь, когда мы знаем, что наши клиенты любят рок-музыку, мы можем решить, каких музыкантов пригласить сыграть на концерте.

Давайте пригласим артистов, которые написали больше всего рок-музыки в нашем наборе данных. Напишите запрос, который возвращает имя исполнителя и общее количество треков топ 10 рок-групп по количеству треков.

Ваш запрос:

| SELECT A.Name AS ArtistName, COUNT(T.TrackId) AS NumberOfTracks  FROM `da-nfactorial.chinook.artist` A  JOIN `da-nfactorial.chinook.album` AL ON A.ArtistId = AL.ArtistId  JOIN `da-nfactorial.chinook.track` T ON AL.AlbumId = T.AlbumId  JOIN `da-nfactorial.chinook.genre` G ON T.GenreId = G.GenreId  WHERE G.Name = 'Rock'  GROUP BY A.Name  ORDER BY NumberOfTracks DESC |
| --- |



#### **Вопрос 6: Кто зарабатывает и кто тратит больше всего?**

Во-первых, найдите, какой артист заработал больше всего по данным из **InvoiceLines**. Теперь используйте этого исполнителя, чтобы определить, какой клиент потратил больше всего на этого исполнителя.

Ваш запрос:

| 1.)  SELECT A.Name AS ArtistName, SUM(IL.UnitPrice \* IL.Quantity) AS TotalEarnings  FROM `da-nfactorial.chinook.artist` A  JOIN `da-nfactorial.chinook.album` AL ON A.ArtistId = AL.ArtistId  JOIN `da-nfactorial.chinook.track` T ON AL.AlbumId = T.AlbumId  JOIN `da-nfactorial.chinook.invoiceline` IL ON T.TrackId = IL.TrackId  GROUP BY A.Name  ORDER BY TotalEarnings DESC  2.)  SELECT C.CustomerId, C.FirstName, C.LastName, SUM(IL.UnitPrice \* IL.Quantity) AS TotalSpent  FROM `da-nfactorial.chinook.customer` C  JOIN `da-nfactorial.chinook.invoice` I ON C.CustomerId = I.CustomerId  JOIN `da-nfactorial.chinook.invoiceline` IL ON I.InvoiceId = IL.InvoiceId  JOIN `da-nfactorial.chinook.track` T ON IL.TrackId = T.TrackId  JOIN `da-nfactorial.chinook.album` AL ON T.AlbumId = AL.AlbumId  JOIN `da-nfactorial.chinook.artist` A ON AL.ArtistId = A.ArtistId  WHERE A.Name = 'Iron Maiden'  GROUP BY C.CustomerId, C.FirstName, C.LastName  ORDER BY TotalSpent DESC |
| --- |

### 

2.)

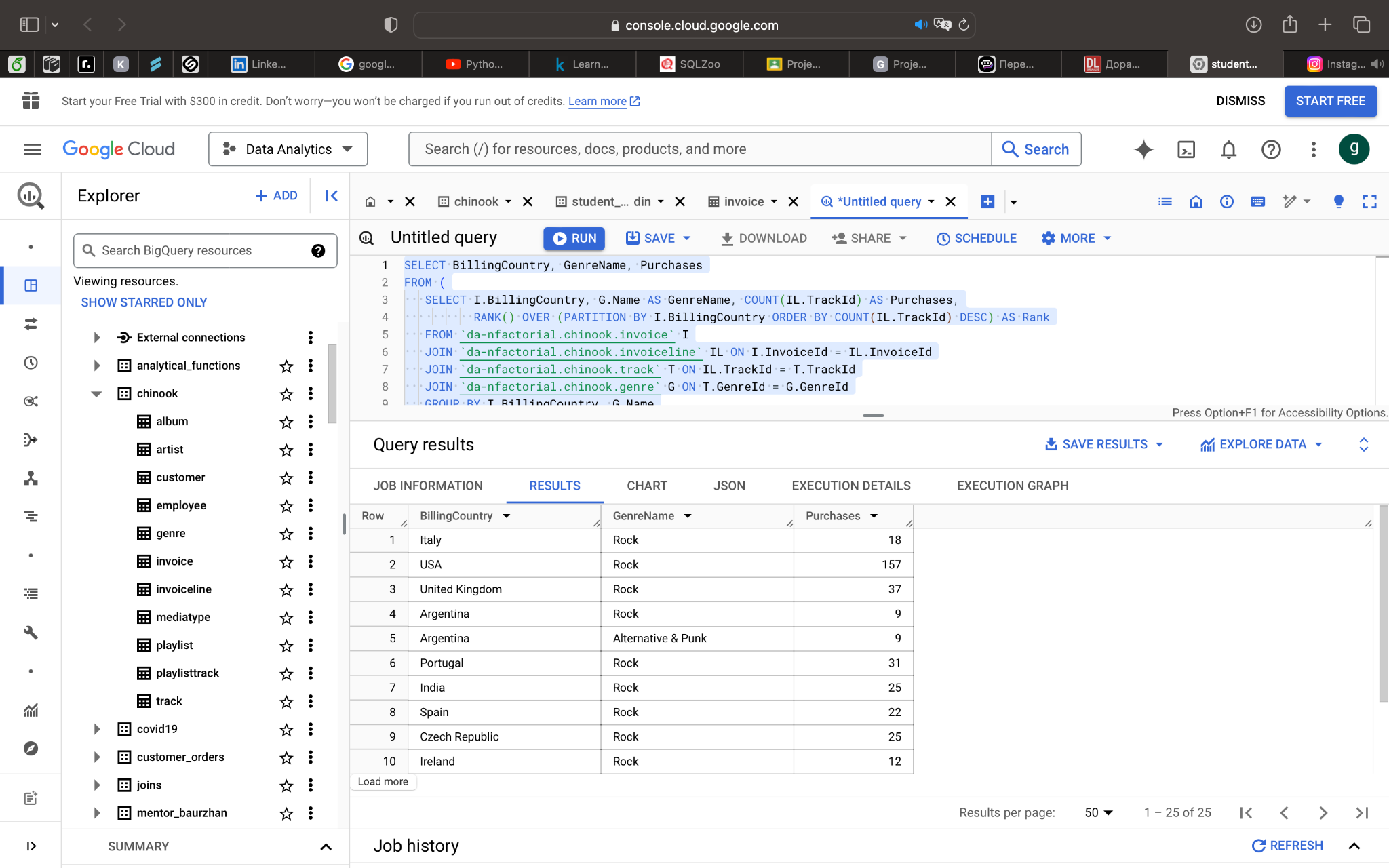
### 

### **Вопрос 7: Какой жанр самый популярный?**

Мы хотим найти самый популярный музыкальный жанр для каждой страны. Мы определяем самый популярный жанр как жанр с наибольшим количеством покупок. Напишите запрос, который возвращает каждую страну вместе с топ 1 жанром. Для стран, где несколько жанров с максимальное количество покупок, верните все жанры соответствующие критериям.

Ваш запрос:

| SELECT BillingCountry, GenreName, Purchases  FROM (  SELECT I.BillingCountry, G.Name AS GenreName, COUNT(IL.TrackId) AS Purchases,  RANK() OVER (PARTITION BY I.BillingCountry ORDER BY COUNT(IL.TrackId) DESC) AS Rank  FROM `da-nfactorial.chinook.invoice` I  JOIN `da-nfactorial.chinook.invoiceline` IL ON I.InvoiceId = IL.InvoiceId  JOIN `da-nfactorial.chinook.track` T ON IL.TrackId = T.TrackId  JOIN `da-nfactorial.chinook.genre` G ON T.GenreId = G.GenreId  GROUP BY I.BillingCountry, G.Name  )  WHERE Rank = 1; |
| --- |



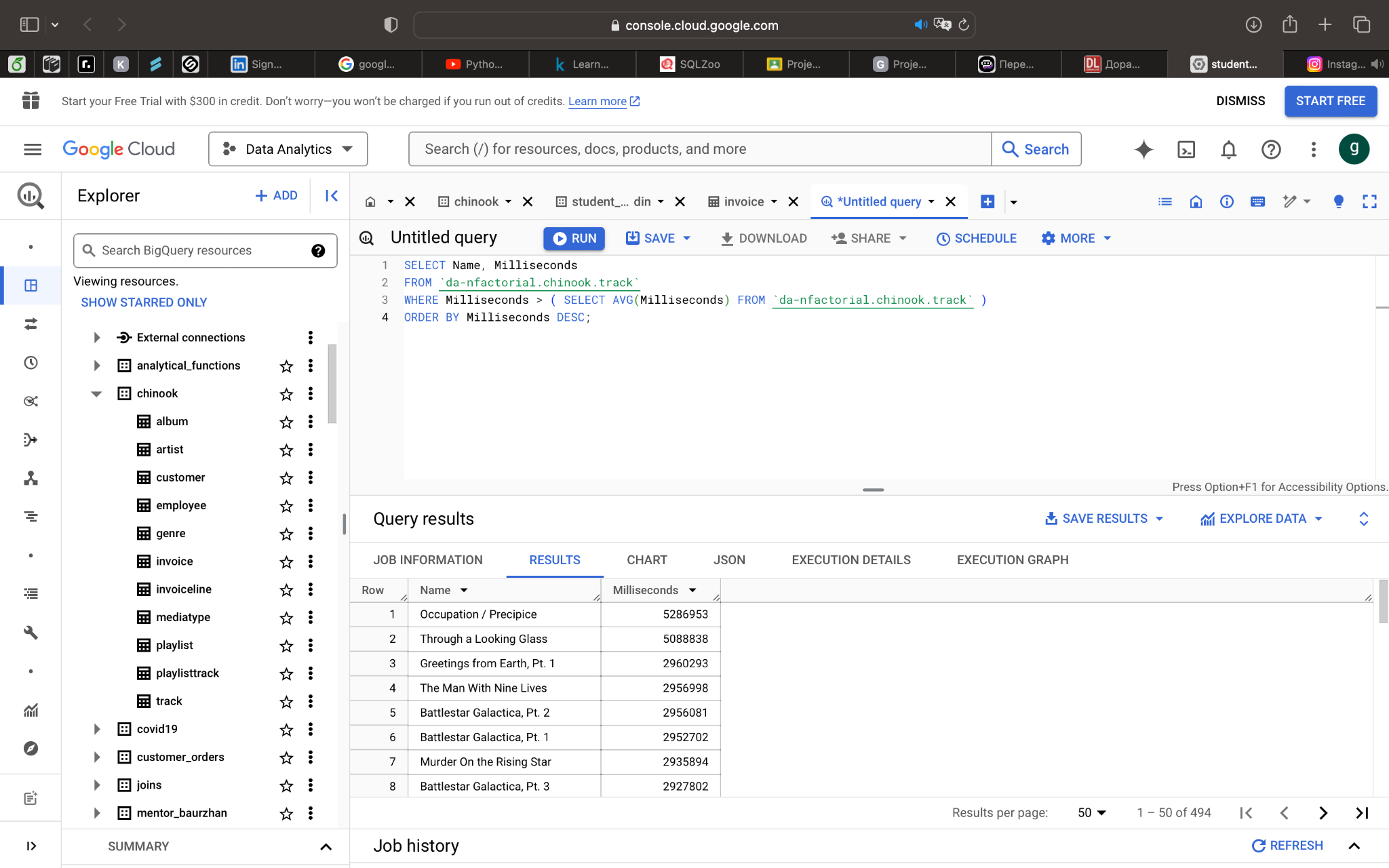
**Вопрос 8: Какая песня самая долгая?**

Предоставьте названия всех песен, длина которых превышает среднюю длину песни в наборе данных.

Предоставьте столбцы **Name** и **Milliseconds** для каждой песни. Упорядочьте по длине песни, самые длинные песни укажите первыми.

Ваш запрос:

| SELECT Name, Milliseconds  FROM `da-nfactorial.chinook.track`  WHERE Milliseconds > ( SELECT AVG(Milliseconds) FROM `da-nfactorial.chinook.track` )  ORDER BY Milliseconds DESC; |
| --- |

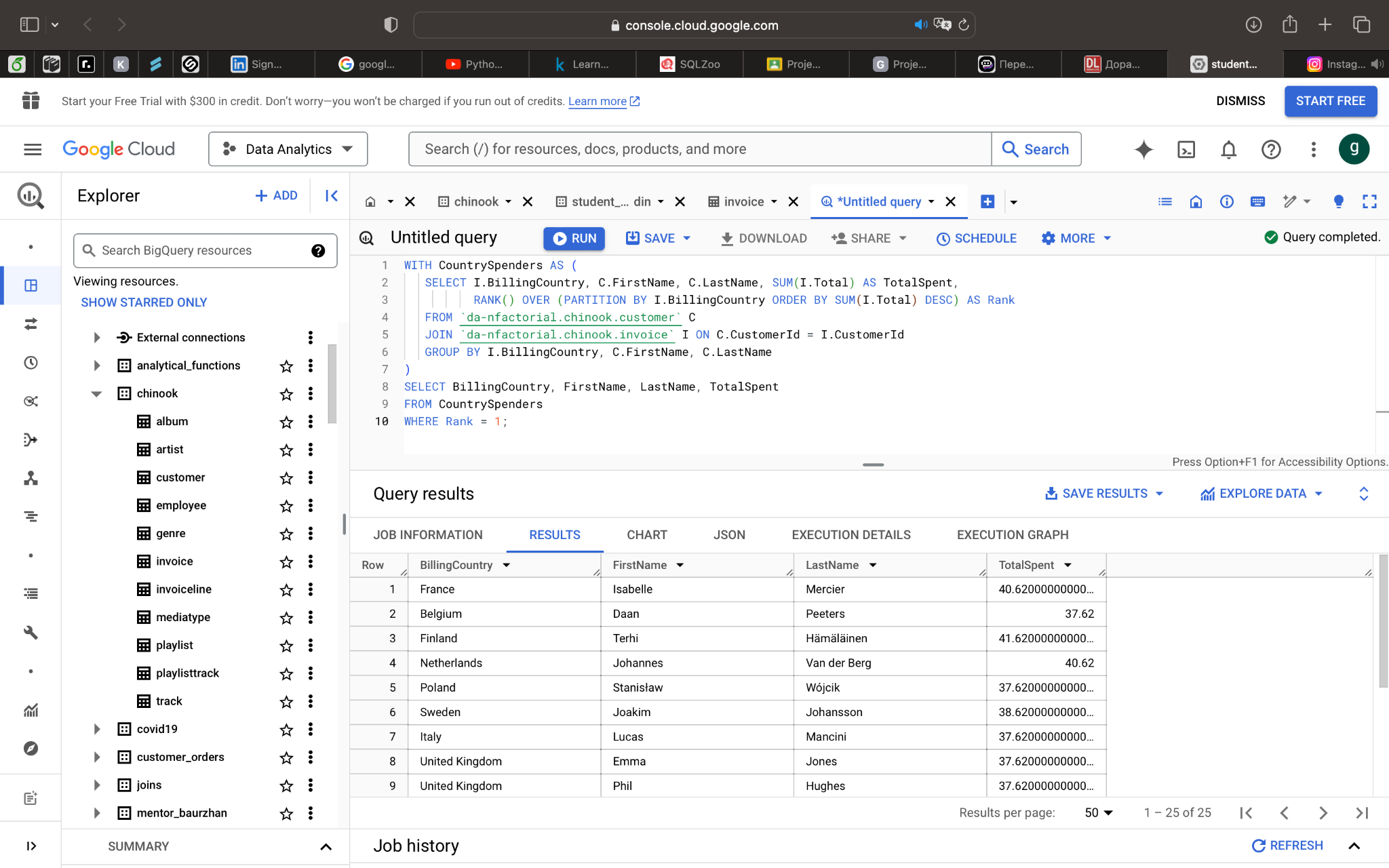


#### **Вопрос 9:**

Напишите запрос, определяющий **клиента, который потратил больше всего на музыку для каждой страны.** Напишите запрос, который возвращает страну вместе с топ 1 покупателем и суммой, которую он потратил. Для стран, в которых есть несколько покупателей потратившие максимальное количество денег, укажите всех клиентов, потративших эту сумму.

Ваш запрос:

| WITH CountrySpenders AS (  SELECT I.BillingCountry, C.FirstName, C.LastName, SUM(I.Total) AS TotalSpent,  RANK() OVER (PARTITION BY I.BillingCountry ORDER BY SUM(I.Total) DESC) AS Rank  FROM `da-nfactorial.chinook.customer` C  JOIN `da-nfactorial.chinook.invoice` I ON C.CustomerId = I.CustomerId  GROUP BY I.BillingCountry, C.FirstName, C.LastName  )  SELECT BillingCountry, FirstName, LastName, TotalSpent  FROM CountrySpenders  WHERE Rank = 1; |
| --- |



## **Часть 2**

* Определите 4 вопроса о Chinook, на который вы хотите ответить на основе анализа данных.
* Затем напишете SQL-запросы, чтобы получить данные, необходимые для успешного ответа на ваши вопросы.
* Визуализируйте полученные данные (используя гистограммы или другие графики), отвечающие на ваш вопрос.
* Объясните ответ в 1-2 предложениях.
* Вопросы, которые вы задаете, зависят от вас, но все четыре запроса должны содержать JOIN и AGGREGATION.

**Вопрос 1:**

### 

### **Какой артист заработал больше всего?**

Ваш запрос и объяснение с графиком:

SELECT A.Name AS ArtistName, SUM(IL.UnitPrice \* IL.Quantity) AS TotalEarnings

FROM `da-nfactorial.chinook.artist` AS A

JOIN `da-nfactorial.chinook.album` AS ALB ON A.ArtistId = ALB.ArtistId

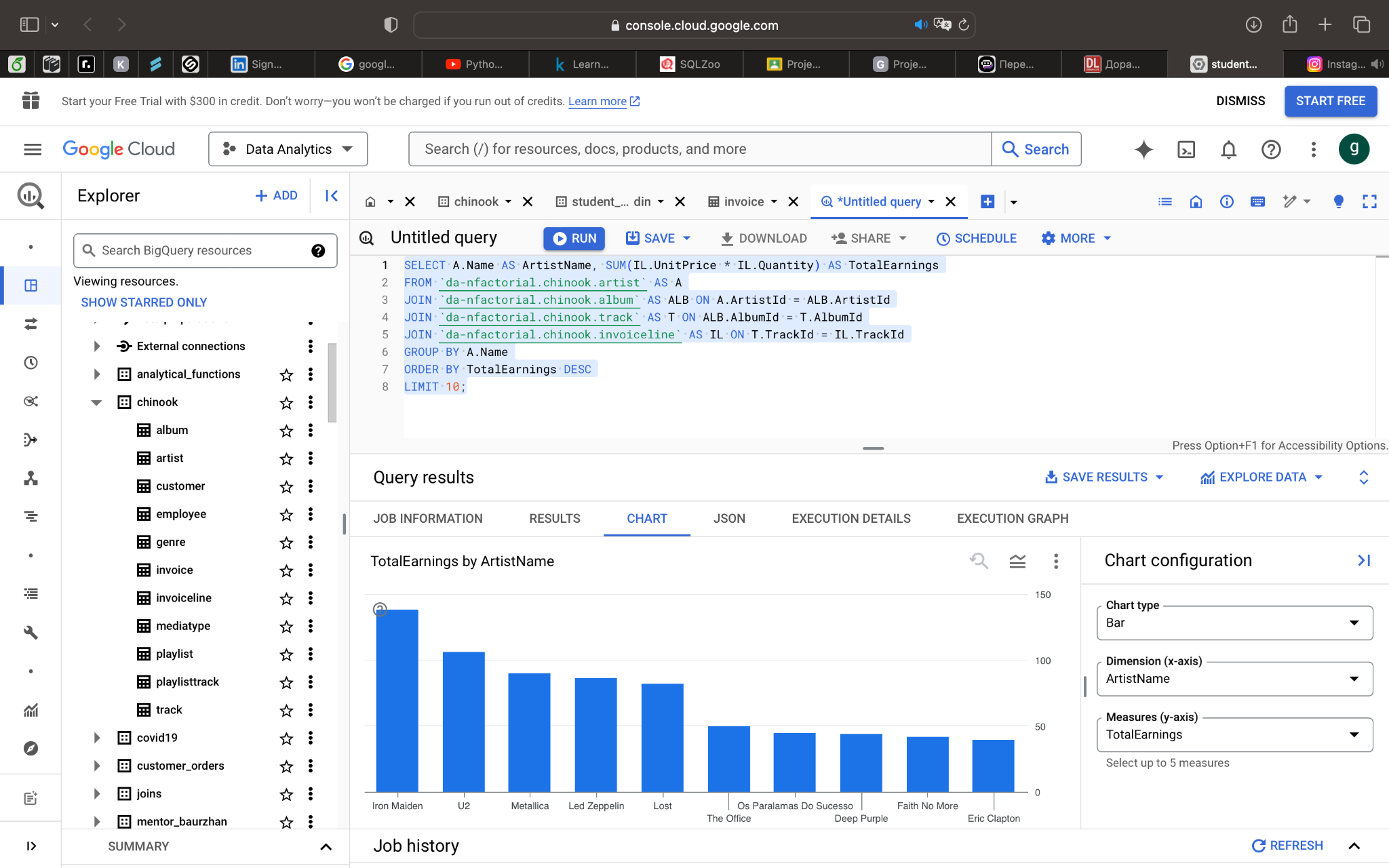
JOIN `da-nfactorial.chinook.track` AS T ON ALB.AlbumId = T.AlbumId

JOIN `da-nfactorial.chinook.invoiceline` AS IL ON T.TrackId = IL.TrackId

GROUP BY A.Name

ORDER BY TotalEarnings DESC

LIMIT 10;



!!! Артист с наибольшим доходом является важным показателем успеха в музыкальной индустрии и может помочь в планировании маркетинга.

**Вопрос 2:**

### **Какое количество треков куплено каждым клиентом?**

Ваш запрос и объяснение с графиком:

SELECT C.FirstName, C.LastName, COUNT(IL.TrackId) AS TracksPurchased

FROM `da-nfactorial.chinook.customer` AS C

JOIN `da-nfactorial.chinook.invoice` AS I ON C.CustomerId = I.CustomerId

JOIN `da-nfactorial.chinook.invoiceline` AS IL ON I.InvoiceId = IL.InvoiceId

GROUP BY C.CustomerId, C.FirstName, C.LastName

ORDER BY TracksPurchased DESC

LIMIT 10;



Понимание покупок клиентов может помочь в создании целевых акций и улучшении услуг.

**Вопрос 3:**

### **Каковы средние расходы клиентов по странам?**

Ваш запрос и объяснение с графиком:

SELECT C.Country, AVG(IL.UnitPrice \* IL.Quantity) AS AverageSpent

FROM `da-nfactorial.chinook.customer` AS C

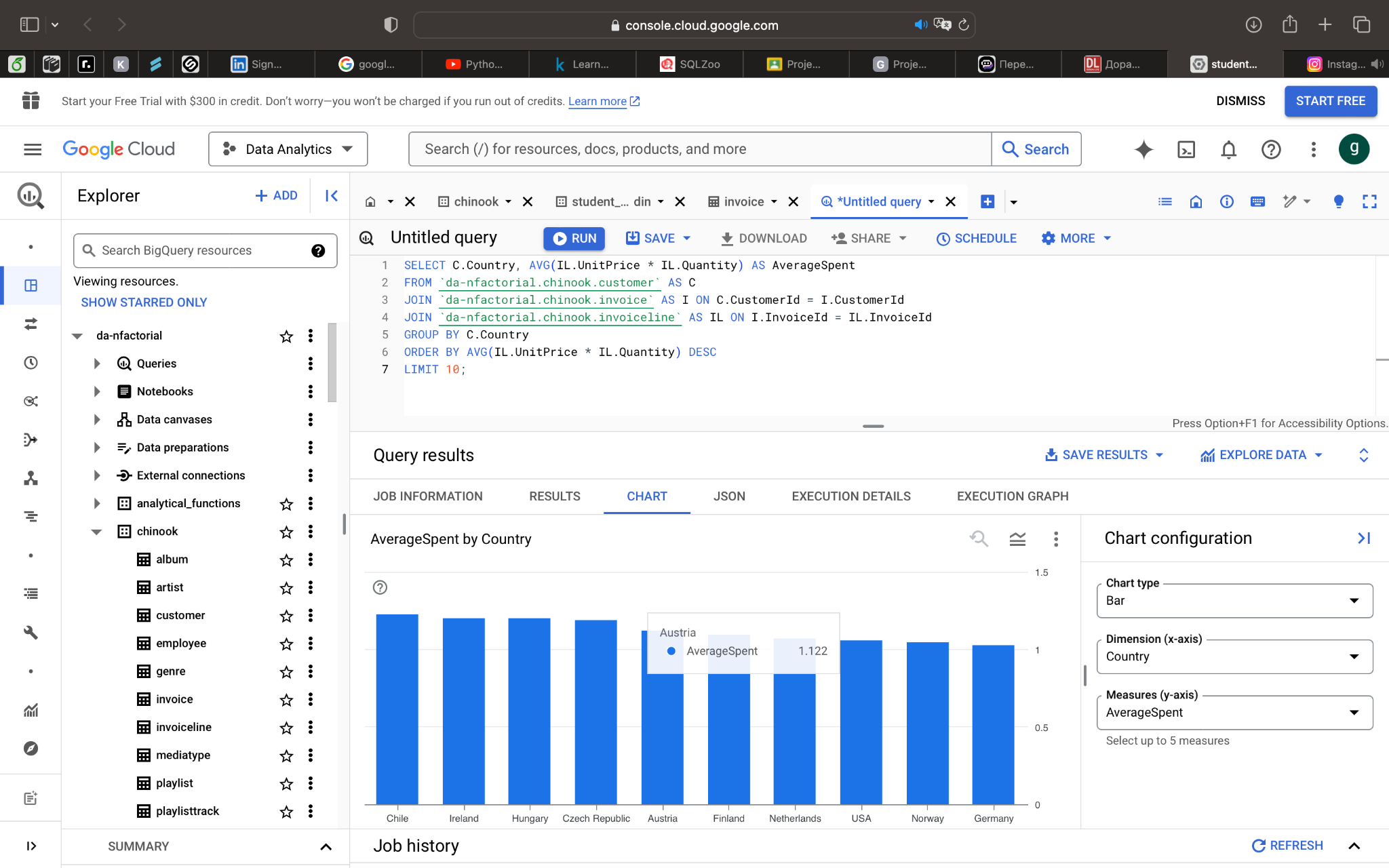
JOIN `da-nfactorial.chinook.invoice` AS I ON C.CustomerId = I.CustomerId

JOIN `da-nfactorial.chinook.invoiceline` AS IL ON I.InvoiceId = IL.InvoiceId

GROUP BY C.Country

ORDER BY AVG(IL.UnitPrice \* IL.Quantity) DESC

LIMIT 10;



Эти данные помогают выявить, в каких странах клиенты тратят больше всего, что может повлиять на стратегию продаж.

**Вопрос 4:**

### **Какое количество альбомов было продано по жанрам?**

Ваш запрос и объяснение с графиком:

SELECT G.Name AS GenreName, COUNT(AL.AlbumId) AS AlbumsSold

FROM `da-nfactorial.chinook.genre` AS G

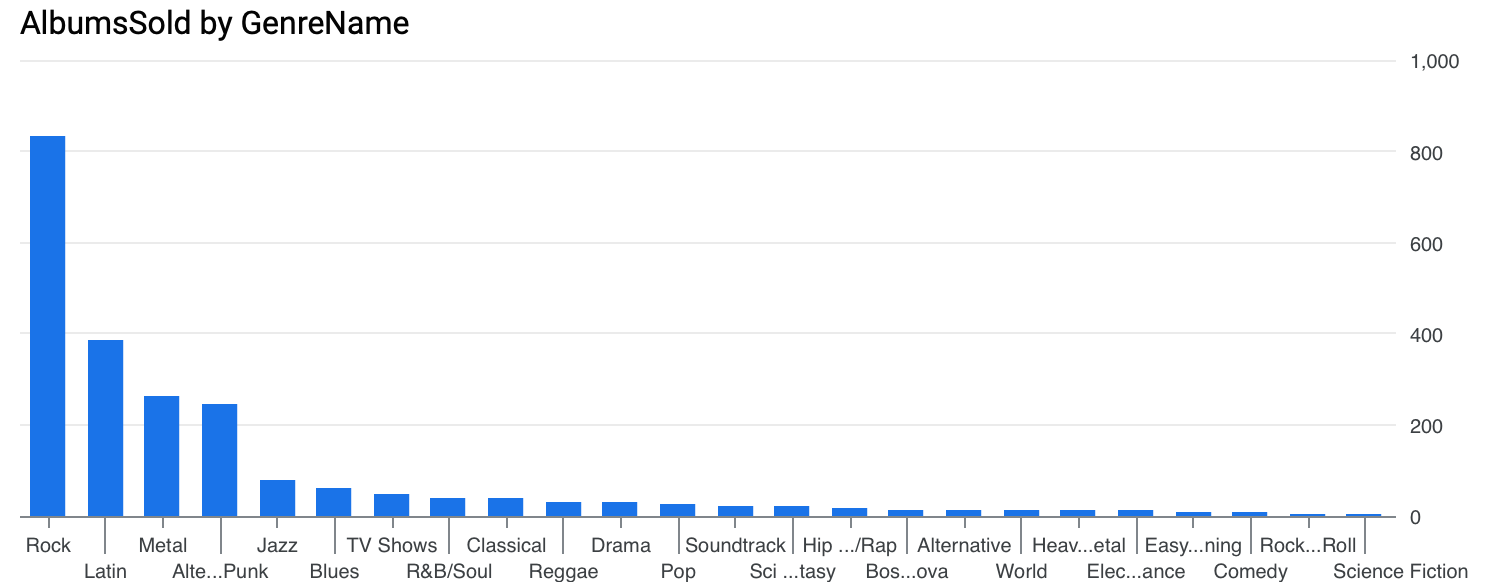
JOIN `da-nfactorial.chinook.track` AS T ON G.GenreId = T.GenreId

JOIN `da-nfactorial.chinook.album` AS AL ON T.AlbumId = AL.AlbumId

JOIN `da-nfactorial.chinook.invoiceline` AS IL ON T.TrackId = IL.TrackId

GROUP BY G.Name

ORDER BY AlbumsSold DESC;



Анализ продаж альбомов по жанрам позволяет понять музыкальные предпочтения клиентов и адаптировать предложения.

Вы закончили свой первый SQL анализ! Отличная работа!