**Data Preparation with SQL – Creating a View**

Throughout this Tracking User Engagement with SQL, Excel, and Python project, you’ll work with a real dataset from our company’s data. The project requires you to analyze whether the new additions to the platform (new courses, exams, and career tracks) have increased student engagement.

The first half of 2022 was expected to be profitable for the company. The reason was the hypothesized increased student engagement after the release of several new features on the company’s website at end-2021. These include enrolling in career tracks and testing your knowledge through practice, course, and career track exams. Of course, we have also expanded our course library to increase user engagement and the platform’s audience as more topics are covered. By comparing different metrics, we can measure the effectiveness of these new features and the overall engagement of our users.

**I. Calculating a Subscription’s End Date**

Use the student\_purchases table from the data\_scientist\_project database to create a result set with the following columns:

* purchase\_id
* student\_id
* plan\_id
* date\_start
* date\_end
* date\_refunded

The date\_start column is the renamed date\_purchased column from the database, adjusted for consistency with the subsequent date\_end column.

To calculate the end date of a subscription (date\_end), add one month, three months, or 12 months to the start date of a subscription for a Monthly (represented as 0 in the plan\_id column), Quarterly (1), or an Annual (2) purchase, respectively.

The only exception is the lifetime subscription (denoted by 3), which has no end date. Refunds will be handled in the following task: II. Re-Calculating a Subscription’s End Date.

**Hint 1:***Research MySQL’s DATE\_ADD function.*

**Hint 2:***You can refer to the Customer Engagement Analysis with SQL and Tableau course (the Retrieving Relevant Data from The Database section), which explains this query in detail.*

**Sanity Check:***Ensure your table has 18,207 rows—i.e., you’ve found each subscription’s start and end dates.*

**II. Re-Calculating a Subscription’s End Date**

Using the query from the previous task (I. Calculating a Subscription’s End Date) as a sub-query, create a new one retrieving the following columns:

* purchase\_id
* student\_id
* plan\_id
* date\_start
* date\_end

Re-calculate the date\_end column so that if an order was refunded—indicated by a non-NULL value in the date\_refunded field—the student’s subscription terminates at the refund date.

**Sanity Check:***Ensure your view has 18,207 rows.*

**III. Creating Two ‘paid’ Columns and a MySQL View**

Using the query you designed in the previous task (II. Re-Calculating a Subscription’s End Date), create a new SQL query that, when executed, stores in the data\_scientist\_project schema a view called purchases\_info which we’ll use in subsequent parts. The view should include the following columns:

* purchase\_id
* student\_id
* plan\_id
* date\_start
* date\_end
* paid\_q2\_2021
* paid\_q2\_2022

The paid\_q2\_2021 and paid\_q2\_2022 columns contain binary values indicating whether a student had an active subscription during the respective year’s second quarter (April 1 to June 30, inclusive). A0 in the column indicates a free-plan student in Q2, while a 1 represents an active subscription in that period.

**Data Preparation with SQL – Splitting Into Periods**

Great job! We created a view in the schema called purchases\_info, which stores information about students’ subscriptions and whether these subscription periods overlap with the second quarters of 2021 or 2022. Now, we’ll utilize purchases\_info to classify students as free-plan and paying in Q2 2021 and Q2 2022.

**I. Calculating Total Minutes Watched in Q2 2021 and Q2 2022**

We’re now interested in analyzing the engagement of our users in terms of the total minutes watched during Q2 2021 and Q2 2022 separately. Additionally, we want to identify which users were paid subscribers during each of these periods.

Your task is to write an SQL query that returns the following columns:

* student\_id – a list of student IDs
* minutes\_watched – the total minutes students have watched in both periods—return a separate table for each period

The information about the minutes watched by each student is available in the student\_video\_watched table.

**Sanity Check:** *Ensure you have 7,639 rows in the result set representing students who watched a lecture in Q2 2021. Then, confirm you have 8,841 rows in the result set representing students who watched a lecture in Q2 2022.*

**I. Creating a ‘paid’ Column**

Now, use the query you designed in the previous task (I. Calculating Total Minutes Watched in Q2 2021 and Q2 2022) to create a result set with the following columns:

* student\_id
* minutes\_watched
* paid\_in\_q2

The last column indicates whether a student had an active subscription in Q2 (represented by 1) or not (represented by 0).

Retrieve the following four datasets and store them in the corresponding CSV files:

* Students engaged in Q2 2021 who haven’t had a paid subscription in Q2 2021 (minutes\_watched\_2021\_paid\_0.csv)
* Students engaged in Q2 2022 who haven’t had a paid subscription in Q2 2022 (minutes\_watched\_2022\_paid\_0.csv)
* Students engaged in Q2 2021 who have been paid subscribers in Q2 2021 (minutes\_watched\_2021\_paid\_1.csv)
* Students engaged in Q2 2022 who have been paid subscribers in Q2 2022 (minutes\_watched\_2022\_paid\_1.csv)

**Note:***Remember that the same student can have multiple subscription records in the purchases\_info view.*

**Sanity check:** *Ensure you retrieve the following number of rows for each case, respectively:*

* 5,334
* 2,305
* 6,055
* 2,786

**Data Preparation with SQL – Certificates Issued**

In the previous two sections, you created a MySQL view and used it to classify students as free-plan or paying on a given date. In this short task, you’ll retrieve information on the minutes watched and the certificates issued to a student. Later in the project, we’ll study the correlation between these two metrics.

**I. Studying Minutes Watched and Certificates Issued**

For this task, consider only the students who’ve been issued a certificate. Create an SQL query to extract the following information for each such student:

* The student ID
* The total minutes watched
* The total number of certificates issued

Assign the corresponding value for students with no minutes recorded as 0. Save the resulting table as minutes\_and\_certificates.csv for later use.

**Sanity Check:***Ensure your table has a total of 658 rows.*

**Data Preprocessing with Python – Removing Outliers**

Excellent work! You should now be equipped with the following CSV files:

* minutes\_watched\_2021\_paid\_0.csv
* minutes\_watched\_2022\_paid\_0.csv
* minutes\_watched\_2021\_paid\_1.csv
* minutes\_watched\_2022\_paid\_1.csv
* minutes\_and\_certificates.csv

We’re now ready to switch technologies and open Jupyter Notebook, where we’ll study the data with the help of distribution plots and remove the outliers so that they don’t skew the analysis we’ll perform later in this project. The following tasks will use the first four CSV files listed above.

**I. Plotting the Distributions**

Plot the distribution of the minutes\_watched variable of each of the four datasets and examine its shape. Are the distributions skewed? If yes, how? What does this tell us about the distribution of minutes watched?

**Hint:***Research pandas’ kdeplot() method. You can create four subplots displaying all four distributions simultaneously for better clarity.*

**II. Removing the Outliers**

Remove the outliers of the data—for each of the four datasets, keep the values lower than the 99th percentile.

Once you’ve retrieved the final datasets, save them as four separate CSV files on your computer:

* minutes\_watched\_2021\_paid\_0\_no\_outliers.csv
* minutes\_watched\_2022\_paid\_0\_no\_outliers.csv
* minutes\_watched\_2021\_paid\_1\_no\_outliers.csv
* minutes\_watched\_2022\_paid\_1\_no\_outliers.csv

**Hint:***To save a DataFrame as a CSV file, use the following line of code:*

df\_name.to\_csv('file\_name.csv', index=False)

*If you don’t specify a path at the beginning of the string, the CSV file will be stored in the same directory as your Jupyter Notebook document.*

**Data Analysis with Excel – Hypothesis Testing**

You’re making significant progress! In the previous part (Data Preprocessing with Python – Removing Outliers), we removed the outliers from four of the datasets and stored the results in the following CSV files:

* minutes\_watched\_2021\_paid\_0\_no\_outliers.csv
* minutes\_watched\_2022\_paid\_0\_no\_outliers.csv
* minutes\_watched\_2021\_paid\_1\_no\_outliers.csv
* minutes\_watched\_2022\_paid\_1\_no\_outliers.csv

Now, open a new Excel sheet and paste the content of each file into a separate tab within the same worksheet. Once you’ve done that, you are ready to proceed with the following tasks.

**I. Calculating Mean and Median Values**

Calculate the mean and median minutes watched by students in the four groups. How does the median compare with the mean in each group? Referring to the distribution plots you created in Data Preprocessing with Python – Removing Outliers, does this result meet your expectations?

##### II. Calculating Confidence Intervals

For each of the four groups, find the minute interval for which you are 95% confident a random person will fall in that interval. Assume a normal distribution.

What conclusions can you draw about students’ engagement in Q2 2021 and Q2 2022 for both free-plan and paying students?

**Optional:***Create a confidence interval bar chart to support your arguments better.*

**III. Performing Hypothesis Testing**

You want to reach a data-driven decision on whether the new features (courses, career tracks, and exams) contribute to the increased number of minutes watched on the platform for free-plan and paying students—i.e., increased student engagement in their study process. You use hypothesis testing on both groups (free-plan and paying) for 2021 and 2022.

Let your null and alternative hypotheses (respectively) be:

* The engagement (minutes watched) in Q2 2021 is higher than or equal to the one in Q2 2022 (μ1≥μ2)(�1≥�2). We test free-plan and paying students separately.
* The engagement (minutes watched) in Q2 2021 is lower than the one in Q2 2022 (μ1<μ2)(�1<�2). We test free-plan and paying students separately.

Additionally, make the following assumptions:

* Assume a normal distribution.
* For free-plan students, perform a two-sample t-test assuming equal variances.
* For paying students, perform a two-sample t-test assuming unequal variances.

**Optional:***Perform a two-sample f-test for variances to support the assumptions.*

What conclusion can you draw from this test? Comment on the results of committing a Type I or a Type II error in this study. Which one would result in higher costs to the company?

## Data Analysis with Excel – Correlation Coefficients

You’re approaching the end of the Excel part of the project. In the previous part (Data Analysis with Excel – Hypothesis Testing), you performed a hypothesis analysis and concluded whether the additions to the platform had increased the minutes watched by students.

Now, you’ll analyze the correlation between the minutes watched on the platform and the certificates issued. You’ll work with the minutes\_and\_certificates.csv file created in the Data Preparation with SQL – Certificates Issued part.

##### I. Calculating Correlation Coefficients

Find the correlation coefficient between the minutes watched and the certificates issued. Interpret the results.

**Optional:** Create a scatter plot to support your arguments better.

## Dependencies and Probabilities

In this part of the project, we analyze the engaged students on the platform. Return to the data\_scientist\_project database and consider **all** students who’ve watched a lecture in Q2 2021 and those who’ve watched a lecture in Q2 2022 as two sets. Let the universal set be **all** students who’ve watched a lecture on the platform—the union of the two sets defined above. Don’t omit any outliers we’ve removed during this project.

##### I. Assessing Event Dependencies

Determine if watching a lecture in Q2 2021 and Q2 2022 are dependent or independent events. Explain your result.

##### II. Calculating Probabilities

What is the probability that a student has watched a lecture in Q2 2021, given that they’ve watched a lecture in Q2 2022?

**Data Prediction with Python**

Congratulations! You’ve applied various technologies like SQL, Python, Excel, and pure theory to extract, preprocess, analyze, and compare data concerning student engagement in Q2 2021 and Q2 2022. You’ve now reached the final part of this project, which deals with applying machine learning techniques. Use the minutes\_and\_certificates.csv file you extracted in the Data Preparation with SQL – Certificates Issued part. Let’s begin!

**I. Creating a Linear Regression**

This part aims to perform a linear regression using the minutes\_watched column as a predictor and certificates\_issued as a target. Having done that, answer the following:

1. What is the linear equation that explains the behavior of the relationship?
2. What is the R-squared value of the regression? How would you interpret it?
3. What is the predicted number of certificates taken by a student who has watched 1200 minutes of content? (Round your result up to the nearest integer.)

**Note:***Use 20% of your data as a test set. Use the number 365 as a random state.*