DATABASE FOUNDATIONS

Assignment 3

Shabnam Heidaripour

24/10/2024

Description of the steps – Importing csv file into PgAdmin

As suggested on the course material I downloaded the dataset from Kaggle. This dataset shows ‘Students Adaptability Level in Online Education’ which we can use for analyzing the adaptability of students in online education. The reason for choosing this dataset is my personal interest in the field of education and how we can receive and analyze data for using it in the educational system.

As we learned, after downloading the CSV file, we open it in VS Code and start by doing data cleaning. By using the Rainbow CSV extension we have a clearer view of this dataset and by using Edit CSV extension I will perform the following cleaning steps:

Step 1 - Deleting Unnecessary Columns:

Here are the columns in the dataset:
Gender, Age, Education Level, Institution Type, IT Student, Location, Load-shedding, Financial Condition, Internet Type, Network Type, Class Duration, Self Lms, Device, Adaptivity Level

I decided to remove the **Location** column because it only contains "Yes" and "No", and this data does not help with our analysis, It doesn’t give clear information neither.
I also removed the **Network Type** column because we already mentioned the type of internet connection and that is enough for our analysis.

Step 2 - Character Encoding:

The next step was to check and ensure that the character encoding was correct. I confirmed that the encoding was set to UTF-8 in VS Code. (This step will also be done in PgAdmin)

Step 3 - Handling Missing Information:

I used Ctrl+F to search for missing values, there were no empty cells.

Step 4 - Removing Commas:

In the import step we can choose the delimeter as comma (,) in the options.

Step 5 - Creating the Table in PgAdmin:

Now, it's time to create the table in PgAdmin. First, I reviewed each column to define the correct data type:

- Num: int - I had to create an additional column named 'num' for uniquely identifying the data, so with this we can have a unique value for each row in the dataset and use it as the primary key.
- Gender: varchar(20) – At the moment, the dataset has two genders: Boy and Girl, but we consider a bigger space for future possibilities (e.g., Non-binary or Transgender) to avoid needing to change the database later.
- Age: varchar(10) – The age is given as a range (e.g., 21-25), so I assigned 5 characters.
- Education Level: varchar(30) – Different stages of education require some space.
- Institution Type: varchar(30) – This can vary between different types of institutions.
- IT Student: varchar(3) – We only have "Yes" or "No", so 3 characters are enough.
- Load-shedding: varchar(20) – It can be "None", "Low", or "High".
- Financial Condition: varchar(20) – This is described as "Poor", "Mid", and "High".
- Internet Type: varchar(20) – Options are "Wifi" and "Mobile Data".

- Class Duration: varchar(10) – We have ranges of duration, so 5 characters should be enough.
- Self Lms: varchar(3) – Since the options are just "Yes" or "No", 3 characters are enough.
- Device: varchar(20) – The device names might vary, so we allow enough space.
- Adaptivity Level: varchar(20) – We have "Low", "Medium", and "High".

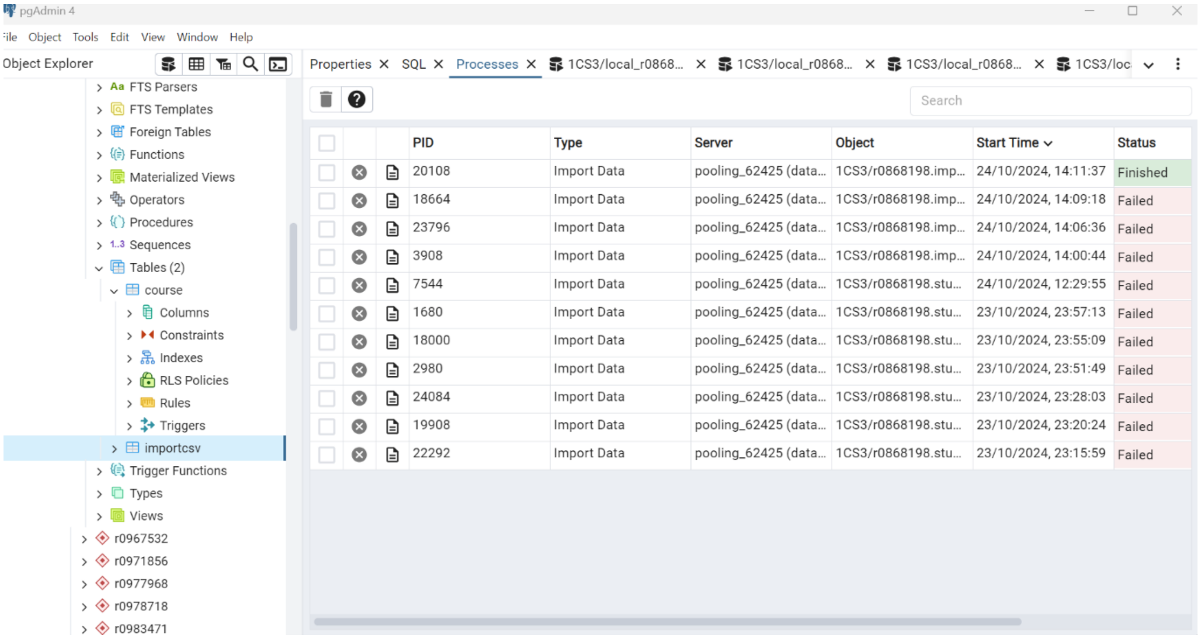
Here’s the SQL code I used to create the table:

```
SET search_path TO r0868198;
CREATE TABLE importcsv(
    num        int,
    gender         varchar(20),
    age           varchar(10),
    education_level   varchar(30),
    institution_type  varchar(30),
    it_student      varchar(3),
    load_shedding     varchar(20),
    financial_condition varchar(20),
    internet_type     varchar(20),
    class_duration    varchar(10),
    self_lms        varchar(3),
    device         varchar(20),
    adaptivity_level  varchar(20),
    CONSTRAINT pk_importcsv PRIMARY KEY(num)
);
```

After creating the table in PgAdmin, and importing the CSV file multiple times into PgAdmin and encountering various errors, I learned a lot through the process of reading and understanding error messages. It took me some time, but each error taught me something new about how to handle dataset issues and syntax mistakes in SQL. Although it was a time-consuming process, I eventually figured out how to correctly how to make that my dataset was properly cleaned and formatted. This experience taught me that problem-solving is a big part of working in IT, and that making mistakes is just a step toward finding the right solution.

In conclusion, while I had some trouble with defining the Primary Key and adjusting data lengths, I managed to successfully import the dataset after resolving these issues. The process not only helped me understand the technical steps but also gave me more confidence in troubleshooting errors effectively.

Please see the pictures below, which show how many times I encountered errors and the time each occurred. (😊But I learned!)



			PID	Type	Server	Object	Start Time	Status
□	⊗	🗎	20108	Import Data	pooling_62425 (data...	1CS3/r0868198.imp...	24/10/2024, 14:11:37	Finished
□	⊗	🗎	18664	Import Data	pooling_62425 (data...	1CS3/r0868198.imp...	24/10/2024, 14:09:18	Failed
□	⊗	🗎	23796	Import Data	pooling_62425 (data...	1CS3/r0868198.imp...	24/10/2024, 14:06:36	Failed
□	⊗	🗎	3908	Import Data	pooling_62425 (data...	1CS3/r0868198.imp...	24/10/2024, 14:00:44	Failed
□	⊗	🗎	7544	Import Data	pooling_62425 (data...	1CS3/r0868198.stu...	24/10/2024, 12:29:55	Failed
□	⊗	🗎	1680	Import Data	pooling_62425 (data...	1CS3/r0868198.stu...	23/10/2024, 23:57:13	Failed
□	⊗	🗎	18000	Import Data	pooling_62425 (data...	1CS3/r0868198.stu...	23/10/2024, 23:55:09	Failed
□	⊗	🗎	2980	Import Data	pooling_62425 (data...	1CS3/r0868198.stu...	23/10/2024, 23:51:49	Failed
□	⊗	🗎	24084	Import Data	pooling_62425 (data...	1CS3/r0868198.stu...	23/10/2024, 23:28:03	Failed
□	⊗	🗎	19908	Import Data	pooling_62425 (data...	1CS3/r0868198.stu...	23/10/2024, 23:20:24	Failed
□	⊗	🗎	22292	Import Data	pooling_62425 (data...	1CS3/r0868198.stu...	23/10/2024, 23:15:59	Failed